NORTH ATLANTIC TREATY ORGANISATION

RESEARCH AND TECHNOLOGY ORGANISATION

AC/323(SAS-045)TP/52

www.rto.nato.int

RTO TECHNICAL REPORT **TR-SAS-045**

Computer Based Decision Support Tool for Helicopter Mission Planning in Disaster Relief and Military Operations

(Outil informatique d'aide à la décision pour la planification des missions d'hélicoptères dans des opérations militaires et de secours en cas de catastrophe)

Final Report of Task Group SAS-045.

Published June 2008

Distribution and Availability on Back Cover

NORTH ATLANTIC TREATY ORGANISATION

RESEARCH AND TECHNOLOGY ORGANISATION

AC/323(SAS-045)TP/52

www.rto.nato.int

RTO TECHNICAL REPORT **TR-SAS-045**

Computer Based Decision Support Tool for Helicopter Mission Planning in Disaster Relief and Military Operations

(Outil informatique d'aide à la décision pour la planification des missions d'hélicoptères dans des opérations militaires et de secours en cas de catastrophe)

Final Report of Task Group SAS-045.

The Research and Technology Organisation (RTO) of NATO

RTO is the single focus in NATO for Defence Research and Technology activities. Its mission is to conduct and promote co-operative research and information exchange. The objective is to support the development and effective use of national defence research and technology and to meet the military needs of the Alliance, to maintain a technological lead, and to provide advice to NATO and national decision makers. The RTO performs its mission with the support of an extensive network of national experts. It also ensures effective co-ordination with other NATO bodies involved in R&T activities.

RTO reports both to the Military Committee of NATO and to the Conference of National Armament Directors. It comprises a Research and Technology Board (RTB) as the highest level of national representation and the Research and Technology Agency (RTA), a dedicated staff with its headquarters in Neuilly, near Paris, France. In order to facilitate contacts with the military users and other NATO activities, a small part of the RTA staff is located in NATO Headquarters in Brussels. The Brussels staff also co-ordinates RTO's co-operation with nations in Middle and Eastern Europe, to which RTO attaches particular importance especially as working together in the field of research is one of the more promising areas of co-operation.

The total spectrum of R&T activities is covered by the following 7 bodies:

- AVT Applied Vehicle Technology Panel
- HFM Human Factors and Medicine Panel
- IST Information Systems Technology Panel
- NMSG NATO Modelling and Simulation Group
- SAS System Analysis and Studies Panel
- SCI Systems Concepts and Integration Panel
- SET Sensors and Electronics Technology Panel

These bodies are made up of national representatives as well as generally recognised 'world class' scientists. They also provide a communication link to military users and other NATO bodies. RTO's scientific and technological work is carried out by Technical Teams, created for specific activities and with a specific duration. Such Technical Teams can organise workshops, symposia, field trials, lecture series and training courses. An important function of these Technical Teams is to ensure the continuity of the expert networks.

RTO builds upon earlier co-operation in defence research and technology as set-up under the Advisory Group for Aerospace Research and Development (AGARD) and the Defence Research Group (DRG). AGARD and the DRG share common roots in that they were both established at the initiative of Dr Theodore von Kármán, a leading aerospace scientist, who early on recognised the importance of scientific support for the Allied Armed Forces. RTO is capitalising on these common roots in order to provide the Alliance and the NATO nations with a strong scientific and technological basis that will guarantee a solid base for the future.

The content of this publication has been reproduced
directly from material supplied by RTO or the authors.

Published June 2008

ISBN 978-92-837-0076-0

Table of Contents

List of Figures

List of Tables

Acknowledgements

This report is prepared to fulfil the requirements of the Programme of Work of NATO SAS-045 RTG which has been supported by France, Germany, Poland, the Netherlands and Turkey on "Computer Based Decision Support Tool for Helicopter Mission Planning in Disaster Relief and Military Operations". We would like to express our deepest gratitude to those in NATO RTA and RTB, who have unanimously contributed to this study.

May 2005

Gülay Barbarosoğlu (Team leader), Turkey

Altan Özkıl (Executıve), Turkey

Davut Aslan, Turkey

Hakan Canlı, Turkey

Mickaël Fontaine, France

Orhan Karasakal, Turkey

Robert Le Fevre, The Netherlands

Patrick Journée, France

Ernst Moellmann, Germany

Hüseyin Onur Mete, Turkey

Marcel Smit, The Netherlands

Ramazan Toper, Turkey

SAS-045 Membership List

FRANCE

Mr. Mickaël FONTAINE
DSP/CAD
16bis avenue Prieur de la Côte d'Or
94114 Arcueil Cedex

Mr. Patrick JOURNEE
Délégation Générale d'Armement
Centre d'analyse de défense
16bis avenue Prieur de la Côte d'Or
94114 Arceuil Cedex

GERMANY

Maj. E. MOELLMANN
Heeresamt III 7 (1)
Bruhler Str. 300
50968 Koln

THE NETHERLANDS

Dr. Marcel SMIT
TNO/FEL, Div1, Grp1-6
Oude Waalsdorperweg 63
PO Box 96864
2509 JG The Hague

POLAND

Col. Prof. Z. SWIATNICKI
Institute of Command Systems Automation and Logistics
Military University of Technology
ul.Kaliskiego 2
00-908 Warszawa

TURKEY

Capt. Davut ASLAN
Turkish Air Force Command TAFC
Hava Kuvvetleri Lomutanligi
Bilkardes Sb. Md.lugu
Bakanliklar
06100 Ankara

Prof. Gülay BARBAROSOGLU
Bogazici University
Chairman of Industrial Engineering Department
Cevdet Pasa Caddesi
34342 Bebek
Istanbul

TURKEY (cont'd)

Lt. Hakan CANLI
TAFC
Scientific Decision Support Branch
Hava Kuvvetleri Lomutanligi
Bilkardes Sb. Md.lugu
Bakanliklar
06100 Ankara

Lt. Orhan KARASAKAL
TAFC Deniz Kuvvetleri K.ligi
APGE Baskanligi Hasa S.
Bakanliklar
06100 Ankara

Capt. R. KUCUKEROL
Turkish General Staff
Genelkurmay Baskanligi
Lojustik Bsk.ligi
Bakanliklar
06100 Ankara

Mr. Huseyin Onur METE
Department of Industrial Engineering
Bogazici University
34342 Bebek
Istanbul

Lt.Col. Dr. Altan ÖZKIL
Turkish General Staff HQ Bilkardem Baskanligi
Bakanliklar
06100 Ankara

Capt. G. REYHAN
TLFC
Kara Kuvvetleri Komutanligi
Bakanliklar
06100 Ankara

1st Lt. R. TOPER
TAFC
Hava Kuvvetleri Lomutanligi
Bilkardes Sb. Md.lugu
Bakanliklar
06100 Ankara

Computer Based Decision Support Tool for Helicopter Mission Planning in Disaster Relief and Military Operations

(RTO-TR-SAS-045)

Executive Summary

The establishment of the Euro-Atlantic Disaster Response Coordination Centre (EADRCC) and Euro-Atlantic Disaster Response Unit (EADRU) in 1998 formalized NATO's role and responsibility in disaster assistance activities. The effectiveness of disaster assistance is highly dependent upon the degree of pre-planning and preparedness, the exchange of information and the ability to supply resources and services. Thus, the decision makers in command of controlling or managing a disaster relief mission need standard and interoperable procedures, guidelines, and regulations to respond quickly and effectively to an emergency situation.

Faced with transportation problems due to destroyed infrastructure and difficulty accessing affected areas, the logistic support, search and rescue activities, and transportation of first aid services is mainly provided by helicopters. Thus, a quick and efficient answer to multiple concurrent tasks requires a structured methodology for scheduling and tasking helicopters. As a result, the System Analysis and Studies (SAS) Panel approved this RTO Task Group in order to propose a framework for a generic and flexible decision support tool that can be used in effective management of helicopter missions both during humanitarian and military operations.

The scope of the effort consisted of conducting the problem analysis, investigating the concept of solutions and determining relevant technical requirements. In the problem analysis phase, the team described the problem areas, processes and functions, and carried out technology surveys. In the concept of solution phase, the team determined the sets of rules and policies, criteria, parameters, exogenous and endogenous input factors and defined the output of the decision support tool. In the technical requirement phase, the team developed and detailed all relevant technical requirements that may directly lead to the development of a computer-based decision tool.

The experience and the output of RTG SAS-045 clearly show that valuable expertise and know-how have been accumulated and the necessary infrastructure has been planned to develop the intended prototype decision support tool. The SAS-045 team strongly recommends that this work be extended to build a prototype, and implementable, system to be used during helicopter operations among NATO nations.

Outil informatique d'aide à la décision pour la planification des missions d'hélicoptères dans des opérations militaires et de secours en cas de catastrophe

(RTO-TR-SAS-045)

Synthèse

La création du Centre euro-atlantique de coordination des réactions en cas de catastrophe (EADRCC) et de l'Unité euro-atlantique de réaction en cas de catastrophe (EADRU) en 1998 a permis de formaliser le rôle et la responsabilité de l'OTAN dans des activités d'assistance en cas de catastrophe. L'efficacité de l'assistance en cas de catastrophe dépend en grande partie du niveau de planification préliminaire et de préparation, de l'échange d'information et de la capacité à fournir des ressources et services. Ainsi, les décisionnaires responsables du contrôle et de la gestion d'une mission de secours en cas de catastrophe ont besoin de réglementations, d'instructions et de procédures standard et interopérables afin de pouvoir réagir rapidement et efficacement à une situation d'urgence.

Face aux problèmes de transport liés à une infrastructure détruite et une difficulté d'accès aux zones touchées, le soutien logistique, la recherche et de sauvetage, ainsi que le transport des premiers secours sont principalement assurés par des hélicoptères. Une réponse rapide et efficace aux multiples tâches simultanées nécessite donc une méthodologie structurée pour la planification et l'attribution des missions aux hélicoptères. En conséquence, la commission Études et analyse des systèmes (SAS) a approuvé ce Groupe de travail RTO pour proposer un cadre structurel pour un outil générique et flexible d'aide à la décision pouvant être utilisé dans la gestion efficace des missions d'hélicoptères à la fois lors d'opérations humanitaires et militaires.

Le champ d'application de ce travail consistait à conduire l'analyse du problème, étudier le concept de solutions et déterminer les exigences techniques appropriées. Dans la phase d'analyse du problème, l'équipe a défini les parties, processus et fonctions du problème, et réalisé des enquêtes techniques. Dans la phase de concept de solutions, l'équipe a déterminé l'ensemble des règles et approches, critères, paramètres, facteurs de production exogènes et endogènes, et défini les résultats apportés par l'outil d'aide à la décision. Dans la phase des exigences techniques, l'équipe a développé et détaillé toutes les exigences techniques appropriées pouvant mener directement au développement d'un outil informatique d'aide à la décision.

L'expérience et les résultats du RTG SAS-045 ont clairement montré l'acquisition d'un savoir-faire et d'une précieuse expertise ; l'infrastructure nécessaire a été planifiée pour développer le prototype de l'outil d'aide à la décision prévu. L'équipe SAS-045 recommande fortement que ce travail soit étendu pour élaborer un prototype, et qu'un système puisse être mis en œuvre et utilisé lors d'opérations d'hélicoptères dans toutes les nations de l'OTAN.

Chapter 1 – INTRODUCTION

Recent disasters such as the Tsunami in Southeast Asia in December 2004 have shown how difficult it is to provide rapid response during disaster relief, which is highly dependent upon the efficiency of the existing communication and the coordination systems. Faced with the transportation problem due to the destroyed infrastructure and the difficulty to access the affected area, the logistic support, search and rescue activities and the transportation of first aid services is mainly provided by helicopters. Not only in disaster relief operations helicopters naturally have a very important role in fulfilling transportation and support and special tasks in joint and combined military operations. Thus a quick and efficient answer to multiple concurrent tasks requires a structured methodology for scheduling and tasking helicopters.

1.1 PURPOSE OF THE RESEARCH

Although NATO's involvement in international disaster assistance has a long history, the establishment of the Euro-Atlantic Disaster Response Coordination Centre (EADRCC) and Euro-Atlantic Disaster Response Unit (EADRU) in 1998 has formalized NATO's role and responsibility in disaster assistance activities. The EADRCC headed by the Director Civil Emergency Planning aims to coordinate the responses of the Euro-Atlantic Partnership Council (EAPC) countries to disasters occurring in EAPC area, to act as the focal point for information sharing among EAPC countries, and to maintain close liaison with the United Nations and the European Union as well as other organizations involved in international disaster response[1]. The effectiveness of disaster assistance is highly dependent upon the degree of pre-planning and preparedness, the exchange of information and the ability to supply resources and services. Thus, the decision makers in command of controlling or managing a disaster relief mission need standard and interoperable procedures, guidelines, and regulations to respond quickly and effectively to an emergency situation.

Furthermore, effective decision making is needed in both Article V and Non-Article V Operations at all command levels. Specifically, operational and tactical level planners need to generate practical and flexible plans for missions supported by helicopters during a crisis situation. In this respect, they should have rapid access to reliable information in a standard format and coordinate helicopter mission planning. Thus it is important to develop computer-based decision support tools (DSTs) which enhance the rapid and effective response capability of NATO commanders at operational and tactical levels.

The main goal of this research is to provide the basis for developing a generic and flexible decision support tool for effective management of helicopter missions by conducting the problem analysis, investigating the concept of solutions and determining relevant technical requirements.

The research will address the operational and tactical decisions concerning management of helicopters during disaster relief and military operations. It aims to recognize the fundamental capabilities needed by the Alliance and NATO nations and to recommend ways of improving or defining a formal structure for their decision making process. Helicopter mission management points to planning, re-planning, scheduling and maintaining control of various processes in operations involving helicopters. This tool is expected to be used as an operational tool.

[1] NATO's Role in Disaster Assistance, 2000.

1.2 BACKGROUND

At the November 1999 Meeting in Brussels, Turkey invited NATO nations and agencies to cooperate in a study which addresses planning and analytical issues relating to helicopter management in various emergency operations. The Panel decided to receive a detailed briefing at the May 2000 Meeting.

At the May 2000 Meeting in Lillehammer, the Studies Analysis and Simulation (SAS) Panel recommended Turkey to submit a formal proposal and Terms of Reference (TOR) for an exploratory team (ET) on helicopter logistics to be discussed at the November 2000 Meeting as it is mentioned on the decision sheet (AC/323(SAS)DS/5) dated 23 June 2000.

At the November 2000 Meeting in Brussels, the Panel decided to set the exploratory team ET.W as it is mentioned on the decision sheet (AC/323(SAS)DS/6) dated 18 December 2000.

The first meeting of ET.W was held in April 2001 in Brussels to prepare the first version of the TOR. The second meeting of ET.W was held in September 2001 in Istanbul to finalize the TOR, Programme of Work (POW) and Technical Activity Plan (TAP).

At November 2001 ET.W proposed to create the Research Task Group (RTG) SAS-045 "Computer-Aided Decision Support Tool for Mission Planning in Disaster Relief and Military Operation". This proposal has been welcomed by the SAS Panel and approved by Research and Technology Board (RTB) at the March 2002 Meeting.

Between November 2001 and March 2005, the RTG SAS-045 held seven meetings, respectively in Brussels, Cologne, Warsaw, Paris, The Hague, Ankara and Brussels, and completed the research on the proposed POW.

1.3 SCOPE

To achieve the objectives tasked, the group shared its work in three phases which constitute the modules of this document.

First, it is aimed to analyze the problem areas, processes and functions and to carry out technology surveys. Consequently the technology mapping and capability matrix will be developed using the identified current needs and capability gaps of existing decision support tools. Operational description of the problem is presented mainly in three aspects: operational context (environment, desired capability and scope), mission types (key mission tasks and functions) and decision-making framework.

Considerable effort was dedicated by the group to carry out three technology surveys on decision support tool technologies, information technologies, and geographical information systems, digital maps and mission data compilation technologies. This investigation was completed by preparing a fourth report on existing models, data and knowledge management repositories and planning process in NATO organizations.

In the concept of solution phase, the set of rules and policies, criteria, parameters, exogenous and endogenous input factors have been determined and the outputs of the decision support tool have been defined. Then, the required models have been developed and solution procedures and algorithms have been proposed to generate efficient and realistic plans. Thus, the concept of solution module presents the mathematical modeling description (i.e. the inputs, constraints, objective functions and outputs), the resolution method (mixed integer programming, heuristics), and computational results on testing scenarios.

The technical requirement phase of this study contains in detail all relevant technical requirements that may directly lead to the development of such a system. The links between existing NATO systems and the interfaces between NATO databases have been explicitly identified and the conformity with these has been sought. In the technical requirement module, information management and database interfacing module, protocols are set out. Database management systems are described and the information support tool dependencies upon other NATO infrastructures and information systems are defined.

The experience and the output of RTG SAS-045 clearly show that valuable expertise and know-how have been accumulated and necessary infrastructure has been planned so as to develop the intended prototype DST. In the last section, concluding remarks and recommendations to extend the activities of RTG SAS-045 in this direction are proposed.

Chapter 2 – THE OPERATIONAL CONTENT

2.1 BASIC DEFINITIONS

In order to develop common terminology and taxonomy, RTG SAS-045 has reviewed NATO glossary and STANAGs and studied NATO practices in the defined helicopter setting. Thus it has been decided that the following definitions would be adopted in the conduct of this research.

Helicopter missions to be covered in this research will focus on transportation functionality to be provided during military combat service support missions, peace support operations, humanitarian missions and disaster relief operations. Combat service support missions are defined in AC/243 TP/8[1].

Disaster is an unforeseen and often sudden event that causes great damage, destruction and human suffering. Though often caused by nature, disasters can have human origins. Disaster causes can be earthquake, hurricane, tornado, volcanic eruptions, fire, flood, blizzard, drought, chemical spill, nuclear accident, etc. Wars, civil disturbances or civil wars that destroy homelands and displace people are also included among the causes of disasters.

Disaster relief operations are conducted to relieve human sufferings, especially in circumstances where responsible authorities in the affected area are unable or unwilling to provide adequate service support to the population. It may be necessary to execute these in the context of a peace support, or as a completely independent mission. Disaster relief operations are included within the class of humanitarian missions. They are conducted in emergency situations to prevent loss of life and property providing emergency relief to victims of natural or man made disasters in response to host government request for immediate help in both non-hostile and conflict containing environment.

Although various decisions are made by the decision makers faced with such an issue, this research is limited to proposing alternative solutions for operational and tactical decisions as defined below.

Operational management is concerned with the operational decisions and tasking of a particular mission. The main aim is to allocate helicopter capacity and to identify the fleet and crew composition for a given mission assuming that the deployment plan of participating nations is known. The allocation and transportation of logistic assets and supplies (for example, transport resources, vehicles, weapons, ammunition, fuel supplies, stocks of spare parts, medical material, non-perishable food and shelter) to the operation sites is determined. Moreover, the restrictions prevailing in the air space control and management are taken into account in the process of determining the sorties for each helicopter.

Tactical management addresses the planning and re-planning of detailed flight routing, re-fuelling and transportation plans to execute the mission. Here, detailed routing plans are developed by taking into account helicopter technical specifications (mobility, flexible payload, vulnerability, life cycle cost, etc.), environmental and atmospheric conditions as well as day/night-time considerations and threat levels. The dynamic crew scheduling to helicopters is also incorporated into the mathematical models so that routing and crew scheduling decisions will be made concurrently in order to improve the quality of the solutions.

[1] Technical Proceedings of NATO Helicopter Symposium, December 1995, The Hague, The Netherlands.

The following assumptions that determine the operational context, environment, desired capability, and scope are taken into account in the development of the proposed DST.

a) An emergency event has occurred in a known area.

b) NATO has realized the mission and decided to respond to the mission, given the basic information:

  - Number of operational bases is known by the decision-makers.
  - Reliable information is available about helicopters, crew, logistics (facilities, fuel, etc.) (Nations declared these numbers to NATO during the Defence Planning Cycle with Defence Planning Questionnaires) (Time = 0).
  - Some embarkation and debarkation nodes for traffic control, maintenance, refuelling, etc., are known at time 0 and there might be additional nodes that might arise at time t. The set of nodes can be updated dynamically and the database associated with a node identifies its type and provides the required information about it.

c) NATO planners estimate the real time requirements based upon the above data.

d) NATO may request more helicopters from nations if it is needed. Whenever there is a change in the availability of helicopters, it will be updated by the operator using the DST.

e) The requesting body within NATO will generate the needed NATO Request for Air Transport Format (NARAT).

f) The Air Transport approving authority as the decision maker will receive transport request in the form of a NARAT. The data of the NARAT will be the input for the DST. A suitable solution is calculated by the DST and allows the decision maker to choose tasks to be executed from a complete list. Thus at the time of replanning, the DM will define the current status of already defined tasks and will choose new tasks and define the workload involved in each task.

g) The DST will generate a solution for a certain time period. (The estimated remaining duration of the mission on a daily basis) and display it in the form of TRANSAR (Transportation for Airlift Response). Whenever there is a change in anyone of the system parameters, the replanning will be done on a regenerative basis with the updated databases.

Both disaster relief and military operations pose quick response situations where time is the key limiting factor and helicopter moves should be planned and conducted very rapidly. Thus, the speed in generating good solutions on a quasi real time basis affects the overall performance of helicopter operations. Since the speed of generating solutions depends upon accessibility to reliable information, a computer-based DST requires the integration of physically distributed databases available in NATO, and interoperability is needed to meet the standardization issue in such a context.

2.2 MILITARY AND HUMANITARIAN MISSION TYPES, TASKS AND FUNCTIONS

A three level taxonomy is used for missions. At the top level, there are different missions which may have different tasks and tasks may include certain functions.

a) Helicopter Mission Types:

  1) Military Combat Service Support Mission;

2) Peace Support Operations;
3) Humanitarian Missions; and
4) Disaster Relief Operations (Natural Disasters, Man Made Disasters, Technological Disasters, Sabotage, Man Made Accidents, Terrorism).

b) Helicopter Tasks:

5) Transportation Tasks:
    - Transportation of human beings, equipment and material;
    - Evacuation of sick, wounded and irradiated people; and
    - Evacuation of possessions and live stocks.

6) Support Tasks:
    - Area reconnaissance (e.g. area photography);
    - Identification of radioactive contamination;
    - Engineering reconnaissance;
    - Protection of command and control systems;
    - Area security; and
    - Observation to detect and report suspicious movement.

7) Special Tasks:
    - Assembling;
    - Launching of floating devices;
    - Mining and mine clearing;
    - Search and rescue from disasters; and
    - Command and control.

NATO decision makers, crisis management centers, and military forces are the key users of DST. There might be multiple decision makers from different organizations within the scope of a given mission. One can classify the crisis situations according to the taxonomy given in Figure 2.1 and Figure 2.2.

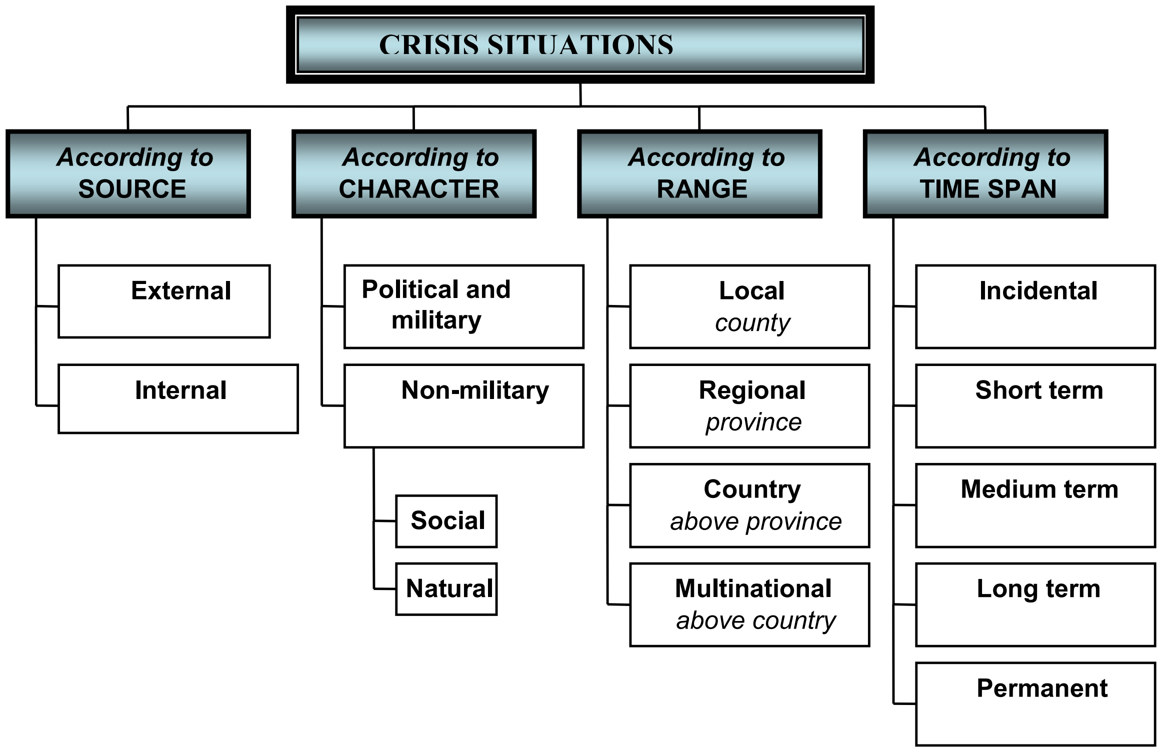


Figure 2.1: Taxonomy for Crisis Situations – 1.

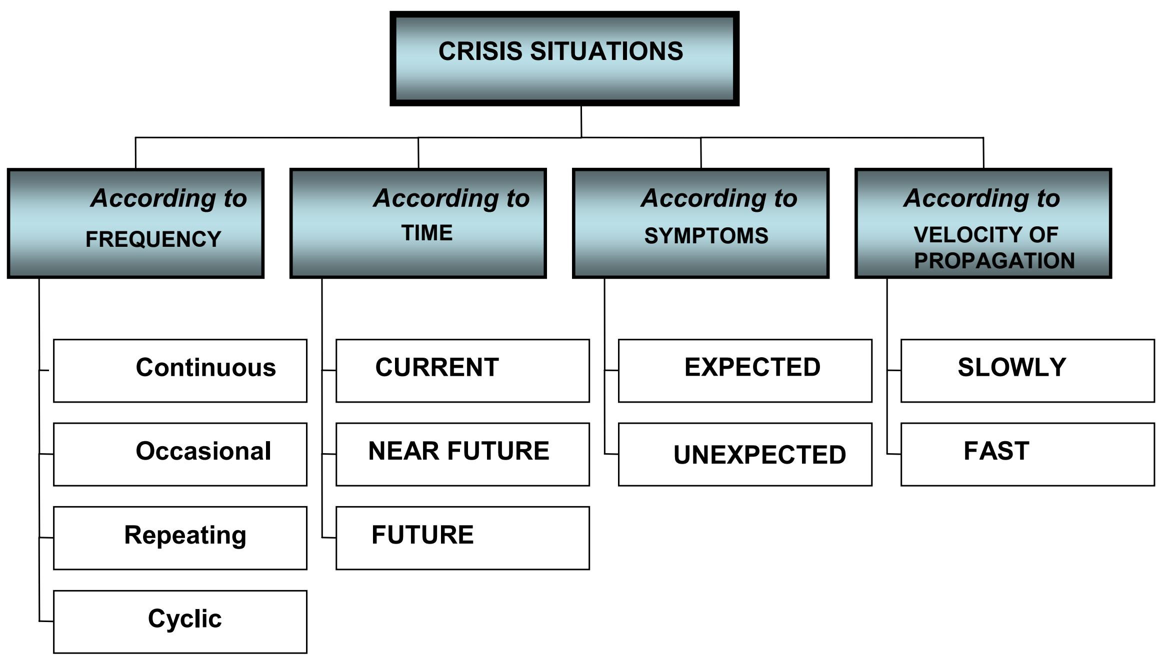


Figure 2.2: Taxonomy for Crisis Situations – 2.

2.3 DECISION-MAKING FRAMEWORKS

Two different decision making frameworks are defined for military operations and disaster relief management. Hierarchical levels, organizational structure and control over aviation units are taken into account, while developing the framework for military operations depicted in Figure 2.3.

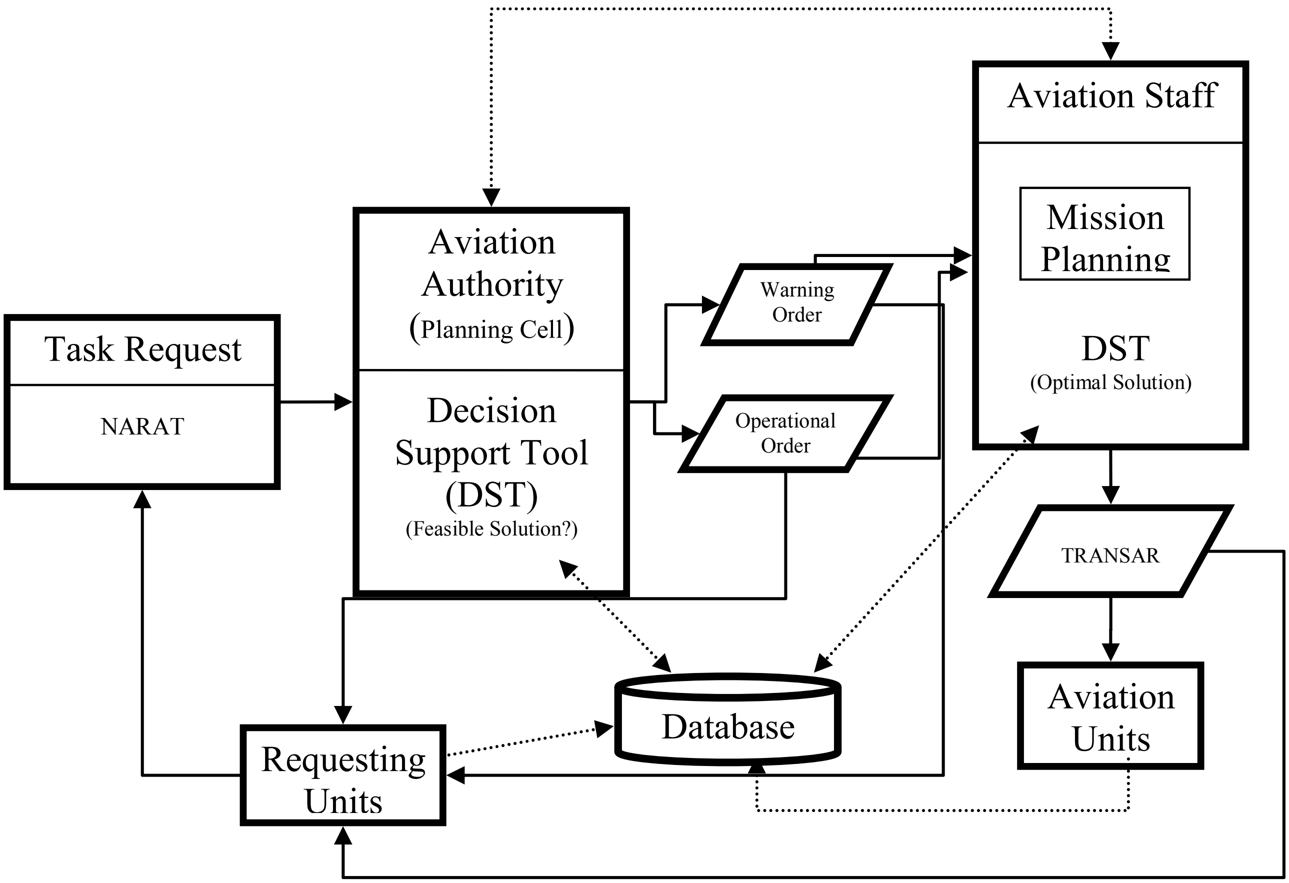


Figure 2.3: Decision-Making Framework for Military Operations.

To carry out the planning activity needed for the case described in Section 2.1, a common database is to be populated with adequate data from different sources. The aviation authority and aviation staff practice the full control of the database in order to keep it up to date. Air transport requesting and aviation units are expected to provide relevant data in order to facilitate the update of database. A unit is not permitted to edit data in database. The following information in databases is to be updated as needed on a real time basis:

a) Helicopters;

b) Crews;

c) Nodes;

d) Tasks (priority, urgency time window, reaction time);

e) Environmental Conditions:

- Air space control; and
- Weather conditions.

f) Other Resources.

Crisis management center (CMC) is to control all available helicopters for effective and timely response. Thus, CMC directly assigns both civilian and military aviation units to the tasks according to our proposed decision-making framework for crisis management. The framework is shown in Figure 2.4. Coordination between military and civilian aviation units is performed by CMC, which controls the database and aviation units provide updated date for central database.

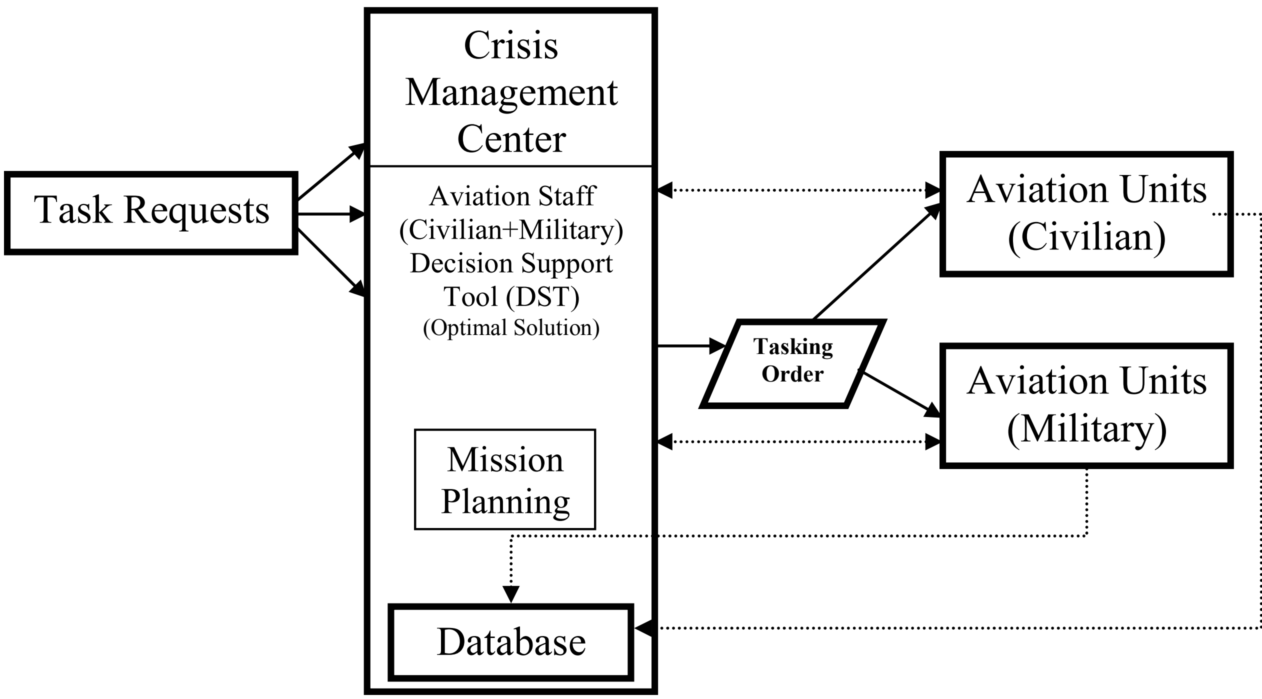


Figure 2.4: Decision-Making Framework for Crisis Management.

Mission planning is done for a certain period of time as long as no high priority task request received. When a high priority task is received or planning period is up, a replanning of missions is carried out for the tasks with no tasking order. If a tasking order is issued for a task, the task is assumed to be performed. Replanning in military operations is carried out by the aviation staff according to the operational orders issued by the aviation authority.

ATP-53(A) describes the responsibilities and procedures for implementing the policies for air transport coordination and cooperation between NATO and nations. Airlift elements and channels for communication in NATO and request-tasking channels for NATO are shown in Figure 2.5 and Figure 2.6. ACE Mobility Coordination Center (AMCC) may function as the aviation authority and JMCC and CECC may function as the aviation staff in proposed military decision making framework.

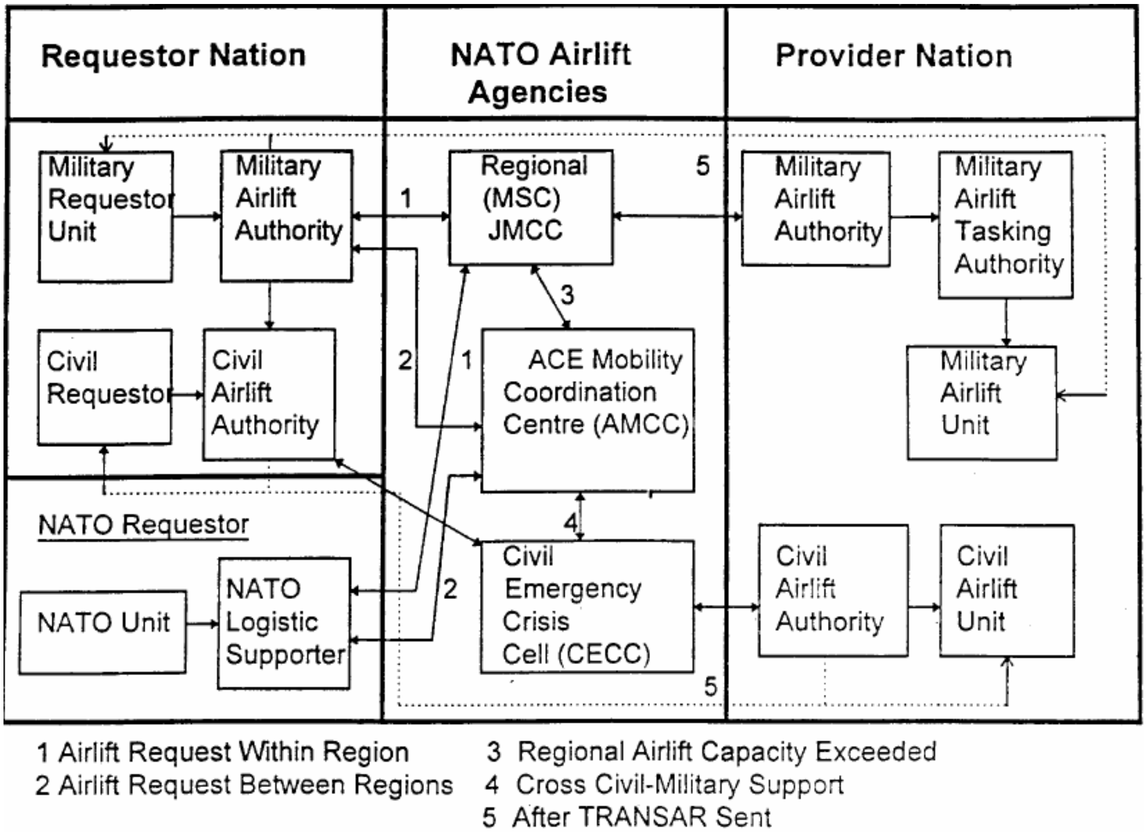
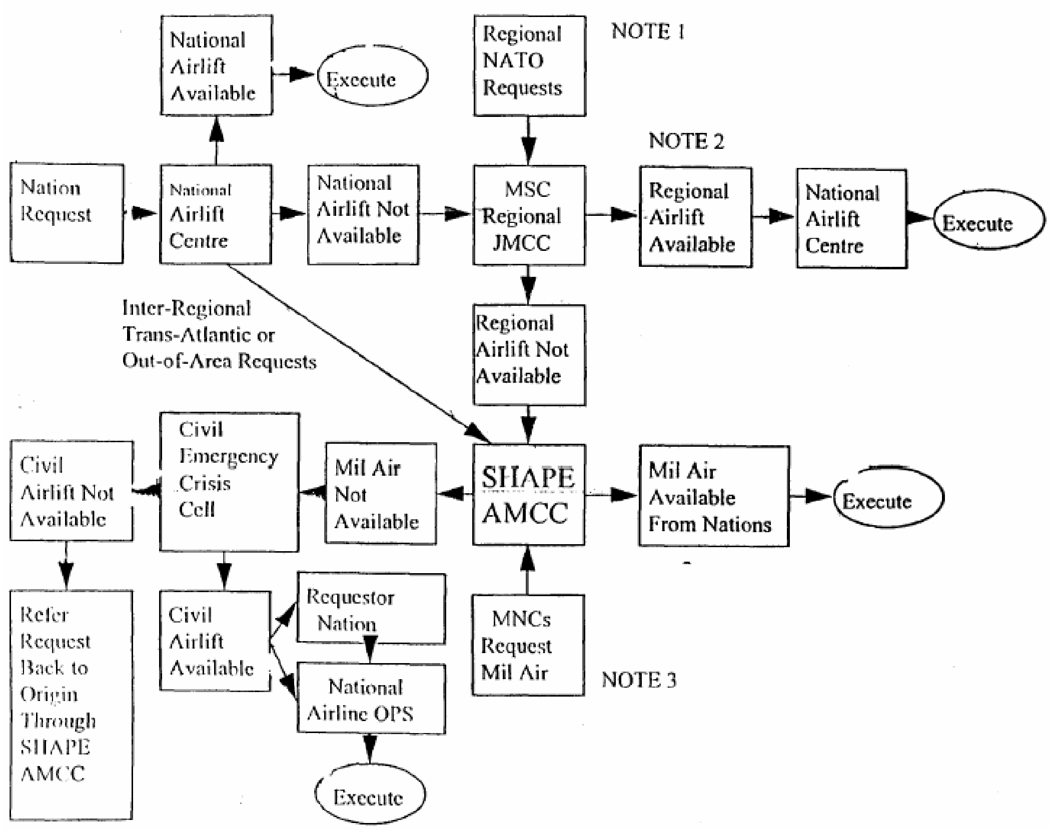


Figure 2.5: Airlift Elements and Channels for Com. in NATO (Source:ATP-53(A)).



Note 1: NATO request to be submitted when national airlift not available /appropriate. NATO agencies such as NAMSA and NATO forces such as NAEW assigned or operating within a region will use this channel.
Note 2: MSCs will consider U.S. airlift forces in their region and made available to the alliance by the U.S. CINCEUR as regional resources, and will request support directly from the regional ARC. NEC requests will be routed to the U.S. ARC.
Note 3: SACEUR may generate airlift requirements and as NATO Executive Agent for airlift will receive airlift requests directly from supported MNCs (SACLANT).

Figure 2.6: Request – Tasking Channels for NATO (Source: ATP-53(A)).

Chapter 3 – TECHNOLOGY SURVEYS

The proposed decision support tool has been designed in a distributed environment in order to facilitate hierarchical command and control and the integration of different decision models corresponding to procedural terms and information processing forms the crux of the research.

Uncertainty which arises from the existence of incomplete and imprecise information inherent in both disaster relief and military operations is another challenging technical aspect that needs to be studied in detail. Neither the task, the duration, the nature nor the multi-national context in which the operation will take place are known in advance: these factors will differ for each mission. Therefore, a versatile approach compatible with international standards should be analysed under different scenarios which encompass this uncertainty.

To achieve these purposes and optimisation requirements of the model, during the Analysis Phase of the project, technology surveys were carried out on modelling, computer and software engineering and data collection technologies; geographical information systems, digital maps, mission data compilation systems; model, data, and knowledge management repositories in NATO nations. This process was identified by the Program of Work (POW). Then, the technology mapping and capability matrix has to be developed using the identified current needs and capability gaps. This chapter presents a comprehensive summary of these investigations.

3.1 DECISION SUPPORT TOOL TECHNOLOGIES

Before going into detail in the conception of a mathematical tool, the group decided to make a survey of the mathematical techniques available for the solution of the model.

It is figured out that many decision problems can be modelled using mathematical programs, which seek to maximize or minimize some objective which is a function of the decisions. The possible decisions are constrained by limits in resources, minimum requirements, etc.

The survey includes a detailed analysis of linear, non-linear, stochastic, constraint satisfaction problem which can be used for the design of decision support model for helicopter mission planning in disaster relief and military operations. Furthermore, it includes simulation, data analysis and mining and artificial intelligence.

Artificial intelligence has specially retained the attention of the group, since many Artificial Intelligence problems are not well defined initially. The process of developing and testing software helps to clarify the problem. Artificial Intelligence problems often require different kinds of knowledge to be represented and different kinds of inference mechanisms to be used and intelligent systems may need to modify themselves dynamically over time.

Deeper researches were engaged, and the techniques of Artificial Intelligence were subject of examination: Expert Systems (ES), Neural Networks, Fuzzy Logic and Genetic Algorithms.

For each of these mathematical techniques of optimisation, simulation, data analysis and mining and artificial intelligence, the group has investigated the related software systems available in the market.

3.2 INFORMATION TECHNOLOGIES

An information system, with its very broad definition is a –computer– system that is used for the storage and retrieval of any type of information – text, numerical, graphical, video, and sound. Before designing an information system, an analysis is to be done to find out the requirements. For decision support systems, a mathematical model should also be considered in the analysis phase. Then, in the design phase, focus is directed towards the realization of the system to meet these requirements.

The group accomplished a significant research on data management issues to be considered for RTG SAS-045 on "Computer Based Decision Support Tool for Helicopter Mission Planning in Disaster Relief and Military Operations." The topics covered include data management approaches, data interaction strategies, Database Management System (DBMS) paradigms and DBMS providers.

This research concludes that the following criteria may be the basis for designing data management architecture for RTG SAS-045:

Performance and Accessibility: Data management should address the time requirements of the decision support algorithm to promote its robustness. Thus the desired system proposed by RTG SAS-045 should respond quickly to an emergency situation.

Interoperability with other NATO models and repositories: Based on another technology survey within analysis phase of RTG SAS-045 [SMI03], most common data repositories in NATO organizations including Integrated Command and Control & Air Command and Control System (ICC and ACC) and Allied Deployment and Movement System (ADAMS), Tool for Operational Planning, Force Activation and Simulation (TOPFAS) have been implemented by using relational database models. In order to generate practical and flexible plans for missions supported by helicopters during a crisis situation, the decision support system should have rapid access to reliable information in a standard format. The system should also be able to distribute its results in a standard format to the related users.

Distribution and Crash Recovery: In disaster-relief operations it is probable that some parts of an information network might be damaged or inaccessible. The data management architecture to be proposed should consider data distribution and crash recovery by planning redundancy and back up system.

3.3 GEOGRAPHICAL INFORMATION SYSTEMS, DIGITAL MAPS AND MISSION DATA COMPILATION TECHNOLOGIES

A Geographical Information System (GIS) is a "computer system for capturing, managing, integrating, manipulating, analyzing and displaying data which is spatially referenced to the Earth." (McDonnell and Kemp, 1995). GIS is an information system that is composed of hardware, software, data and personnel. The current generation of GIS products support data integration and manipulation, query, thematic and overlay analysis, and visualization in two and three dimensions. Geographic information technology has emerged as an important operations research tool for exploiting increasingly available amounts of valuable geographic data in a form easily comprehended by analysts and decision makers (Hanigan, 1989).

What distinguishes GIS from other forms of information systems, such as databases and spreadsheets, is that GIS deals with spatial information. GIS has the capability to relate layers of data for the same points in space, combining, analyzing and, finally, mapping out the results. Spatial information uses location, within a coordinate system, as its reference base. The most common representation of spatial information is a map on which the location of any point could be given using latitude and longitude, or other grid references.

An extensive work on digital map format and co-ordinate reference system was performed. The applicable STANAGs were considered, and automatic data capture, remote sensing and tactical data link systems have also been investigated.

The group stated that a generic Decision Support System (DSS) should interact with the analyst and the decision maker for exploiting different course of actions and present the current status and the results of the actions in a form easily comprehended by analysts and decision makers. Geographic information systems and mission data compilation systems can help to fulfill this requirement.

3.4 MODELS, DATA AND KNOWLEDGE MANAGEMENT REPOSITORIES AND PLANNING PROCESS IN NATO ORGANISATIONS

Since several models are known to exist to support an operation inside NATO, the group has investigated to perceive if the possible application of the model in support of the helicopter planning that will be developed within this study could be achieved successfully in NATO nations.

The group observes that NATO nations developed several models for planning purposes, but none of them has specifically been developed to support helicopter operations. However, it seems to be sensible that the helicopter model that will be developed within RTG SAS-045 scope will need to use similar software platforms and compatible database systems used in NATO models like ADAMS.

The group concludes that some planning models have been developed within the nations; however, most of the models are developed to support the planning of aircraft. So far only The Netherlands have developed a model called FELPATH to support the operational planning of helicopters. FELPATH, however, supports only a limited number of the functionalities that are required for the new NATO helicopter planning model. On the other hand it was rational to use the lessons learnt from The Netherlands, while developing the model, and probably use FELPATH as a starting point for developing the decision support tool for planning helicopter operations.

Chapter 4 – DATABASE MANAGEMENT ARCHITECTURE

As it is explained in Chapter 2, in case of a crisis realized by NATO, the NATO requesting body will summon a multi-national fleet based upon the impact of the crisis, and access to all relevant databases will be created to align with the resources provided by NATO nations. Using this information, the NARATs will be generated and delivered to the air transport authority as task requests. The NARATs will be implicitly connected to all helicopter related data files available in NATO systems. The helicopter tasking and routing will be done by the DST and the databases will be updated by considering the current status of the crisis situation. A conceptual architecture is depicted in Figure 4.1.

At the stage of designing the data model of the DST, capabilities needed for operational reasons are first reviewed and summarized as follows:

- Robustness;
- Speed (Quasi real-time performance);
- Accessibility to accurate data on a real-time basis;
- Standardization;
- Inter- operability;
- Flexibility;
- Re-planning possibility;
- Capability to deal with randomness, fuzziness and incompleteness;
- What-if analysis capability;
- Alternative solutions with respect to different objective functions under various assumptions;
- Provision of a realistic applicable solution; and
- Prioritisation of capabilities.

Based upon these design requirements, the catalogue, mission and task databases are developed in accordance with the general flow depicted in Figure 4.2.

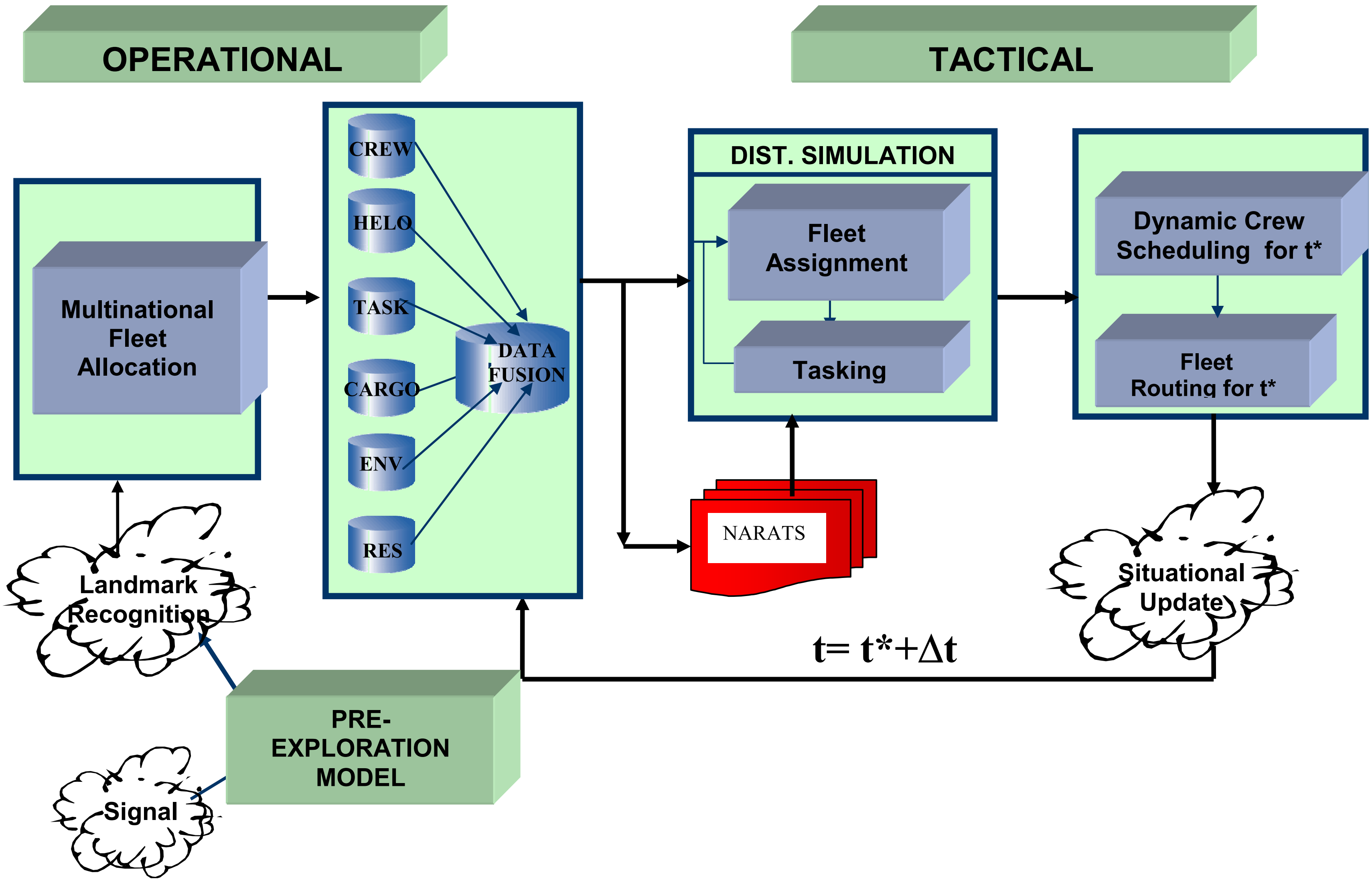


Figure 4.1: A Conceptual Architecture of the DST.

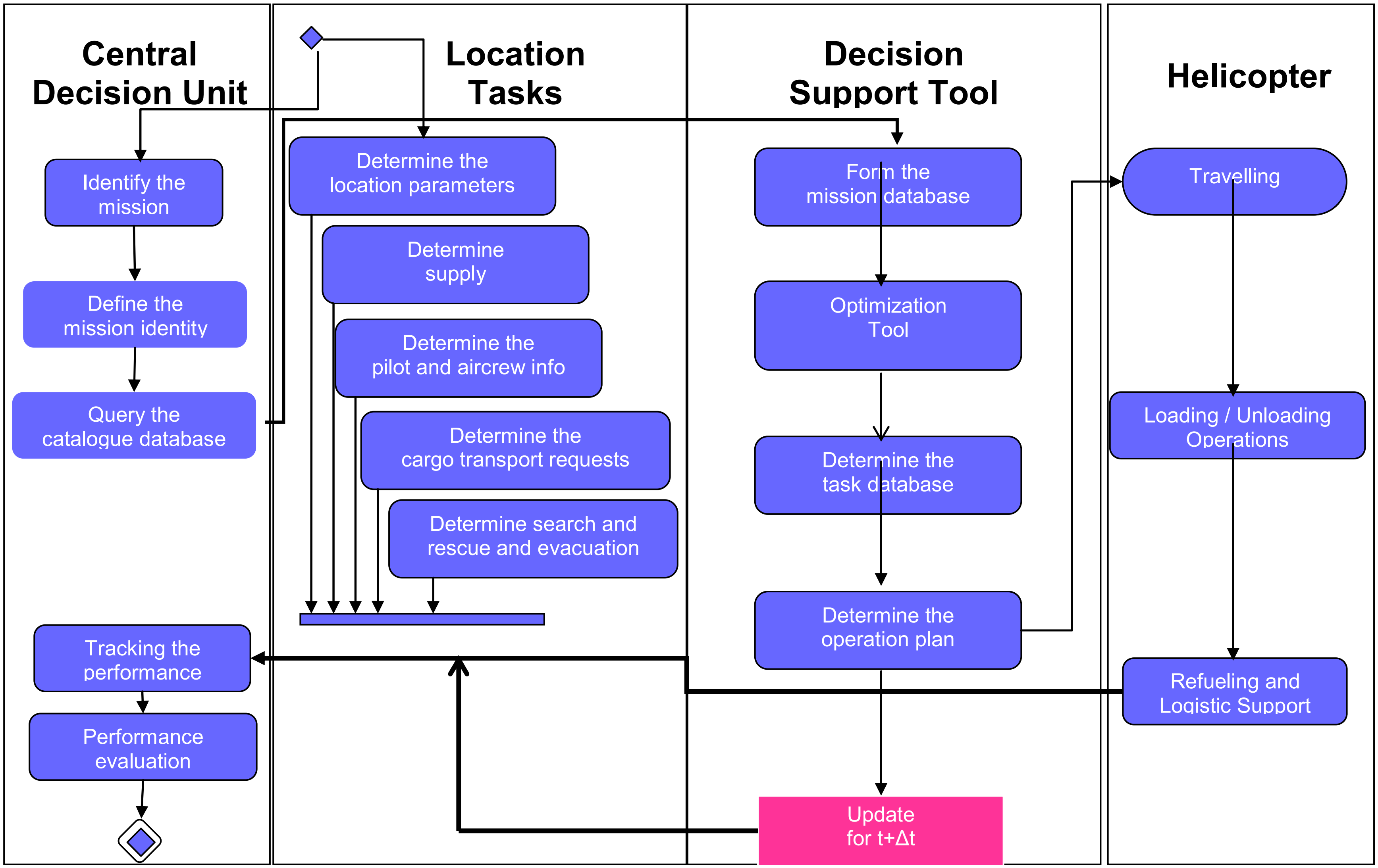


Figure 4.2: General Flow of the DST.

4.1 PRINCIPLES OF DATA MODELING

At this point, it is attempted to design the database management system of the intended decision support system by employing relational data modeling approach. Relational data model organizes and represents data in the form of tables or relations. (Hansen and Hansen, 1996) The term relation refers to a two-dimensional table of data. In other words, according to the model, information is arranged in columns and rows. The term relation, rather than matrix is used here because data values in the table are not homogeneous. A relation has a unique name and represents a particular entity. Each row of a relation, referred to as a tuple, is a collection of facts about a particular individual of that entity. In other words, a tuple represents an instance of the entity represented by the relation; that is, the tuples are the rows of the relation. The columns of a relation hold values of attributes that are associated with each entity instance. (Maurer et al., 1998) Each column in the relation is an attribute of the relation, and the set of all possible values that an attribute may have is the domain of the attribute. (Hansen and Hansen, 1996)

The specification of a relation schema must include the declaration of at least one key. A key is a set of attributes whose values uniquely determine a single entity. For a relational table, each row must have a distinct value for its set of key attributes. (Riccardi, 2001) While a relation may have two or more candidate keys, one must be selected and designated as the primary key in the database schema. When the primary key values of one relation appear in other relations, they are termed foreign keys, which may duplicate occurrences in a relation, while primary keys may not. (Maurer et al., 1998)

There are three basic integrity constraints in database design. The Entity Integrity is summed such that no key attribute of any row in a relation may have a null value. However, the Referential Integrity requires that either every foreign key must be null, or its value must be the actual value of a key in another relation. The third constraint is Functional Dependencies. This constraint provides a means for defining additional constraints on a relational schema. The essential idea is that a tuple's value in one attribute uniquely determines the tuple's value in another attribute. (Hansen and Hansen, 1996)

The relationships are transformed into relational model in three different ways depending on the relationship's cardinality. In One-to-Many relationship types, the relation describing the object on the many side of the relationship receives the foreign key column that points to the other object. However, in transforming Many-to-Many relationships, an intersection relation is created between two tables. The intersection relation is also a table, which includes the key fields of both related tables. (Hansen and Hansen, 1996) The third type relationship is One-to-One relationships. In transforming this type, both tables must include the other's key. (Riccardi, 2001)

4.2 THE STRUCTURE OF THE DATABASE MANAGEMENT SYSTEM

The database developed in this study for the decision support system which contains three main parts is depicted in Figure 4.3: Catalogue Database, Mission Database, and Task Database. In the optimization process, the main flow starts from the Catalogue Database, which is updated by central NATO units, and Catalogue Database is queried to form the Mission Database. Mission Database is updated by central NATO units and domestic units. Optimization tool takes the Mission Database and carries out the task of determining the assignments. Task database is the result of the optimization process and it is queried to build the task assignment reports.

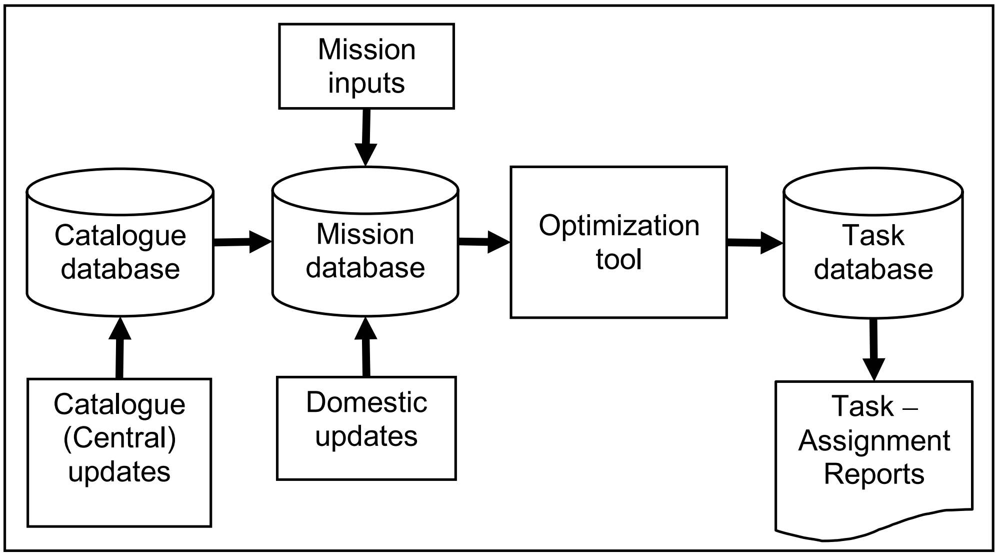


Figure 4.3: The Information Flow in the Database Management System.

4.2.1 Catalogue Database

Catalogue Database is more static relative to the other databases. This database includes invariable data about specifications of helicopter types, location properties of operation bases and properties of cargo types. It is taken as a "Catalogue" in the main flow as being the vital input of the Mission database. Catalogue Database is only updated by the central NATO unit. The capacity, flight characteristics of helicopters, the properties of operation bases are entered in this level by central units.

4.2.2 Mission Database

Mission Database includes mission specific information in addition to the catalogue database. Characteristics of the mission, the availability information about helicopters, pilots, and aircrew, material, and supply amounts of commodities at different locations are the main fields in this database.

Mission Database is updated by both central NATO units and local decision-making units (DMUs). The general mission orders and characteristics are supplied by NATO. DMUs at demand locations provide the inputs about their requirements, like types and amounts of material demand, extent of human evacuation, etc. Moreover, operation bases provide information about their supply capacity, like material supply amount and medical service availability.

4.2.3 Task Database

Task Database is the output of the Optimization Tool. As the result of optimization, the task assignments are obtained. These assignments cover the routes of helicopters; material and human load/unload locations and amounts; and refueling points and amounts. In addition, Task Database is employed to form Task – Assignment Reports, which are mainly the assignment orders for the operations.

4.3 MODEL A1: MISSION-BASED RELATIONAL DATA MODEL

4.3.1 Relations in the Model

The whole database structure is modeled by relational database modeling approach. In this approach, tables of the database are linked via the key fields. The model is given in Figure 4.4 with only the titles of the tables. In order to study the relations in the database in a descriptive manner, the relations are grouped into six parts that are given below:

- Participation of Forces to Missions;
- Disposition Relations;
- Helicopter and Aircrew Availability;
- Locations Visits;
- Bin Packing of Helicopters; and
- System Parameters.

4.3.2 Tables for "Participation of Forces to Missions"

- Statement of Requirement;
- Force;
- Mission; and
- Task.

In the first part, participations of forces to missions are illustrated in Figure 4.5. "Statement of Requirement" table describes the details of the plan. Since different nations may contribute to the same mission, "Force" table summarizes the contribution of nationalities to the missions. Obviously, there may be more than one task in a mission and "Mission" and "Task" tables hold information about the zone and time data, respectively. The task database includes detailed information like cost, status, and priority, as well.

4.3.3 Tables for "Disposition Relation"

- Force;
- Material;
- Material Category;
- Pax;
- Pax Category;
- Equipment;
- Equipment Category; and
- Location.

Disposition relations include information about material, equipment, and pax supply for each national force participating in the mission. (Figure 4.6.) These three of types have catalogue information that inherits form their categories. These are linked to the location table because they are supplied and demanded by locations. In the location table, there are fields including the coordinates, flight and landing properties for helicopters.

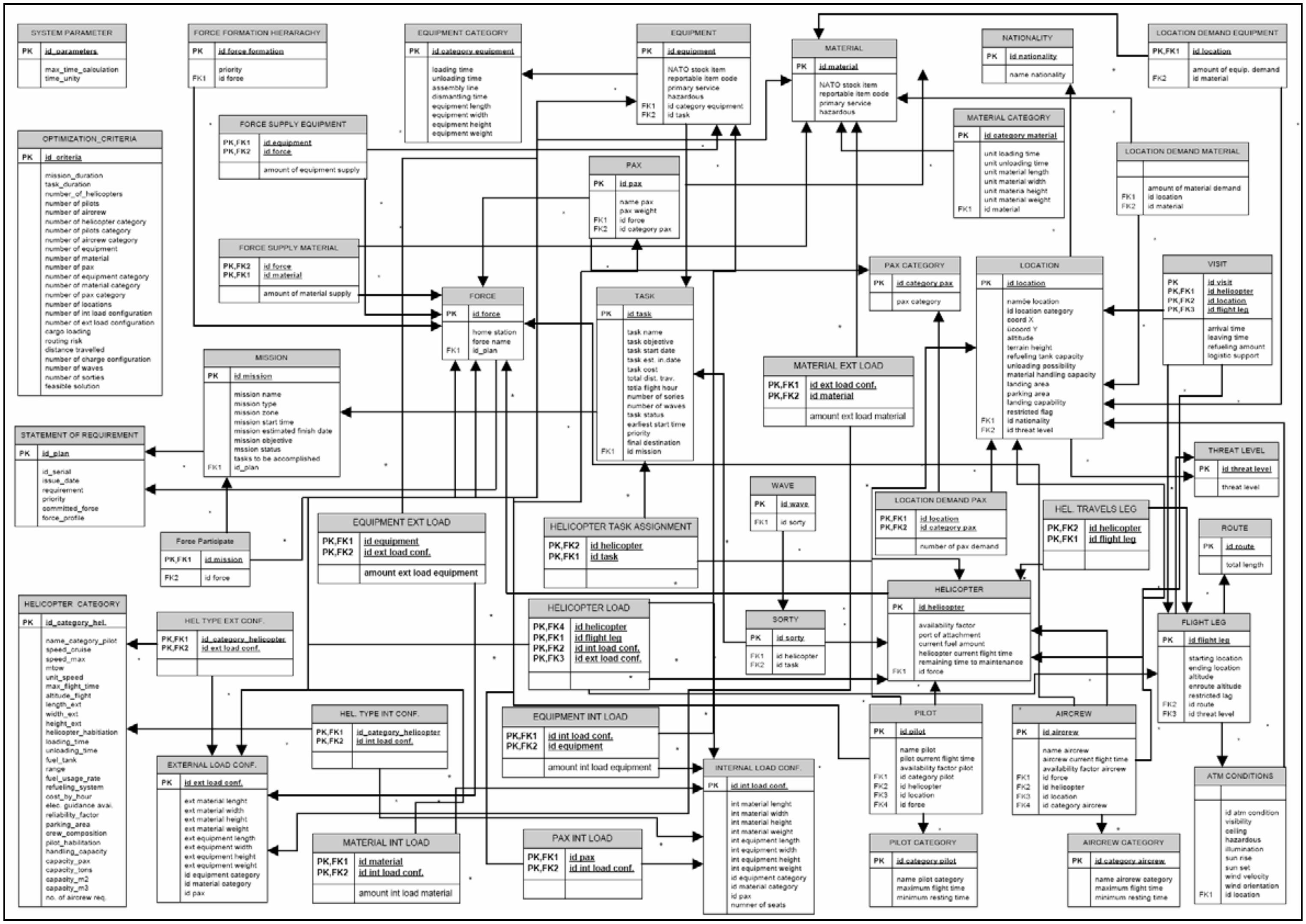


Figure 4.4: Relational Data Model.

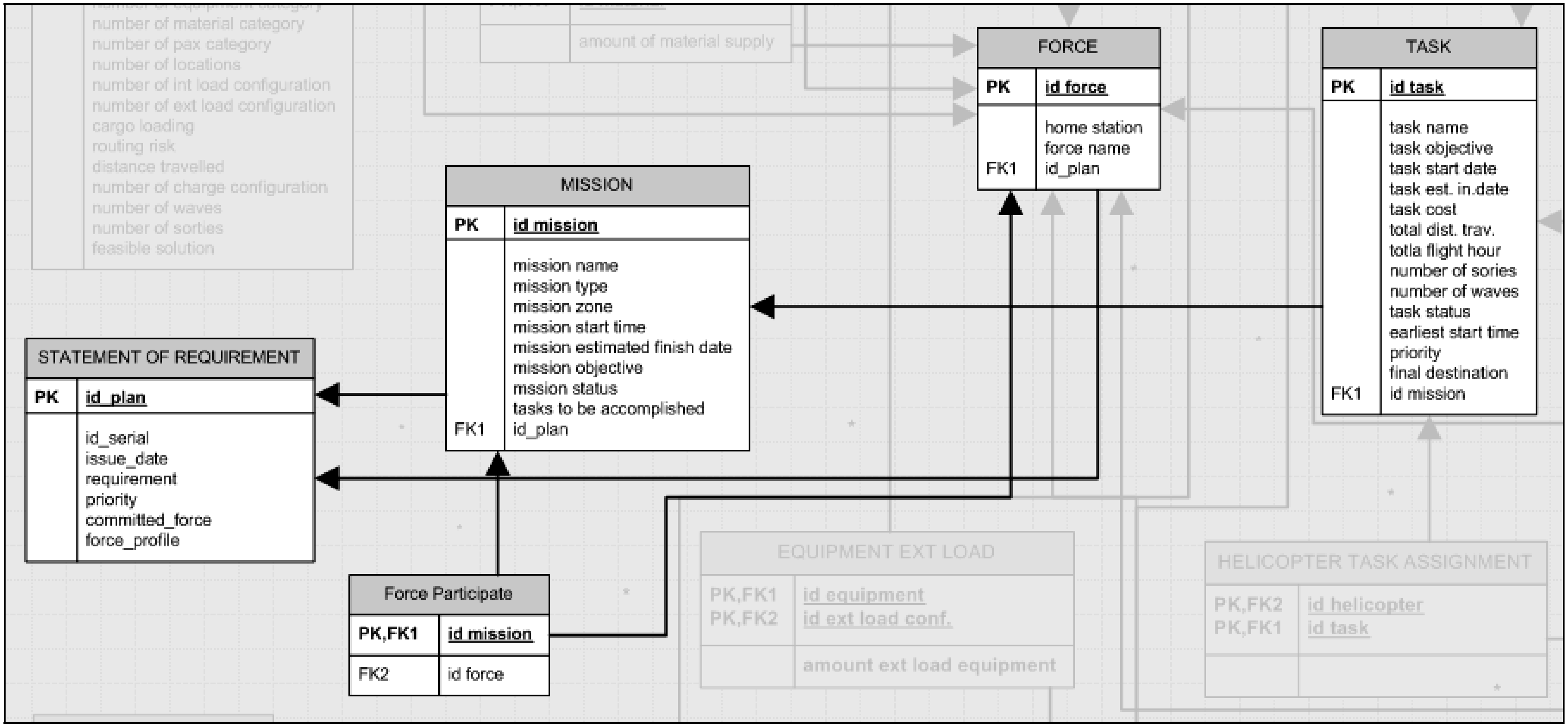


Figure 4.5: Participation of Forces to Mission.

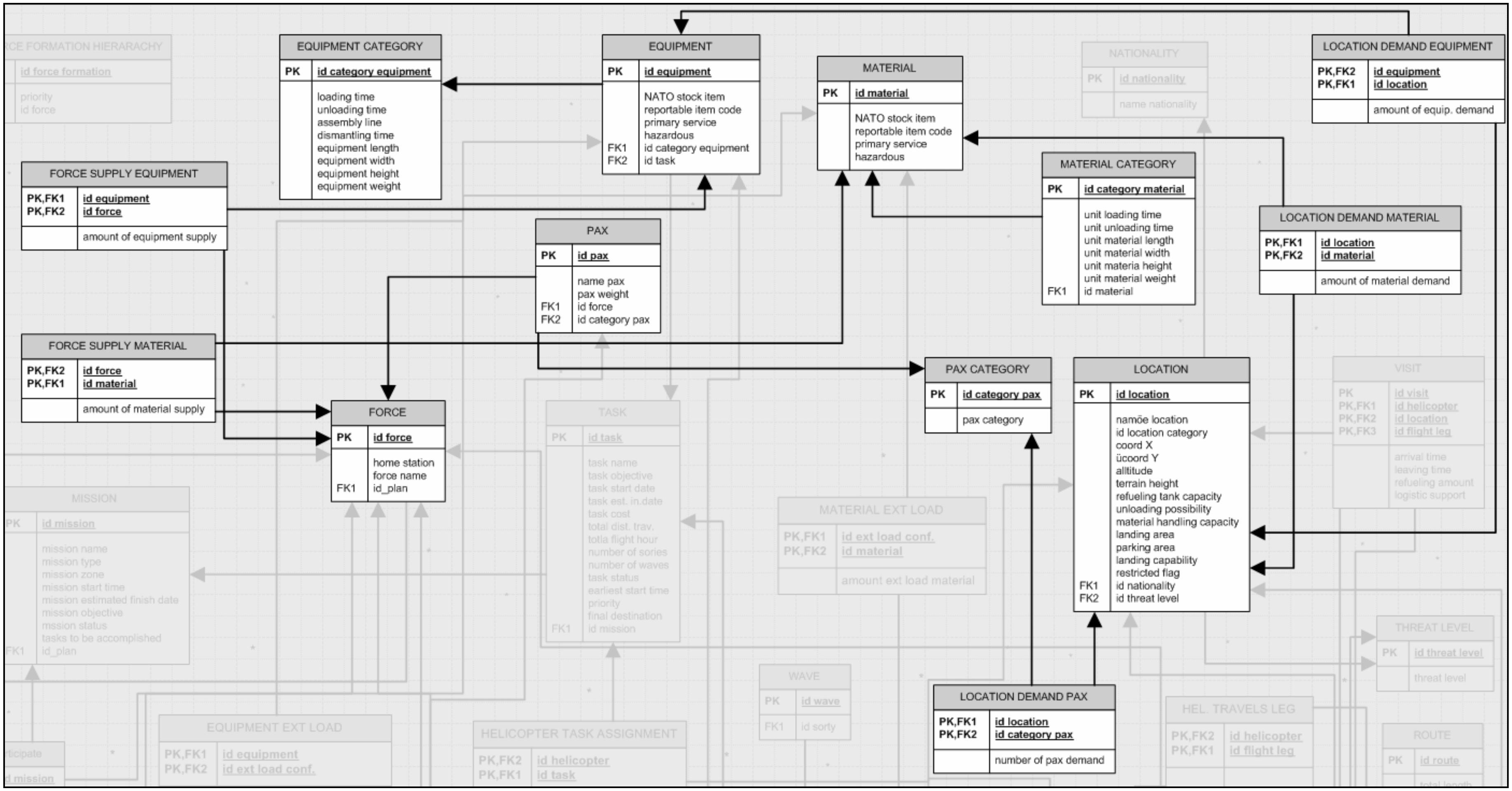


Figure 4.6: Disposition Relations.

4.3.4 Tables for "Helicopter and Aircrew Supply"

- Force;
- Aircrew;
- Aircrew Category;
- Pilot;
- Pilot Category;
- Helicopter; and
- Helicopter Category.

The participating forces supplies not only material, but also more importantly helicopter, pilot, and aircrew for the mission. (Figure 4.7.) The catalogue information is also defined for aircrew and pilots. In these category tables, there are fields about the flight limitations of aircrew and pilots that will be assigned to helicopters individually. Helicopter table has both very detailed categorical information, and individual information for helicopters.

4.3.5 Tables for "Location Visits"

- Helicopter;
- Location;
- Route;
- Flight Leg;
- Threat Level;
- Atmosphere (ATM) Conditions;
- Visit Relation;
- Task;
- Sorty; and
- Wave.

This part of the database covers the visits of the helicopters to the locations. (Figure 4.8.) Visit Relation is also demonstrated as a table, which shows the links between "Helicopter, Location, and Flight Leg" tables. The routes of the helicopters are divided into flight legs, which consist of two locations, one for departure, and one for arrival.

There are four key fields in the "Visit" table. According to these, this relationship can be summarized as follows: "A helicopter visits a location as an arrival node of a flight leg." In this visit, it should be indicated if refueling and logistic support will be provided. In addition, the arrival and departure times are stored in this table.

As the limiting variables to the visits of helicopters, ATM conditions and threat levels are as included in the relational database model. These tables are both linked to "Location and Flight Leg" tables. The relation between the "Task" table and the "Sorty" table represents the assignment of sorties to tasks. Tasks form "Wave"s. Finally, Helicopters are assigned to sorties.

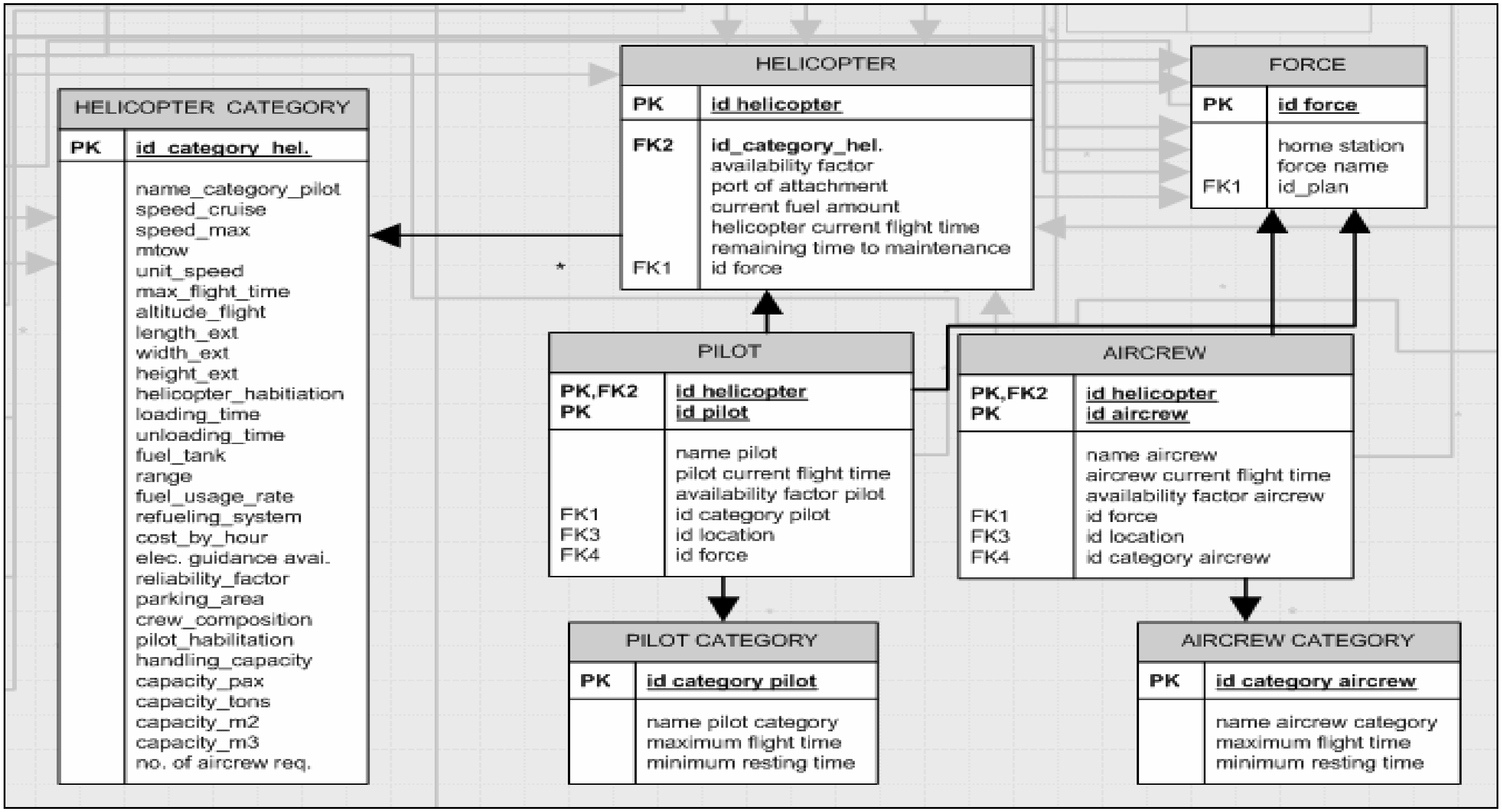


Figure 4.7: Helicopter and Aircrew Supply.

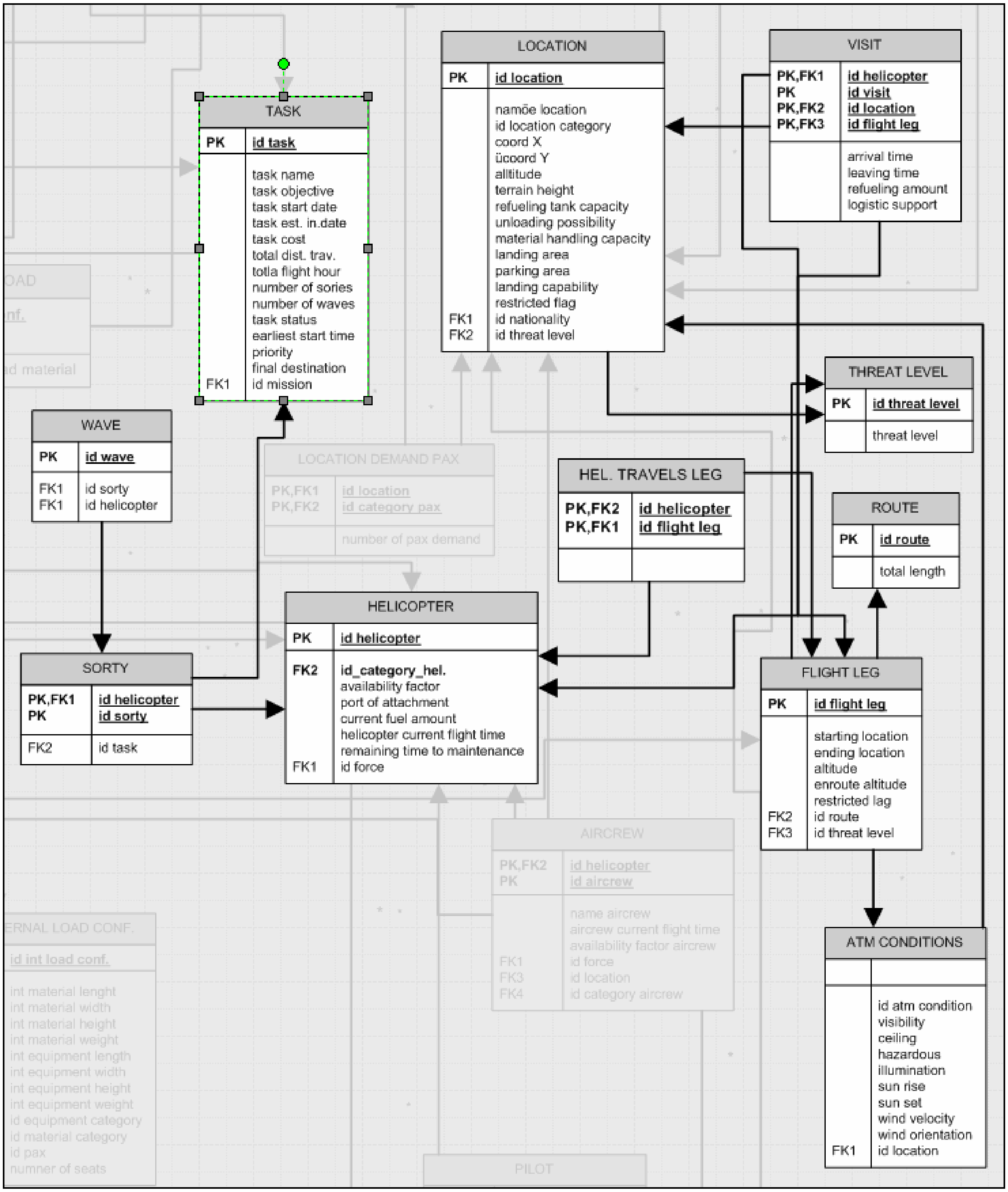


Figure 4.8: Location Visits.

4.3.6 Tables for "Bin Packing of Helicopters"

- Material;
- Equipment;
- Pax;
- Helicopter;
- Helicopter Category;
- Internal Load Configuration;
- External Load Configuration; and
- Flight Leg.

There are two types of transportation activity performed by helicopters: external load and internal load as shown in Figure 4.9. The internal load and external load capacities of helicopters are categorized according to their categories. The helicopter load relation is demonstrated with a table in this model. An important restriction is related with the loading patterns of the helicopters such that a helicopter should be assigned one of the strict load configurations while flying through a flight leg. Therefore, load configurations of helicopters are stored in the database as it changes in the flight legs.

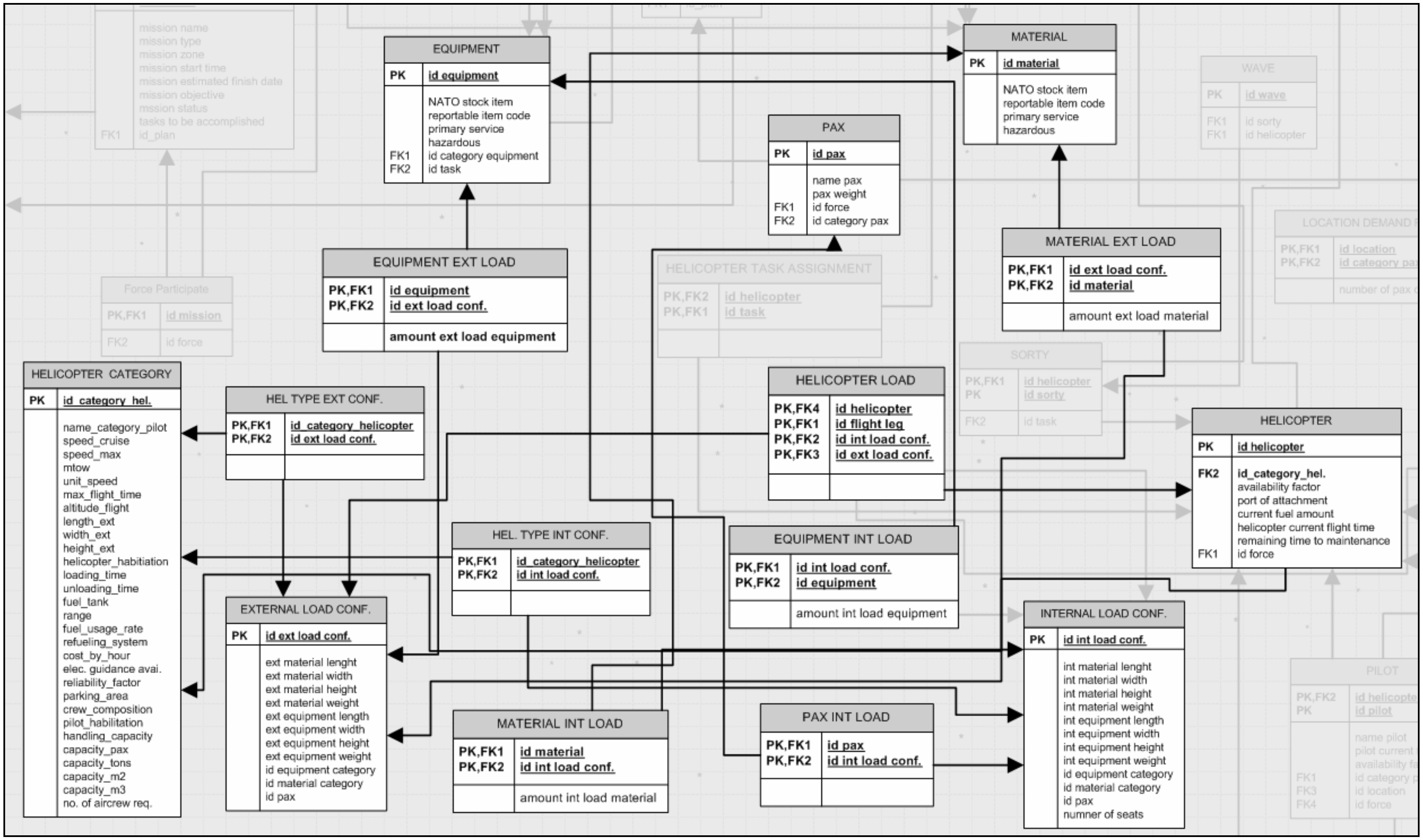


Figure 4.9: Binpacking of Helicopters.

4.3.7 Tables for "System Parameters"

- System Parameter; and
- Optimization Criteria.

In the last portion of the database, there is "System Parameters" table shown in Figure 4.10, which includes parameters needed in solving the optimization problem, and "Optimization Criteria" table, which includes alternative optimization criteria.

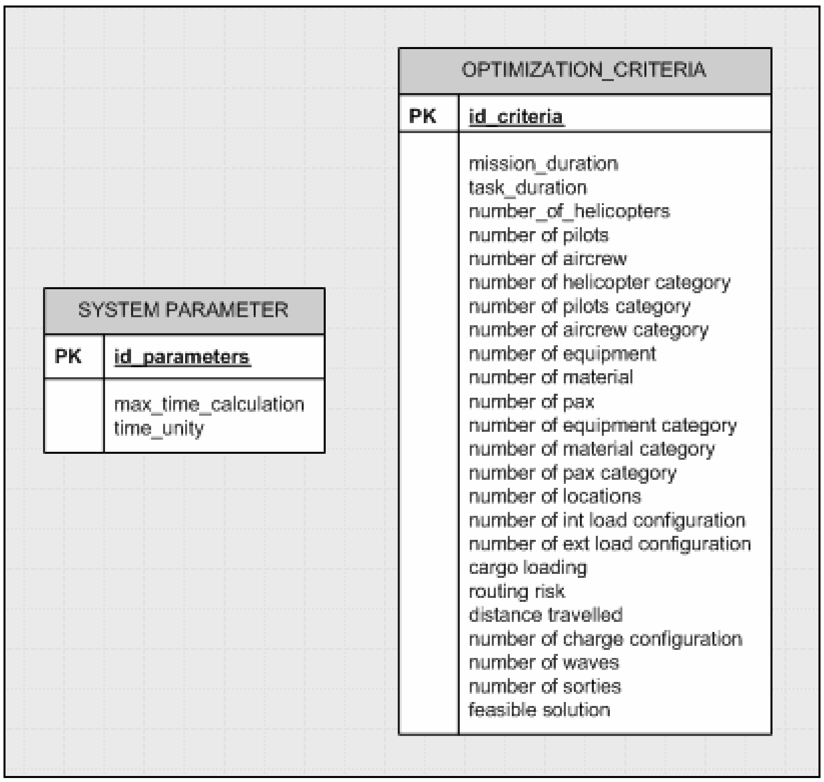


Figure 4.10: System Parameters.

4.4 MODEL NA2: NARAT-BASED RELATIONAL DATA MODEL

The three major components modeled in the relational database model in this section are catalogue, mission and the task database, as mentioned above. In NATO operations, NATO Request for Air Transport (NARAT) corresponding to the mission database provides a standardized message format to be used by national military airlift authorities to request airlift assistance from other NATO countries. This is also used by a NATO commander to request airlift support for a purely NATO unit. The details are provided in Appendix 4.1. All the information about the cargo, pax and air evacuation demand exists in NARAT document as well as the identity information of the task. Furthermore, the task database, which includes the result of the optimization module, is represented in the form of Transportation for Airlift Response (TRANSAR) document. This is used by the NATO military airlift agencies to confirm to all parties that a nation has accepted a tasking for operation of an airlift mission through NATO coordination. In this regard, it is a formal tasking to the providing nations and a formal reply to the requesting nation. The details are provided in Appendix 4.2. In connection with a NARAT, a TRANSAR presents the order information and assignments as the result of planning process. In order to take into account the NATO procedures a new relational data model including the NARAT and TRANSAR is developed. The new model is given in Figure 4.11.

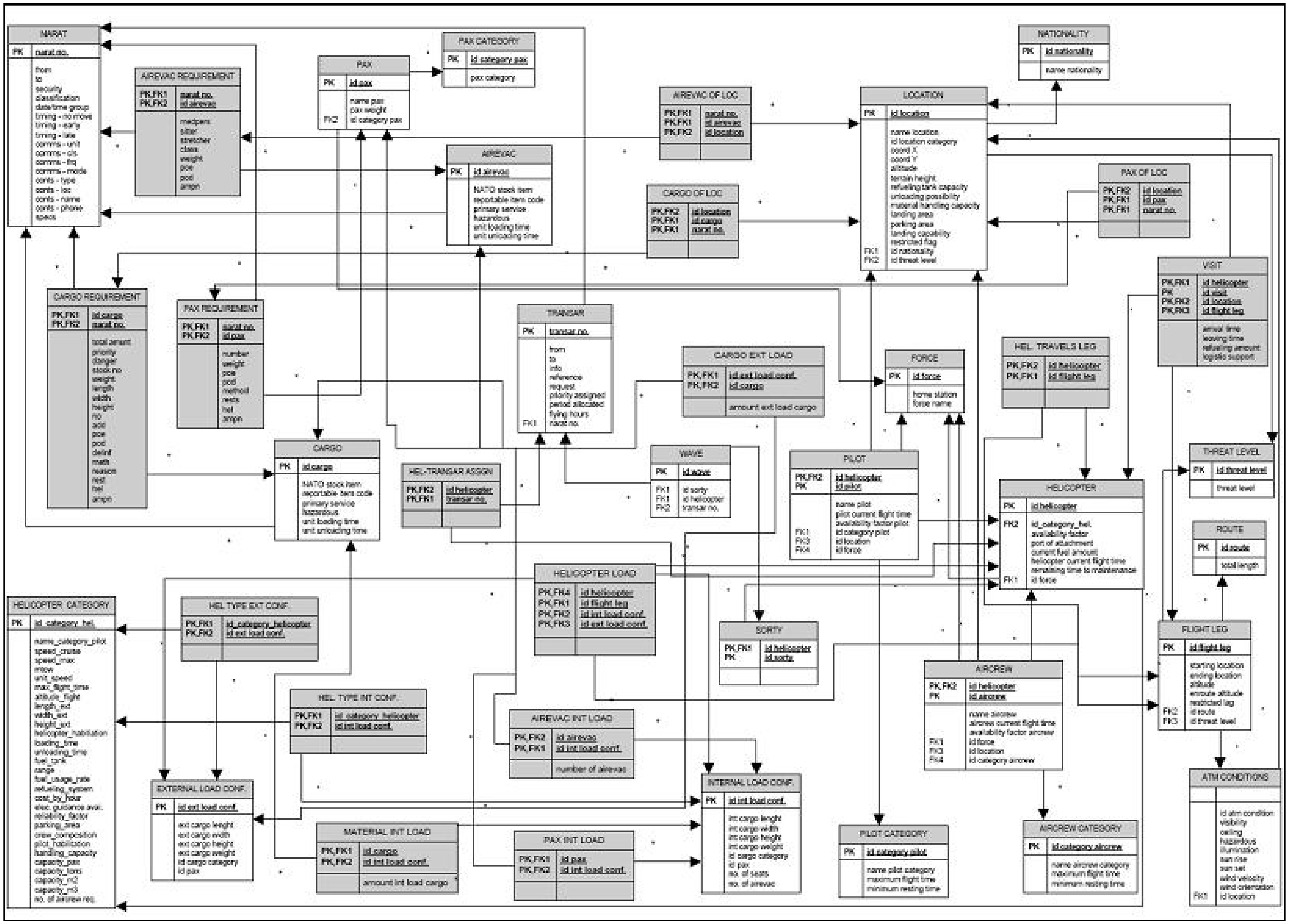


Figure 4.11: NARAT – TRANSAR Version of the Relational Database Model.

4.4.1 Tables for "NARAT – TRANSAR Relations"

- NARAT;
- TRANSAR;
- Location;
- Cargo;
- Pax;
- Air Evacuation;
- Cargo Requirement;
- Pax Requirement;
- Airevac Requirement;
- Location of Air Evacuation;
- Cargo of Air Evacuation;
- Helicopter; and
- Wave.

The main changes in the relational data model are zoomed in Figure 4.12. The identity of a NATO operation is represented in the NARAT table with NARAT No. as its primary key. The relationship between Cargo, Pax and Air Evacuation tables are modeled by requirement tables for the three requirement types. The amount, dimension and weight fields exist in the requirement tables as well as the point of embarkation and point of debarkation of the three types. These points are connected to the Location table through a relation table, namely, Location of Cargo, Location of Pax and Location of Air Evacuation.

The result of the optimization tool is given in TRANSAR table. In this table, the information with primary key "TRANSAR no." includes the identity information of this task assignment and detailed assignment information and the related NARAT record. The assignment of Helicopters and assigned waves to the helicopters are modeled in connection with TRANSAR.

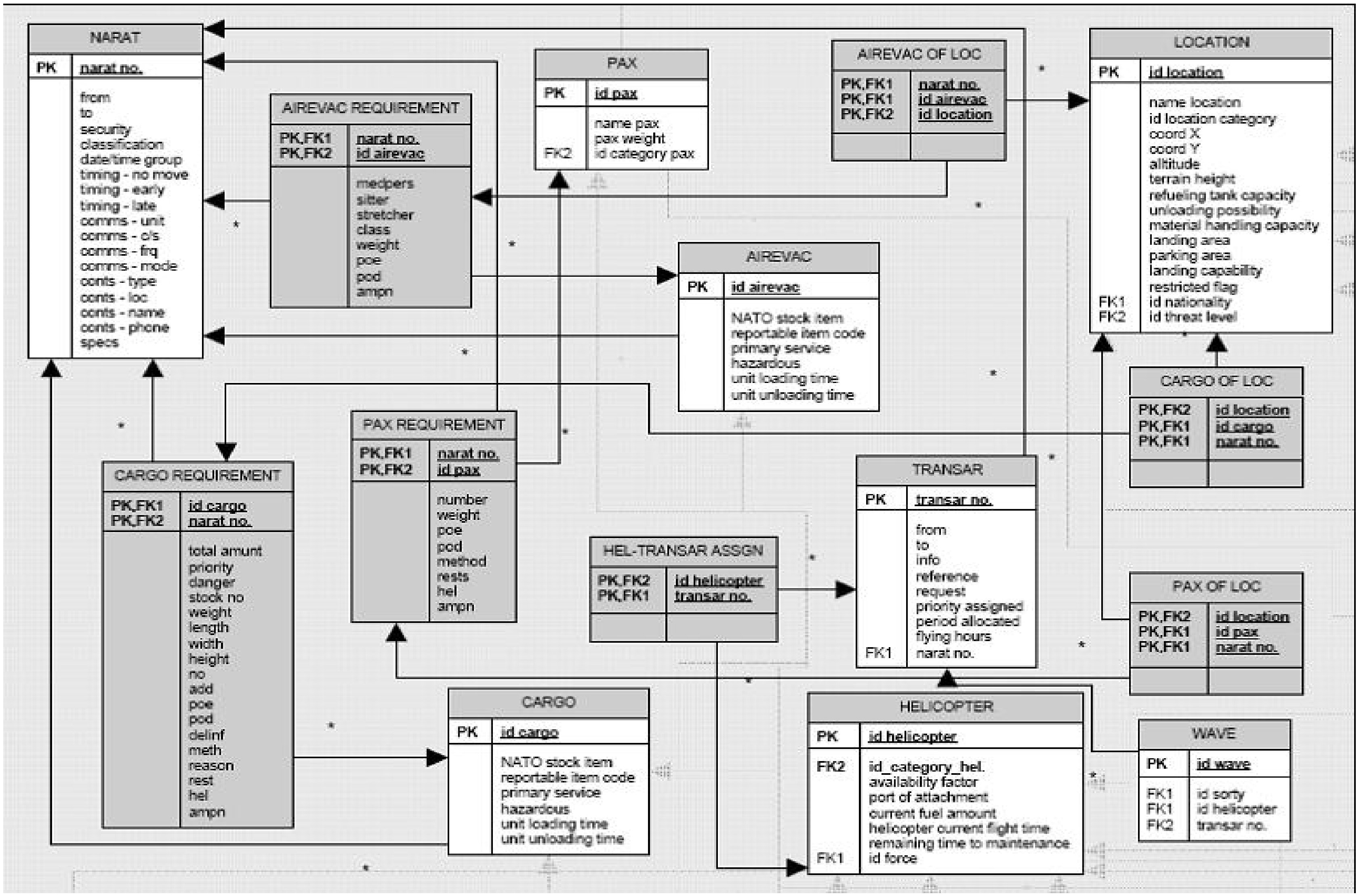


Figure 4.12: NARAT – TRANSAR Relations.

Appendix 4.1 – NATO REQUEST FOR AIR TRANSPORT (NARAT)

MANDATORY	CONDITIONAL	EXPLANATION
		a) Message Header Line
Precedence Action		Immediate requests should normally be given the precedence of OPERATIONAL IMMEDIATE.
		For planned requests – as required.
FROM		Requesting unit/formation
TO		Tasking Agency
INFO		Others
Security Classification		Normally RESTRICTED or high classification, but in an emergency it may be sent in clear. Use ZULU time.
Date/Time Group		
EXER		Exercise identification
OPER		Operation name
		b) Message Identifier/Subject
NARAT		Air Transport Request
No…		Request number. The request letter and number group will identify the requesting formation and the serial number of its request; identifying letters and blocks of numbers will be allocated to formations by the senior formation headquarters in the theatre. Whenever possible this request number should be used as the task/ mission number.
		c) Message Text
1) PAX	A) NUMBER	Carriage of Passengers/Troops
		Number of Troops
	B) WEIGHT	Calculated estimation of troop weight, including baggage/ equipment, followed by letters: STN for Short Tons (2000 lbs) MTT for Metric Tons LTN for Long Tons (2240 lbs) KGS for Kilograms LBS for Pounds

MANDATORY CONDITIONAL		EXPLANATION
	C) POE	Point of Embarkation
		Given as one of the following or a combination thereof
		a) Co-ordinates in either
		Lat/Long
		UTM-Grid or
		GEOREF
		b) Identification
		(e.g. ICAO-station designator identification of DZ/EZ)
	D) POD	Port of Debarkation
		(Given as the same was as POE)
	E) METHOD	**Preferred delivery method**
		LAND – air land
		PARA – drop of personnel, automatic parachute activation
		PARF – drop of personnel, manual parachute activation
		PARG – drop of personnel, paragliders
		Height band as per STANAG 3737
		ULL – ultra low level
		LOW – low level
		MED – medium level
		HIG – higher level
	F) RESTS	**Restrictions, if POD is an Airstrip**
		TAXI – limited LAND – air land
		RAMP – limited/no parking facilities
		OBST – obstacles approach/climb out sector
		COMS – limited communication facilities
		NORA – no radar surveillance/approach aids
		INST – no instrument approach facilities
		MARK – no R/W – Appr. Markings/lighting
	G) HEL	Helicopter Landings
		LP – Number of landing points for helicopter within landing
	H) AMPN	Amplification
		Additional information as required

MANDATORY	CONDITIONAL	EXPLANATION
2) CARGO	A) TOTAL	**Carriage of Cargo** Total weight of cargo, given in numbers followed by Letters: STN – 2000 lbs MTT – Metric Tons KGS – Kilograms LTN – Long Tons (2240 lbs) LBS – Pounds
	B) TYPE	**Type of Cargo** AIRF – airframe or major part thereof BAGG – baggage CNET – cargo net CONT – container DORB – door bundle HELI – helicopter IGLO – miscellaneous general cargo un palletized (to be defined in set AMPN) NOST – non-standard/outsized cargo (to be defined in set AMPN) PALL – pallets ROLL – rolling stock (vehicles, trolleys, etc.) TANK – external/collapsible fuel tanks, tip tanks WEDG – equipment on wedge system E – cargo carried externally
	C) PRIORITY	**Priority code of cargo** (As per STANAG 3631) 1 = priority 1 (IMMEDIATE) 2 = priority 2 (URGENT) 3 = priority 3 (ROUTINE)
	D) DANGER	Use either danger classification as detailed in ICAO Regulations, if known, or Y for "YES" if cargo is dangerous. Otherwise letter N for "NO".
	E) STOCK NO	If cargo is dangerous, this field must be completed. It is used to define dangerous cargo by giving its article or stock number, UN-Number or actual nomenclature.
	F) WEIGHT	Weight of individual piece of cargo, as given in TOTAL

MANDATORY	CONDITIONAL	EXPLANATION
	G) LENGTH	Length of cargo, given in numbers, followed by letter: M for Meters C for Centimeters F for Feet I for Inches
	H) WIDTH	Width of cargo given as above
	I) HEIGHT	Height of cargo given as above
	J) NO	No. of items of cargo as described under type
	K) ADD	Additional information
	L) POE	Point of Embarkation (given the same way as for transport of Pax)
	M) POD	Point of Debarkation (given the same way as for transport of Pax)
	N) DELINF	Delivery information
	O) METH	**Preferred Delivery Method** LAND – air land CARG – cargo drop, gravity CAEX – cargo drop, extraction chute DORP – door bundles, parachute (side doors) DORF – door bundles, freefall (side doors) WEDG – cargo drop be wedge system LAPE – low Attitude Parachute Extraction System AGRA – airdrop, gravity (no chute) RESC – airdrop of rescue equipment LIQU – drop of liquids (water for firefighting, chemicals against pollution, etc.) **Height band as per STANAG 3737** ULL – ultra low level LOW – low level MED – medium level HIG – high level UHL – ultra high level NRQ – not required

MANDATORY CONDITIONAL		EXPLANATION
	P) REASON	Reason for preferred delivery method TRG – training requirement TAC – tactical requirement GEO – geographical requirement TIM – time requirement CGO – cargo requirement GND – restrictions due to limited ground handling facilities
	Q) REST	**Restrictions if POD/POD is an airstrip** TAXI – limited/no taxiways RAMP – limited/no parking facilities OBST – obstacles approach/climb out sector COMS – limited communication facilities NORA – no radar surveillance/approach aides INST – no instrument approach facilities MARK – no R/W – Approach markings/lighting
	R) HEL	Helicopter landing (given the same way as for transport of Pax)
	S) AMPN	Amplification Additional information as required
3) AIREVAC		Aeromedical evacuation
	A) MEDPERS	No. of medical personnel nominated to accompany
	B) SITTER	Total number of sitting patients using one, two or three numbers followed by M (male) and one, two or three numbers followed by F (female).
	C) STRETCHER	Total number of stretcher patients given as above
	D) CLASS	Number of sitter/stretcher patients by classification as per STANAG 3204 AMD CLASS NUMBER M of F 1A / 1B/ 2A / 2B / 3 and 4 as shown
	E) WEIGHT	Total weigh of patients and attendants given in numbers, followed by letters: TON – Tons MTT – Metric Tons LTN – Long Tons KGS – Kilograms LBS – Pounds
	F) POE	Point of Embarkation (given the same way as for transport of Pax)

MANDATORY	CONDITIONAL	EXPLANATION
	G) POD	Point of Debarkation (given the same way as for transport of Pax)
	H) AMPN	Amplification Additional information as required (e.g. requirements for special cabin altitude, additional medical personnel, supplies and/or equipment in flight or at destination)
4) TIMING		Timing
	A) NO MOVE	No move before given time expressed as DTG + month (3 letters)
	B) EARLY	Earliest delivery time (given as above)
	C) LATE	Latest delivery time, after which air transport not required or unacceptable (given as above)
5) COMMS		Communications
	A) UNIT	Unit
	B) C/S	Callsign
	C) FRQ	Frequency + Priority P = Primary S = Secondary T = Tertiary
	D) MODE	Transmission mode (e.g. Voice, SSB, CW)
6) CONTS		Contacts
	A) TYPE	Type of contact (onload/offload)
	B) LOC	State location of contact
	C) NAME	Provide full name and rank
	D) PHONE	Identify office/home phone number to include service and/or commercial line
7) SPECS		Special instructions, any relevant information not covered above

Appendix 4.2 – TRANSPORTATION FOR AIRLIFT RESPONSE (TRANSAR)

FROM: NATO Military Airlift Agency concerned (MSC or AMCC).

TO: Requesting Agency.

INFO: Agencies involved in coordination.

(NATO Classification)
EXER/Exercise Name or OPER/Operation Name.

SUBJECT: TRANSAR number and DTG

1) Reference (NARAT message number and DTG).
2) Request (accepted/refused). (Short explanation if refuse and any alternative offered).
3) Priority assigned.
4) Type and number of aircraft allocated.
5) Period allocated (From/to DTG).
6) Flying hours and sorties allocated.

Chapter 5 – GENERIC MODELLING FRAMEWORK

As it is mentioned in the previous chapters, the optimization problem inherent in the decision support tool is to determine the optimum routes for helicopters to carry out material and human transfer between locations. The problem has some unique characteristics. First, helicopters initially sit on operation bases, but they can return to any one of the operation bases. Considering the cargo capacity and incorporating fuel consumption into the problem, the trade-off between the flight length of helicopters on a route and the transportation amount is clearly treated in the model.

It is aimed to determine the optimum routes for helicopters, load/unload quantities, refueling activities and load configurations while carrying out material and human transfer between locations. With reference to the NATO glossary, two types of locations are mentioned in the study: Operation Bases and Demand Locations. Operation Bases are the locations, which supply material and civilian and military personnel to demand points and provide medical service to the evacuated people as well. Demand Locations represent the hazardous and disaster points scattered geographically in an emergency area over which NATO forces serve to carry out disaster relief.

There are three basic requirements in this problem.

i) Material transfer from operation bases to demand locations;

ii) Pax transfer from operation bases to demand locations; and

iii) Human evacuation from demand locations to operation bases.

Different model formulations are developed and discussed in the following sections with different assumptions.

5.1 MATHEMATICAL MODELS

5.1.1 Basic Formulation: MODEL C1

The principal inputs of the model can be classified as; requirements of demand locations, supply quantities of operation bases, availability, and capacities of helicopters.

5.1.1.1 Requirements of Demand Locations

The demand amount for each type of material and human evacuation requirement is declared by demand locations.

5.1.1.2 Supply Information of Operation Bases

The supply amount for each type of material, human and medical service availability should be provided.

5.1.1.3 Helicopter Parameters

Human and material cargo capacities and human evacuation capabilities for each type of helicopter are major inputs of the system regarding transportation by helicopters.

5.1.1.4 Routing Related Parameters

In order to organize the routing, the distances between locations, initial locations of the helicopters, and fuel tank capacities of the helicopters are needed.

5.1.1.5 Time Related Parameters

In order to take into account time-related constraints, speed, load, unload times of cargo and fuel consumption of helicopters must be provided.

5.1.2 Outputs of the Model

The model is developed to provide the optimum solutions for;

5.1.2.1 Routes of Helicopters

The departure and landing points of each helicopter on their routes will be detailed in this context.

5.1.2.2 Material – Human Load and Unload Quantities

Helicopters load material and human in operation bases, deliver, and unload them at demand locations. However, the evacuation is done from demand locations to operation bases. While the helicopters are being routed, the loading and unloading amounts and locations will be determined by solving the model.

5.1.2.3 Refueling Locations and Amounts

Refueling can only be done in operation bases. The fuel tank capacity and the remaining amount of fuel restrict the distance that the helicopter can fly. Therefore, the information should be generated as an operation order by the model.

5.1.2.4 Bin Packing of Helicopters at Each Flight Leg

The loading pattern of cargo, pax and evacuation needs presents itself as a modeling decision; however for the sake of simplicity different load configurations are predetermined among which the decision support tool will select the best one. In addition, there are some hazardous materials and health related restrictions that constrain the simultaneous transportation of some types of material and pax.

5.1.3 Assumptions of the Model

The main assumptions of the model are given below:

5.1.3.1 Routing Assumptions

Helicopters must start from their initial nodes, which is an operation base, and can finish the route at any operation base. In addition, each helicopter can fly between two nodes only once.

5.1.3.2 Cargo Capacity Assumptions

Different cargo types are assumed to have an average weight and helicopters cannot exceed their overall capacity.

5.1.3.3 Demand Assumptions

All requirements of each demand location must be met by the operation bases. Human evacuation requested by demand locations must be carried to the operation bases, which provide medical service.

5.1.3.4 Fuel Assumptions

Each type of helicopter has a known and constant fuel consumption rate per distance and it cannot complete any flight leg without sufficient amount of fuel. Refueling of helicopters can only be done in operation bases, while there is no limitation on the fuel amount provided at each operation base. However, a helicopter cannot drain fuel. The fuel consumption rate of the helicopters is assumed constant and independent of the current load of the helicopter. Therefore, the maximum value of the fuel consumption rate should be taken to prevent infeasibility.

5.1.4 Model Formulation

First, a mixed integer formulation of the problem stated in previous sections is developed under the assumption that the type and number of available helicopters is fixed and limited as given below:

$$\min \sum_i \sum_j d_{ij} \sum_t X_{ij}^t \tag{5.1}$$

subject to:

$$\sum_j X_{ji}^t = \sum_j X_{ij}^t \qquad \forall (i \in N_D, t) \tag{5.2}$$

$$\sum_j X_{ji}^t \geq \sum_j X_{ij}^t \qquad \forall (i \in N_S, t : H_i^t = 0) \tag{5.3}$$

$$\sum_i X_{ij}^t \leq 1 \quad \forall (j, t) \tag{5.4}$$

$$\sum_j X_{ij}^t \leq 1 \quad \forall (i, t) \tag{5.5}$$

$$\sum_j X_{ij}^t \leq \mathrm{M}^*(\mathrm{H}_\mathrm{i}^\mathrm{t} + \sum_j X_{ji}^t) \qquad \forall (\ i, t) \tag{5.6}$$

$$U_i^t = 1 \qquad \forall (i, t : H_i^t = 1) \tag{5.7}$$

$$U_j^t - U_i^t \geq -M * (1 - X_{ij}^t) + 1 \qquad \forall (i, j, t : H_j^t = 0) \tag{5.8}$$

$$\sum_k (UWM_k * MC_{kij}^t) + F_{ij}^t + UWH * HC_{rij}^t + UWH * EC_{ij}^t \leq C^t \qquad \forall (i, j, t) \tag{5.9}$$

$$\sum_k (UWM_k * MC_{kij}^t) \leq CM^t \qquad \forall (i, j, t) \tag{5.10}$$

$$\sum_r (UWH * HC_{rij}^t) \le CH^t \qquad \forall(i,j,t) \tag{5.11}$$

$$UWH * EC_{ij}^t \le CE^t \qquad \forall(i,j,t) \tag{5.12}$$

$$MC_{kij}^t \le M * YMC_{kij}^t \qquad \forall(i,j,t,k) \tag{5.13}$$

$$HC_{rij}^t \le M * YHC_{rij}^t \qquad \forall(i,j,t,r) \tag{5.14}$$

$$EC_{ij}^t \le M * YEC_{ij}^t \qquad \forall(i,j,t) \tag{5.15}$$

$$YHC_{'1'ij}^t + YHC_{ij}^t \le 1 \qquad \forall(i,j,t) \tag{5.16}$$

$$\sum_t MU_{ik}^t = md_i^k \qquad \forall(i \in N_D^k, k) \tag{5.17}$$

$$\sum_t ML_{ik}^t \le ms_i^k \qquad \forall(i \in N_S^k, k) \tag{5.18}$$

$$\sum_j MC_{kij}^t \le ms_i^k \qquad \forall(i \in N_S^k, k, t : H_i^t = 1) \tag{5.19}$$

$$\sum_j MC_{kij}^t = \sum_j MC_{kji}^t + ML_{ik}^t \qquad \forall(i \in N_S^k, k, t) \tag{5.20}$$

$$\sum_j MC_{kji}^t - MU_{ik}^t = \sum_j MC_{kij}^t \qquad \forall(i \in N_D^k, k, t) \tag{5.21}$$

$$MC_{kij}^t \le M * X_{ij}^t \qquad \forall(i,j,k,t) \tag{5.22}$$

$$\sum_t HU_{ir}^t = hd_i^r \qquad \forall(i \in N_D^r, r) \tag{5.23}$$

$$\sum_t HL_{ir}^t \le hs_i^r \qquad \forall(i \in N_S^r, r) \tag{5.24}$$

$$\sum_j HC_{rij}^t \le hs_i^r \qquad \forall(i \in N_S^r, r, t : H_i^t = 1) \tag{5.25}$$

$$\sum_j HC_{rij}^t = \sum_j HC_{rji}^t + HL_{ir}^t \qquad \forall(i \in N_S^r, r, t) \tag{5.26}$$

$$\sum_j HC_{rji}^t - HU_{ir}^t = \sum_j HC_{rij}^t \qquad \forall(i \in N_D^r, r, t) \tag{5.27}$$

$$HC_{rij}^{t} \leq M \cdot X_{ij}^{t} \qquad \forall(i,j,r,t) \tag{5.28}$$

$$\sum_{t} EL_{i}^{t} = her_{i} \qquad \forall(i \in N_{D}) \tag{5.29}$$

$$\sum_{j} EC_{ij}^{t} = \sum_{j} EC_{ji}^{t} + EL_{i}^{t} \qquad \forall(i \in N_{D}, t) \tag{5.30}$$

$$\sum_{j} EC_{ji}^{t} - EU_{i}^{t} = \sum_{j} EC_{ij}^{t} \qquad \forall(i \in N_{S}, t) \tag{5.31}$$

$$EC_{ij}^{t} \leq M * X_{ij}^{t} \qquad \forall(i,j,t) \tag{5.32}$$

$$EU_{i}^{t} \leq M * hpt_{i} \qquad \forall(i,t) \tag{5.33}$$

$$F_{ij}^{t} \geq X_{ij}^{t} * d_{ij} * fc_{t} \qquad \forall(i,j,t) \tag{5.34}$$

$$F_{ij}^{t} \leq X_{ij}^{t} * FTC_{t} \qquad \forall(i,j,t) \tag{5.35}$$

$$RFA_{i}^{t} = \sum_{j} F_{ij}^{t} \qquad \forall(i \in N_{S}^{k}, t : H_{i}^{t} = 1) \tag{5.36}$$

$$\sum_{j} F_{ji}^{t} - \sum_{j} (D_{ji} * FC_{t} * X_{ji}^{t}) + RFA_{i}^{t} = \sum_{j} F_{ij}^{t} \qquad \forall(i \in N_{S}^{k}, t : H_{i}^{t} = 0) \tag{5.37}$$

$$\sum_{j} F_{ji}^{t} - \sum_{j} (D_{ji} * FC_{t} * X_{ji}^{t}) = \sum_{j} F_{ij}^{t} \qquad \forall(i \in N_{D}^{k}, t) \tag{5.38}$$

$$RFA_{i}^{t} \geq 0 \qquad \forall(i,t) \tag{5.39}$$

$$\sum_{l} LC_{lij}^{t} = 1 \qquad \forall(i,j,t) \tag{5.40}$$

$$\sum_{k} MC_{kij}^{t} \leq M * (LC_{'2'ij}^{t} + LC_{'5'ij}^{t} + LC_{'7'ij}^{t} + LC_{'8'ij}^{t}) \qquad \forall(i,j,t) \tag{5.41}$$

$$\sum_{r} HC_{rij}^{t} \leq M * (LC_{'3'ij}^{t} + LC_{'5'ij}^{t} + LC_{'6'ij}^{t} + LC_{'8'ij}^{t}) \qquad \forall(i,j,t) \tag{5.42}$$

$$EC_{ij}^{t} \leq M * (LC_{'4'ij}^{t} + LC_{'6'ij}^{t} + LC_{'7'ij}^{t} + LC_{'8'ij}^{t}) \qquad \forall(i,j,t) \tag{5.43}$$

$$\sum_{k} MC_{kij}^{t} \geq LC_{'2'ij}^{t} + LC_{'5'ij}^{t} + LC_{'7'ij}^{t} + LC_{'8'ij}^{t} \qquad \forall(i,j,t) \tag{5.44}$$

$$\sum_r HC^t_{rij} \geq LC^t_{'3'ij} + LC^t_{'5'ij} + LC^t_{'6'ij} + LC^t_{'8'ij} \qquad \forall (i,j,t) \tag{5.45}$$

$$EC^t_{ij} \geq LC^t_{'4'ij} + LC^t_{'6'ij} + LC^t_{'7'ij} + LC^t_{'8'ij} \qquad \forall (i,j,t) \tag{5.46}$$

$$\sum_k MC^t_{kij} \leq M * (1 - LC^t_{'1'ij}) \qquad \forall (i,j,t) \tag{5.47}$$

$$\sum_r HC^t_{rij} \leq M * (1 - LC^t_{'1'ij}) \qquad \forall (i,j,t) \tag{5.48}$$

$$EC^t_{ij} \leq M * (1 - LC^t_{'1'ij}) \qquad \forall (i,j,t) \tag{5.49}$$

$$X^t_{ij},\ P^t_{pij},\ LC^t_{lij},\ YMC^t_{kij},\ YHC^t_{ij},\ YEC^t_{ij} \ \in \{0,1\} \tag{5.50}$$

$$MC^t_{kij}, HC^t_{rij}, EC^t_{ij}, ML^t_{ik}, MU^t_{ik}, HL^t_{ir}, HU^t_{ir}, EL^t_i, EU^t_i, F^t_{ij}, RFA^t_i \geq 0 \tag{5.51}$$

Sets

i , j	total number of nodes
t	number of helicopters
p	number of pilots
k	number of types of materials
r	different genders of human beings
N_D	set of demand nodes
N_S	set of operation bases

Decision Variables

X^t_{ij}	a binary variable to indicate if helicopter t visits node i from node j
U^t_i	visiting sequence of node i of helicopter t (integer)
MC^t_{kij}	amount of material k carried by helicopter t from i to j (integer)
HC^t_{rij}	amount of human k carried by helicopter t from i to j (integer)
EC^t_{ij}	amount of human k evacuated by helicopter t from i to j (integer)
YMC^t_{kij}	a binary variable that indicates if helicopter t carries material type k from node i to node j
YHC^t_{ij}	a binary variable that indicates if helicopter t carries human from node i to node j
YEC^t_{ij}	a binary variable that indicates if helicopter t carries evacuated human
ML^t_{ik}	amount of material k loaded at node j by helicopter t (integer)

MU_{ik}^{t} amount of material k unloaded at node j by helicopter t (integer)

HL_{ir}^{t} number of people carried by helicopter t from node i (integer)

HU_{ir}^{t} number of people left at node i by helicopter t (integer)

EL_{i}^{t} number of people evacuated by helicopter t from node i (integer)

EU_{i}^{t} number of people left at node i by helicopter t (integer)

F_{ij}^{t} fuel amount of helicopter t while traveling from node i to j

RFA_{i}^{t} refueling amount of helicopter t in node i

P_{pij}^{t} a binary variable that indicates if pilot p is assigned to helicopter t in order to fly from node i to node j

LC_{lij}^{t} a binary variable that indicates if helicopter t has load configuration l while flying from node i to j

Parameters

d_{ij} distance between node i and node j

h_{i}^{t} initial location of helicopter t

c^{t} total cargo capacity of helicopter t

uwm_{k} unit weight of material k

uwh unit weight of a human being

cm_{t} material cargo capacity of helicopter t

eh_{t} human evacuation capacity of helicopter t

ch_{t} human transport capacity of helicopter t

md_{i}^{k} amount of material demand of type k of node i

ms_{i}^{k} amount of material supply of node i

hd_{i}^{r} personnel demand of type r of node i

hs_{i}^{r} personnel human supply of node i

her_{i} number of human beings to be evacuated at node i

hpt_{i} a binary number that indicates the availability of medical service in node i

fc_{t} fuel consumption of helicopter t per distance

ftc_{t} fuel tank capacity of helicopter t

5.1.4.1 Objective Function

Minimizing the total distance traveled by all helicopters is selected to be the objective function for this MIP model, which is expressed in (5.1). However, different objective functions can be used in the model; minimization of the total flight duration, maximization of human evacuation, minimization of the number of helicopters.

5.1.4.2 Routing Constraints

Since helicopters are supposed to start from their initial nodes, they have to leave the demand nodes that they visit. However, helicopters can finish their routes either in their initial nodes or in some other operation base. Constraints (5.2) and (5.3) are included to serve these purposes. In addition to these constraints, (5.4) and (5.5) ensures that the helicopters visit each node at most once. Equations (5.6) – (5.8) are sub-tour elimination constraints. In (5.6), each helicopter is just allowed to leave a node if either it arrives at that node or the node is its initial position. Nevertheless, this constraint cannot prevent sub-tours alone. Among the sub-tour elimination constraints, the last two provide a numbering of the visited nodes for each helicopter. In (5.7) the numbering of the initial node of each helicopter is initialized. The constraint (5.8) assigns numbers to successive nodes as they are visited by incrementing the numbering variable. Since the numbers of visited nodes are incremented sequentially, a helicopter cannot make a sub-tour that does not contain its initial node.

5.1.4.3 Cargo Capacity Constraints

The capacities of the helicopters are limited and determined by its type. The total load of a helicopter consists of three types of loads: material load, human load (both evacuating and pax), and fuel as in (5.9). Besides this total capacity constraint, (5.10) – (5.12) limit the transportation amount of each type of load in a helicopter. Equations (5.13) – (5.15) are included determine whether a certain type of load is transported by a helicopter at a particular flight leg. Some specific types of material may be restricted to be transported together in the same helicopter as given in constraint (5.16).

5.1.4.4 Material and Human Transportation Constraints

Material and personnel demand of each location is satisfied by delivering the desired type of material and human to the right location. This is provided by (5.17) for material and (5.23) for personnel. According to the assumptions of the problem, material and personnel availability for each type is limited and given in (5.18) and (5.24). (5.19) and (5.25) describe the material and human loading procedure in an operation base while a helicopter is visiting that operation base. Equations (5.20) and (5.26) prevent load-cycling in the model. Material and human delivery are described in (5.21) and (5.27) by decreasing the respective amounts from one flight leg to next one. (5.22) and (5.28) represent the necessity of assigning a flight to a helicopter, where transportation is required through that flight by that helicopter.

5.1.4.5 Human Evacuation Constraints

Human evacuation requirements of the demand locations are met completely by using the constraint (5.29). Loading and unloading procedure in human evacuation is described in (5.30) and (5.31), respectively. Constraint (5.32) guarantees to establish a flight through the legs in which evacuated human transportation is needed. In addition, the availability of the medical service in operation bases is included in the model by giving (5.33).

5.1.4.6 Fuel Constraints

The fuel amount available in the tank during a flight leg should be sufficient to fly that distance as expressed in (5.34). Adding the constraint (5.35) sets a limitation on the fuel amount in each helicopter for each flight leg. The initial fueling of helicopters is given in (5.36). Refueling procedure, which can be summarized as subtracting the consumed fuel amount from the current fuel amount and adding refueling amount, is given in (5.37). This constraint is required only when visiting the operation bases; on the other hand, there is just fuel consumption for the visits of demand locations. This case is expressed in (5.38). Refueling is forced to be positive by (5.39) in order to prevent emptying the tank.

5.1.4.7 Bin Packing Constraints

Bin packing problem is embedded into the model by (5.40) – (5.49). Eight types of load configurations are given in Table 5.1. The “X” mark in a row shows that the type of load given in the column is included in the respective load configuration. For example, if the load configuration index of helicopter is 5 in a flight leg, it means that the helicopter is transporting both human and material, but not evacuating people in that flight leg.

Table 5.1: Load Configurations

Load Configuration	Material Transportation	Human Transportation	Human Evacuation
1	-	-	-
2	X	-	-
3	-	X	-
4	-	-	X
5	X	X	-
6	-	X	X
7	X	-	X
8	X	X	X

The necessity of assigning one and only one load configuration to each helicopter on each flight leg is given in (5.40). By (5.41) – (5.49), cargo transport variables are transformed into load configuration variables without bringing any burden on the model.

Since the resulting model is NP-complete, it is important to analyze the dimensionality of the problem. The importance of determining the number of variables and constraints inherits from their effect on the solution time. The number of variables and constraints depends on the size of the sets. These sets are cardinalities of demand nodes (d), operation bases (ob), helicopters (t), pilots *(p)* types of materials (k), and types of human (r). The number of variables and constraints in MODEL-C1 are given below.

$$\text{Number of Variables} = t*(d+ob)^2*\left(2k+r+5\right) \\ +2\left(d+ob\right)\left(k+r+2\right) \tag{5.52}$$

$$\text{Number of Constraints} = t*(d+ob)^2\left[2(k+r)+7\right] + (d+ob)\left[t(d+k+r+8)+2k+r\right]+o(k+1)t \quad (5.53)$$

These equations show that the number of helicopters and total number of nodes mainly affect both the number of variables and constraints. Equation (5.52) and Equation (5.53), the number of variables and number of constraints for different problem sizes, reveal that multiplying the total number nodes with q would approximately make the number of variables multiplied by q^2. This second order multiplier relationship is also valid for the number of constraints. Both the number of variables and the number of constraints are directly proportional to the number of helicopters in the model as it is naturally expected.

Table 5.2: Dimensionality of MODEL-C1

d+ob	d	ob	t	k	r	Number of	
						Variables	Constraints
7	5	2	2	2	2	2128	2981
14	10	4	2	2	2	8176	11448
21	15	6	2	2	2	18144	25403
28	20	8	2	2	2	32032	44846
6	5	1	2	2	2	1584	2213
12	10	2	2	2	2	6048	8456
18	15	3	2	2	2	13392	18731
7	5	2	2	2	2	2128	2981
7	5	2	3	2	2	3192	4455
7	5	2	4	2	2	4256	5929
25	20	5	1	2	2	12800	17971
25	20	5	2	2	2	25600	35822
25	20	5	3	2	2	38400	53673
25	20	5	4	2	2	51200	71524

5.2 FORMULATION FOR UNLIMITED HELICOPTER CASE: MODEL – NC2

In the previous model, the objective of the formulation is to minimize the total distance traveled as to complete the operation in the shortest possible duration under the assumption that the number of helicopters is limited. In critical NATO operations, especially in case of disaster relief, the number of helicopters can be increased beyond limit or there might exist certain situations when it is desired to minimized the number of helicopters are used to accomplish the mission. Thus, another formulation is developed where the number of helicopters is relaxed and the model is constructed so that the number of needed helicopters is to be determined by minimizing the cost of the complete operation. The major cost account in this problem is the flight cost of helicopters. The cost of using a helicopter, represented with $HelC_t$ depends on its type. In order to record the utilization of helicopters, a new binary variable, $HelU_t$, is defined to represent whether helicopter *t* is used or not. Then, the new objective function to replace (1) is given by

$$\min \sum_{t} HelC_t \times HelU_t \tag{5.54}$$

To evaluate the new variable, the following constraint is added to the original model.

$$HelU_t \leq M \times \sum_{i} \sum_{j} X_{ij}^{t} \qquad \forall(t) \tag{5.55}$$

5.3 FORMULATION BASED ON NARATS: MODEL – AN3

In NATO operations, the air transportation requirements are announced by the NATO Requests for Air Transport (NARAT). During military combat service support missions, peace support operations, humanitarian missions, and disaster relief operations NARAT documents, which present the necessary information on the transfer of cargo and pax as well as the air evacuation needs, are formed and published.

In the new formulation, NARATs are defined as the transportation requirements. In each NARAT, three types of transportation requirements, cargo and pax transportation and air evacuation, with its point of embarkation and points of debarkation are given. Most of the problem characteristics are preserved in this model. Again, two types of locations are mentioned in the study: Operation Bases and Demand Locations. Operation Bases are the locations, which supply material and civilian and military personnel to demand points and provide medical service to the evacuated people as well. Demand Locations represent the hazardous and disaster points scattered geographically in an emergency area over which NATO forces serve to carry out disaster relief. The MODEL-C1 determines the assignment of supply transfers from operation bases to demand locations simultaneously with the other helicopter assignment and routing decisions. That means, which demand requirement will be satisfied from which operation base is not specified a-priori, but the mapping between operation bases and demand locations, in other words the mapping between embarkation and debarkation locations, is achieved within the model engine. On the other hand, when NARATs are being used as the main task request for the model, the transportation scheme will have been semi-processed before running the model and the embarkation and debarkation nodes will have been paired. This simplifies the model structure considerably and requires a rather different approach to the problem. In the simplified form, the fuel consumption can also be treated off-line outside the model, and there is no need to include the fuel consumption of the helicopters within this new approach.

A Mixed Integer Programming (MIP) model is developed to determine which NARATs can be executed and in the execution of these NARATs the optimum routes for helicopters, load/unload quantities, refueling activities and load configurations while carrying out material and human transfer between locations.

5.3.1 Input Parameters of the Model

Although the complete list of catalogue parameters is provided in Chapter 4 the critical parameters needed for the optimization problem are defined below for the sake of completeness.

5.3.1.1 NARATs – Requirements and Supply Locations

The requirement and supply information are provided in NARATs, with point of embarkation and point of debarkation. In detail, the amount of cargo, pax, evacuation requirements, with arrival and departure locations are declared.

5.3.1.2 Helicopter Parameters

Pax, material cargo capacities and human evacuation capabilities for each type of helicopter are major inputs of the system regarding transportation by helicopters.

5.3.1.3 Routing Parameters

In order to organize the routing, the distances between locations and initial locations of the helicopters are needed.

5.3.2 Outputs of the Model

The model is developed to provide the optimum solutions for;

5.3.2.1 Accepted NARATs

The NARATs that are accepted to be executed are determined.

5.3.2.2 Routes of Helicopters

The departure and landing points of each helicopter on their routes will be detailed in this context. In addition, the assignments of helicopters to the accepted NARATs are produced by the formulation. The result is summarized in TRANSAR format.

5.3.3 Assumptions of the Model

The assumptions of the MIP model are given below:

5.3.3.1 Routing Assumptions

Helicopters must start routing from their initial nodes, which is an operation base, and can finish the route at any operation base. In addition, each helicopter can fly between two nodes only once on its route.

5.3.3.2 NARAT Assumptions

When a NARAT is accepted all three types of transfers declared in NARAT will be executed by the helicopter that is assigned to that NARAT if there are multiple requests in a given one. The requirements mentioned in NARATs are given in terms of mass.

5.3.3.3 Helicopter Transportation Capacity Assumptions

Helicopters have limited capacities in each type of transportation. Therefore, the number of NARATs that the helicopters are assigned are limited.

5.3.4 Model Formulation

$$\max \sum_{l} \left(pr_l \times AN_l \right) - \sum_{t} \sum_{i} \sum_{j} X_{ij}^{t} \tag{5.56}$$

subject to

$$\sum_j X_{ji}^t = \sum_j X_{ij}^t \quad \forall (i \in N_D, t) \tag{5.57}$$

$$\sum_j X_{ji}^t \geq \sum_j X_{ij}^t \quad \forall (i \in N_S, t : h_i^t = 0) \tag{5.58}$$

$$\sum_i X_{ij}^t \leq 1 \quad \forall (j,t) \tag{5.59}$$

$$\sum_j X_{ij}^t \leq 1 \quad \forall (i,t) \tag{5.60}$$

$$\sum_j X_{ij}^t \leq \mathrm{M}^*(\mathrm{h}_\mathrm{i}^\mathrm{t} + \sum_j X_{ji}^t) \quad \forall (i,t) \tag{5.61}$$

$$U_i^t = 1 \quad \forall (i, t : h_i^t = 1) \tag{5.62}$$

$$U_j^t - U_i^t \geq -M * (1 - X_{ij}^t) + 1 \quad \forall (i, j, t : h_j^t = 0) \tag{5.63}$$

$$AN_l = \sum_t HAN_l^t \quad \forall (l) \tag{5.64}$$

$$\sum_j X_{ij}^t \geq HAN_l^t \quad \forall (i, t, l : in_l^i = 1) \tag{5.65}$$

$$\sum_j X_{ji}^t \geq HAN_l^t \quad \forall (i, t, l : fn_l^i = 1) \tag{5.66}$$

$$U_i^t - U_j^t \leq M * (1 - HAN_l^t) - 1 \quad \forall (i, j, t, l : in_l^i = 1 \ \& \ fn_l^j = 1) \tag{5.67}$$

$$\sum_l \left(HAN_l^t \times mn_l \right) \leq mc_t \quad \forall (t) \tag{5.68}$$

$$\sum_l \left(HAN_l^t \times hn_l \right) \leq hc_t \quad \forall (t) \tag{5.69}$$

$$\sum_l \left(HAN_l^t \times en_l \right) \leq ec_t \quad \forall (t) \tag{5.70}$$

$$X_{ij}^t, AN_l^t, HAN_l^t \in \{0,1\} \tag{5.71}$$

$$U_i^t \geq 0 \tag{5.72}$$

Sets

i, j	total number of nodes
t	number of helicopters
l	number of NARATs
N_D	set of demand nodes
N_S	set of operation bases

Decision Variables

X_{ij}^t	a binary variable that indicates if helicopter t visits node i from node j
U_i^t	visiting sequence of node i of helicopter t (integer)
AN_l	visiting sequence of node i of helicopter t (integer)
HAN_l^t	visiting sequence of node i of helicopter t (integer)

Parameters

h_i^t	initial location of helicopter t
mc_t	material cargo capacity of helicopter t
hc_t	human transport capacity of helicopter t
ec_t	human evacuation capacity of helicopter t
in_l^i	initial node of NARAT l
fn_l^i	final node of NARAT l
pr_l	priority of NARAT l
mn_l	material load in NARAT l
hn_l	human load in NARAT l
en_l	evacuation load in NARAT l

5.3.4.1 Objective Function

Maximizing the number of NARATs accepted with highest priority is selected to be the objective function, which is expressed in (5.56). This objective function serves for the main challenge of this problem: selection of the most valuable NARATs.

5.3.4.2 Routing Constraints

Since helicopters are supposed to start from their initial nodes, they have to leave the demand nodes that they visit. However, helicopters can finish their routes either in their initial nodes or in some other operation base. Constraints (5.57) and (5.58) are included to serve these purposes. In addition to these constraints, (5.59) and (5.60) ensures that the helicopters visit each node at most once. Constraints (5.61) – (5.63) are sub-tour elimination constraints. In (5.61), each helicopter is just allowed to leave a node if either it arrives at that node or the node is its initial position. Nevertheless, this constraint cannot prevent sub-tours alone. Among the sub-tour elimination constraints, the last two provide a numbering of the visited nodes for each helicopter. In (5.62) the numbering of the initial node of each helicopter is initialized. The constraint (5.63) assigns numbers to successive nodes as they are visited by incrementing the numbering variable. Since the numbers of visited nodes are incremented sequentially, a helicopter cannot make a sub-tour that does not contain its initial node.

5.3.4.3 NARAT Assignment Constraints

As the NARATs are accepted, they should be assigned to helicopters. This assignment is given in (5.64). The helicopters that are assigned to a NARAT should leave the initial node of the NARAT and arrive to the final node of the NARAT once in its tour. This constraint is expressed by equations (5.65) and (5.66), respectively. Equation (5.67) guarantees that the final nodes of NARATs are going to be visited after their initial nodes.

5.3.4.4 Helicopter Capacity Related Constraints

Since the capacities of helicopters are limited in terms of cargo transportation, pax transportation, and air evacuation in each helicopter should be assigned to NARATs considering its total capacity. These considerations are given in equations (5.68) – (5.70) for three types of transportations, respectively.

This model has a smaller dimensionality than the previous one. Therefore, it would be computationally easier to solve this model. The number of variables and the number of constraints are given in equations (5.73) and (5.74) which show that the number of helicopters and the total number of nodes mainly affect both the number of variables and constraints. The number of NARATs has less effect on the problem. It can be observed that helicopter routing related constraints are the main cause of computational complexity.

$$\text{Number of Variables} = t * \left[(d+ob)^2 + (d+ob) + l + 1 \right] \tag{5.73}$$

$$\text{Number of Constraints} = t * \left[(d+ob)^2 + 4(d+ob) + 2l + 4 \right] + l \tag{5.74}$$

5.4 AN ALTERNATIVE FORMULATION BASED ON NARATS: MODEL – AN4

This mixed integer programming model presents another perspective to determine the number and routes of helicopters to execute the requirements announced by NARATs, as well as the load/unload details of helicopters in this execution. The main difference in this model is that helicopters are not identified uniquely, but as grouped according to their types; thus the main issue is to determine the number of helicopters of each type that should be assigned to execute each NARAT.

5.4.1 Input Parameters of the Model

5.4.1.1 NARATs – Requirements and Supply Locations

The requirement and supply information are provided in NARATs, with point of embarkation and point of debarkation. The amounts of cargo and pax transportation requirements are declared in detail.

5.4.1.2 Routing Related Parameters

In order to organize the routing, the distances between locations, initial locations of the helicopters are needed.

5.4.1.3 Helicopter Parameters

Human and material cargo capacities for each type of helicopter are major inputs of the system regarding transportation by helicopters. In addition, the number of available helicopters of each type should be known.

5.4.2 Outputs of the Model

The model is developed to provide the optimum solutions for:

5.4.2.1 Number of Assigned Helicopters

The numbers of assigned helicopters of each type to each NARAT with their initial locations are determined as major output.

5.4.2.2 Number of Sorties

In order to satisfy the demand of each NARAT, the number of sorties that will be done by each type of helicopter from each initial location is computed.

5.4.3 Model Formulation

$$\min\ TNH \text{ or } \min\ TFD \text{ or } \min\ TDT \tag{5.75}$$

subject to

$$\sum_t \sum_i MAT_{il}^t = mn_l \qquad \forall(l) \tag{5.76}$$

$$\sum_t \sum_i HAT_{il}^t = hn_l \qquad \forall(l) \tag{5.77}$$

$$\sum_t \sum_l MAT_{il}^t = M * mc_t \qquad \forall(t) \tag{5.78}$$

$$\sum_t \sum_l HAT_{il}^t = M * hc_t \qquad \forall(t) \tag{5.79}$$

$$\frac{MAT_{il}^t}{mc_t} - AT_{il}^t \leq 0 \qquad \forall(t,i,l) \tag{5.80}$$

$$\frac{HAT_{il}^t}{hc_t} - AT_{il}^t \leq 0 \qquad \forall(t,i,l) \tag{5.81}$$

$$SORTY_{il}^{t} \geq AT_{il}^{t} \qquad \forall (t,i,l) \tag{5.82}$$

$$SORTY_{il}^{t} \leq AT_{il}^{t} + 1 \qquad \forall (t,i,l) \tag{5.83}$$

$$SORTY_{il}^{t} - AT_{il}^{t} \leq 0.999 + M * BI_{il}^{t} \qquad \forall (t,i,l) \tag{5.84}$$

$$AT_{il}^{t} - SORTY_{il}^{t} \leq 0.999 + M * (1 - BI_{il}^{t}) \qquad \forall (t,i,l) \tag{5.85}$$

$$MAT_{il}^{t} \leq M * X_{li}^{t} \qquad \forall (t,i) \tag{5.86}$$

$$HAT_{il}^{t} \leq M * X_{li}^{t} \qquad \forall (t,i) \tag{5.87}$$

$$\sum_{l} SORTY_{il}^{t} - nh_{t}^{i} \leq M * BIII_{t}^{i} \qquad \forall (t,i) \tag{5.88}$$

$$-\sum_{l} SORTY_{il}^{t} + nh_{t}^{i} \leq M * \left(1 - BIII_{t}^{i}\right) \qquad \forall (t,i) \tag{5.89}$$

$$SORTY_{il}^{t} - X_{il}^{t} \leq M * BIII_{t}^{i} \qquad \forall (t,i) \tag{5.90}$$

$$-SORTY_{il}^{t} + X_{il}^{t} \leq M * BIII_{t}^{i} \qquad \forall (t,i) \tag{5.91}$$

$$nh_{t}^{i} - \sum_{l} X_{il}^{t} \leq M * \left(1 - BIII_{t}^{i}\right) \qquad \forall (t,i) \tag{5.92}$$

$$-nh_{t}^{i} + \sum_{l} X_{il}^{t} \leq M * \left(1 - BIII_{t}^{i}\right) \qquad \forall (t,i) \tag{5.93}$$

$$X_{il}^{t} \leq SORTY_{il}^{t} \qquad \forall (t,i) \tag{5.94}$$

$$TNS = \sum_{t} \sum_{i} \sum_{l} SORTY_{il}^{t} \tag{5.95}$$

$$TFD = \sum_{t} \sum_{i} \sum_{l} \sum_{j} \left(X_{il}^{t} * d_{ij} * in_{l}^{j} + SORTY_{il}^{t} * d_{ij} * in_{l}^{i} * fn_{l}^{j} + X_{il}^{t} * d_{ij} * fn_{l}^{j} \right) \tag{5.96}$$

Sets

i, j	number of nodes
t	number types of helicopters
l	number of NARATs

Decision Variables

X_{li}^{t}	number of helicopters of type t with initial node i assigned to NARAT l

$SORTY_{li}^{t}$	number of sorties for NARAT *l* that will be done by helicopters of type *t* with initial node i
MAT_{li}^{t}	the material transfer amount to be done for NARAT *l* by helicopters of type *t* with initial node i
HAT_{li}^{t}	the human transfer amount to be done for NARAT *l* by helicopters of type *t* with initial node i
AT_{li}^{t}	the maximum of material and human transfer sorties to be done for NARAT *l* by helicopters of type *t* with initial node i
TNS	total number of sorties
TFD	total flight distance
BI_{il}^{t}, BII_{i}^{t}	binary variables for extended formulation

Parameters

d_{ij}	distance between location i and j
mc_t	material cargo capacity of helicopter of type t
hc_t	human transport capacity of helicopter of type t
nh_t^i	number of available helicopters of type at node i
in_l^i	initial node of NARAT l
fn_l^i	final node of NARAT l
mn_l	material load in NARAT l
hn_l	human load in NARAT l

5.4.3.1 Objective Function

Two alternatives are considered as the objective function for the formulation in (5.75). Minimizing the total number of sorties, which is calculated in (5.95), can be used if the main consideration is to minimize the total fixed cost of sorties. In addition, minimizing the total flight distance is a possible objective when the main consideration is to minimize the operation duration or variable cost of flights. The calculation of the total distance given in (5.96) requires the summation of distance of sorties and the distance between initial location of helicopters and initial – final locations of NARATs.

5.4.3.2 NARAT Requirement Constraints

The material and human transportation requirements of the NARATs will be satisfied by the constraints (5.76) and (5.77), respectively. These equations ensure that the total material and human transfer initialized from the initial node of each NARAT equals to the material and human transportation amounts of each NARAT.

5.4.3.3 Helicopter Assignment Constraints

The constraints (5.78) and (5.79) provide the assignment of material and human transfer only to the helicopters that have related capacities. By evaluating AT_{li}^{t}, the maximum of number of material and human transfer sorties is found in (5.80) and (5.81). These values are assigned to the integer variable $SORTY_{il}^{t}$ by (5.82) – (5.85). (5.82) and (5.83) provide rounding these assignments up to the nearest the integer, where using the binary variable in (5.84) and (5.85). BI_{il}^{t} prevents forthcoming problems in case AT_{li}^{t} and $SORTY_{il}^{t}$ turn out to be equal.

The necessity of assigning helicopters to material or human transportation is represented in (5.86) and (5.87). The condition expressed by the constraints (5.88) – (5.93) guarantee to utilize all the available helicopters before using a helicopter for two sorties. The constraints (5.88) and (5.89) help to determine the maximum of sum of number of sorties by each type of helicopter from each initial point and the number of helicopters of that type at their initial points. If the number of sorties is less than the number of helicopters, (5.90) and (5.91) provide that the number of helicopters used is equal to the number of sorties. Else, (5.92) and (5.93) utilizes all the helicopters. In addition, the constraint (5.94) prevents more helicopters than the number of sorties.

5.4.3.4 Dimensionality of the Model

$$\text{Number of Variables} = 6*i*t*l + i*t \tag{5.97}$$

$$\text{Number of Constraints} = 6*i*t*l + 9*i*t + 2*l + 2*t \tag{5.98}$$

5.5 SOLUTION PROCEDURES

Due to the complexity of the problem, it is unrealistic to try to solve such a problem in a real-life case with an MIP model. Not only the time constraint in case of emergency, but also the cost of handling a large MIP model requires a fast and efficient algorithm. Some commercial optimizers are tested on this problem like CPLEX. However, because of the size of the problem it takes unacceptably long computation times. Especially, in real life cases with large number of locations and helicopters, these commercial optimizers are insufficient to respond in a fast manner. Therefore, it is necessary to search some other solution procedures to solve this problem. Heuristic algorithms are favorable in solving vehicle routing problems. In the context of this study, several heuristic approaches are proposed below:

5.5.1 Input Parameters of the Model

A single pass heuristic algorithm is developed to be used later as the first step of a possible metaheuristic algorithm as forming an initial solution if desired.

The general structure of the algorithm is based on selecting the flight leg for helicopters, which initially stay on operation bases. The algorithm starts with helicopter selection from the pool of available helicopters. Then, the demand node, which is going to be visited, is selected from the set of unassigned demand nodes. In this assignment, the feasibility checks are considered. In case feasibility cannot be achieved by the node under consideration, the next unvisited demand node will be selected. After deciding on an assignment, the conditions are updated and the algorithm starts from the helicopter selection step again. This loop is run until all demand nodes are reached or it is understood that some of them cannot be assigned in this iteration.

If there are still unassigned demand nodes, it will be checked whether these demand nodes can be reached from operation bases, which have no helicopters at that time. If this occurs, first an available helicopter is assigned to fly to the related operation base, and then it is going to reach the demand node under consideration. At the end of the algorithm, all helicopters are forced to finish their routes on one of the operation bases.

5.5.1.1 Notation

$cobd_d$: The closest operation base to demand node d

mfa_d^t : Minimum fuel needed for helicopter t returning from demand node d

H : Set of helicopters

AH : Set of available helicopters

D : Set of demand nodes

UAD : Set of unassigned demand nodes

TUD : Set of unassigned demand nodes which have been already checked

cl_t : Current location of helicopter t

5.5.1.2 The Algorithm

Initializations:

$$\text{TUD}=\varnothing \tag{5.99}$$

Step 0: Determine the closest operation base for each demand node ($cobd_d$) and the minimum fuel amount (mfa_d^t) for each demand node by each helicopter.

$$\textit{Go to Step } 1. \tag{5.100}$$

Step 1: Check the availability of helicopters.

$$\begin{aligned} &\textit{If } AH = \varnothing, \\ &\qquad \textit{Go to Step } 7. \end{aligned} \tag{5.101}$$

Step 2: Select the candidate helicopter and label it as t.

$$\textit{Go to Step } 3. \tag{5.102}$$

Step 3: Select the candidate demand node.

$$\begin{aligned} &\textit{If } \; UAD = \varnothing, \textit{ go to Step } 10. \\ &\textit{Else} \\ &\qquad \textit{Label it as } d. \\ &\qquad \textit{Add to } \; TUD. \\ &\qquad \textit{Go to Step } 4. \end{aligned} \tag{5.103}$$

Step 4: Check the feasibility of flying to the candidate demand node.

$$\begin{aligned} & \textit{If it is feasible, go to Step } 5, \\ & \quad \textit{Else} \\ & \quad\quad \textit{If } TUD = AUD, \\ & \quad\quad\quad \textit{Go to Step } 2. \\ & \quad\quad \textit{Else,} \\ & \quad\quad\quad AH = AH - \{t\} \\ & \quad\quad\quad \textit{Go to Step } 1. \end{aligned} \tag{5.104}$$

Step 5: Assign the flight of the candidate helicopter to the candidate demand node.

$$\begin{aligned} & UAD = UAD - \{d\} \\ & AH = H \\ & \textit{Go to Step } 1. \end{aligned} \tag{5.105}$$

Step 6: Check whether the candidate helicopter is able to fly to another demand node.

$$\begin{aligned} & \textit{If it is possible, go to Step } 1. \\ & \quad \textit{Else} \\ & \quad\quad \textit{Force helicopter } t \text{ to } \textit{turn to.} \\ & \quad\quad \textit{Go to Step } 1. \end{aligned} \tag{5.106}$$

Step 7: Check whether the unassigned demand nodes can be reached from the operation bases, which have no helicopters.

$$\begin{aligned} & \textit{If at least one flight is possible, go to Step } 8. \\ & \quad \textit{Else} \\ & \quad\quad \textit{Go to Step } 10. \end{aligned} \tag{5.107}$$

Step 8: Select a candidate helicopter to go to the operation base, which can reach the unassigned demand node.

$$\textit{Go to Step } 9. \tag{5.108}$$

Step 9: Assign the helicopter to the unassigned demand node through that operation.

$$\begin{aligned} & UAD = UAD - \{d\} \\ & AH = H \\ & \textit{Go to Step } 1. \end{aligned} \tag{5.109}$$

Step 10: Make all the helicopters turn back to an operation base.

Before starting to run the algorithm, some calculations are done in Step 0. In this step, for each element of the set of demand nodes, $D = \{d_i\}$, the closest operation base ($cobd_d$) and the fuel amount to fly the distance between the considered demand node and its closest operation base (mfa_d^t) are determined. This calculation is going to be useful in considering the sufficiency of fuel amount to go back to an operation base while deciding on a flight to a demand node.

In Step 1, it is checked if the set of available helicopters (AH) is empty. If the set AH is empty, the algorithm will go to Step 7. Otherwise, in Step 2 the candidate helicopter to be assigned is selected and labeled as t. Step 3 determines the candidate demand node from the set of unassigned demand nodes (UAD). The candidate demand node is labeled as d. The helicopter and the demand node selections are described in detail below. After determining the helicopter and the target demand node, some feasibility checks are required to prove the practicability of the flight of t from the current location of helicopter t (cl_t) to d. The details of the feasibility checks are clarified below. If the feasibility check is not positive, the algorithm goes back to Step 3, which is demand node selection. If none of unassigned demand nodes satisfies the feasibility check, the helicopter t is marked as unavailable and removed from AH. In case of marking all helicopters unavailable, the algorithm will go to Step 7.

Table 5.3: Pseudo Code for the Single Pass Algorithm

```
Choose the helicopter selection method, hsm;
Choose the demand node selection method, dsm;
While (UAD ≠ ∅){
    if (AH ≠ ∅), then {
        Select h from AH by hsm;
        while (UAD ≠ TUD){
            Select d from UAD - TUD by dsm;
            if the flight of h to d is feasible, then {
                Force h from cl_h fly to d;
                cl_h = d;
                Update capacity usage of h;
                AH = H;
                continue;
            }
            else {
                AH = AH + {h};
                TUD = TUD - {d};
            }
        }
    }
    else {
        if a fly is feasible to an element of UAD, then {
            Select d from UAD by dsm;
            Select h from H that can fly to d;
            Force h fly from cl_h to cobd_d;
            Force h fly from cobd_d to d;
            cl_h = d;
            Update capacity usage of h;
            AH = H;
        }
}
```

If a feasibility check is satisfactory, some assignments are done in Step 5. The fuel amount of helicopter t is reduced by the fuel consumption of the flight of t from cl_t to d. The material demand, personnel demand, and human evacuation requirements load of demand node d will be added to the total starting material amount, starting human load, and human evacuation, respectively. In addition, the total starting load of helicopter t is updated as the total of material and personnel demand amount of demand node d and initial fuel amount of helicopter t. The cl_t is updated as d. Demand node d is removed from UAD. Finally, all helicopters are marked as available after each assignment. If UAD becomes empty set, the algorithm goes to Step 10.

Otherwise, Step 6 is initiated and the practicability of another flight for helicopter t in this route is checked. If the result is negative, the helicopter will finish its route in the closest operation base to d. After Step 6, the algorithm goes to Step 1 again.

If the set of available helicopters (AH) is empty in Step 1, the algorithm directly proceeds to Step 7. There is a check about the possibility of reaching the unassigned demand nodes from operation bases, which have no helicopters. Helicopters that can fly through an operation base to an unassigned demand node are listed in this step. The feasibility checks are also considered in this listing. If this list is empty, the algorithm goes to Step 10 by leaving some demand nodes unassigned. In Step 8, a helicopter is selected from the list according to the objective of minimizing the distance traveled to reach an unassigned demand node. After selecting the helicopter and operation base, the assignment of helicopter to the demand node through the related operation base is done in Step 9. Like the procedure in Step 5, the cl_t is updated as d, demand node d is removed from UAD and all helicopters are marked as available after each assignment. The algorithm goes back to Step 1 to start another iteration.

When Step 10 is reached, either all demand nodes will have been reached or there will be some unassigned demand nodes, which can never be reached in this configuration. In Step 10, all helicopters are made to turn to the closest operation bases and complete their routes.

5.5.1.3 Helicopter Selection Methods

In Step 2, a helicopter selection method is needed so that it will measure the distance between the current location of helicopters and unassigned demand nodes and the demand amount unassigned demand nodes. There are three methods considered in this thesis.

Min max distance: In this method, the farthest unassigned demand node to each helicopter is found and the helicopter that gives the minimum of this maximum value is selected.

Min max distance over demand: For each helicopter and unassigned demand node, the ratio of the distance between helicopter and demand node to the total demand amount of demand node is calculated and the maximum of this ratio is selected for each helicopter. The helicopter that gives the minimum of this maximum value is selected.

Min total distance over demand: For each helicopter, the ratios of the distance between the helicopter and demand node over the total demand amount of demand node for all unassigned demand nodes are summed, and the helicopter that gives the minimum of this summation is selected.

5.5.1.4 Demand Node Selection Methods

The demand node to be visited is selected in Step 3 by four different methods. These methods depend on two factors: the distance between the helicopters and the unassigned demand nodes, and their demand.

The farthest demand node: The farthest demand node to the selected helicopter is chosen in this method.

The closest demand node: The closest demand node to the selected helicopter is chosen in this method.

The demand node with the highest demand requirement: The demand node with the highest demand requirement is selected in this method.

The demand node with the lowest demand requirement: The demand node with the lowest demand requirement is selected in this method.

5.5.1.5 Feasibility Checks

Feasibility checks are done for three constraints: Fuel feasibility, helicopter capacity, and supply availability.

5.5.1.6 Fuel Feasibility Checks

According to the assumptions of the problem, each helicopter has a constant fuel consumption amount and a given fuel tank capacity. Fuel feasibility implies that the helicopters have sufficient amount of fuel to start a route, visit demand nodes and reach to an operation base, which have refueling service. In order to satisfy this feasibility, in Step 0 for each demand node, the closest operation base ($cobd_i$) and the fuel amount to fly the distance between the considered demand node and its closest operation base (mfa_i) are determined. Fuel feasibility is ensured by checking if the helicopter to can reach operation base after visiting demand node, say d, by comparing the fuel amount in its tank and mfa_i. If the fuel amount in its tank is not less than mfa_i, the helicopter is allowed to visit demand d on its route.

5.5.1.7 Helicopter Capacity Checks

Since helicopters have strict cargo capacity constraints, assigning a helicopter to a demand node implies capacities checks. Material and human transportation capacities are checked by summing all requirements of demand nodes on the route and comparing this value by the cargo capacity of the helicopter. Since helicopters start their routes by transporting the sum of all requirements of the demand nodes on their routes and finish their routes by transporting the sum of all human evacuation requirements, these comparisons are needed.

5.5.1.8 Supply Feasibility Checks

Since the operation bases have limited amount of material and personnel supplies, every assignment requires checking whether the total supply amount of the starting operation base of the route under consideration is exceeded or not.

5.5.2 Hybrid Algorithm

Metaheuristic algorithms are very successful in solving vehicle routing problems. For this problem, a metaheuristic algorithm, which is a combination of tabu search and simulated annealing is proposed. It is aimed to utilize the inherent capability of both algorithms.

The algorithm starts with an initial solution generation mechanism. This mechanism involves solving the aggregated version of the mixed integer programming and disaggregating again. After an initial solution is produced, the neighborhood of this solution is searched by the hybrid algorithm.

The hybrid algorithm is based on the simulated annealing algorithm supported by tabu search where a tabu list is used in the decision of accepting the neighbor solutions in the simulated annealing flow. The aspiration criteria for tabu search is defined as accepting the best ever met before. The detailed flow of the simulation algorithm is given below.

STEP 1 – INITIAL SOLUTION GENERATION

1.1 *Input*

number of locations;
number_of_helicopters;
initial_location_of_helicopter $\forall\ t$
C(t); // Capacity of Helicopter $\forall\ t$

1.2 *Initialization*

$i \leftarrow 0$;
$t \leftarrow 1$; *// t is the number of helicopters*
$t^* \leftarrow 1$;
$O_{(i)} \leftarrow \emptyset$; *// the set of helicopters in location i*
$\forall\ i \quad AC_{(t^*(i))} \leftarrow 0$; *// aggregated capacity of helicopter t in location i*

1.3 *Helicopter Aggregation*

While(t < number of helicopters)
  do begin
    i = initial_location_of_helicopter(t)
    $O_{(i)} = O_{(i)} \cup \{t\}$;
    $t \leftarrow t+1$;
  end;
While(i < number of locations)
  do begin
    if($O_{(i)} \neq \emptyset$)
      then begin
        initial_location_of_helicopter(t) = i;*
        $AC(t^*) = \sum_{t \in O(i)} C(t)$;
        $t^* \leftarrow t^*+1$;
      end;
  end;
number_of_helicopters = t*;*

1.4 Solve The Problem in the MIP Model for number_of_helicopters*, t*, initial_location_of_helicopter(t*), AC(t*)

1.5 Helicopter Disaggregation

i ← 1;
While (i < number_of_helicopters)*
do begin
i ← initial_location_of_helicopter t;*
run the minimum number of helicopter selection algorithm for $O_{(i)}$;
assign the route of t to the selected helicopters;
end;

1.6 Output

Routes, Load and Unload decisions of helicopters

STEP 2 – HYBRID ALGORITHM

2.1 Input

X_i ; // the initial solution generated in Step 2.

2.2 Initialization

$T \leftarrow T_0$; // the initial temperature
$L \leftarrow L_T$; // the initial search size
$X^* \leftarrow X_i$; // the best solution is initialized
$X_C \leftarrow X_i$; // the current solution is initialized
$X_N \leftarrow X_i$; // the neighbor solution is initialized
tabu_contr ← 0; // tabu counter

2.3 Neighborhood Search

```
while (!STOP)
    do begin
        while (L< L_T)
            do begin
                X_N ← A neighbor solution of X_C;
                while (X_N ∈ tabu_list)
                    do begin
                        if (tabu_contr = max_tabu_search)
                        then begin
                            STOP;
                        end;
                        if (X* > X_N)
                        then begin
                            X_N ← A neighbor sol. of X_C;
```

```
                                        tabu_contr ← tabu_contr+ 1;
                                end;
                                else begin //aspiration
                                        X_C ← X_i;
                                        tabu_list ← tabu_list\{X_C};
                                end;
                end;
                if (Obj(X_N) < Obj(X_C))
                        then begin
                                X_C ← X_N;
                        tabu_list ← tabu_list ∪ {X_C};
                                if (Obj(X_N) < Obj(X*))
                                        then begin
                                                X* ← X_N;
                                        end;
                        end;
                        else begin
                                if (e^(-(Obj(X_N) - Obj(X_C))/T) > U)
                                then begin
                                        X_C ← X_N;
                                        tabu_list ← tabu_list ∪ {X_C};
                                end;
                        end;
                L ← L+1;
        end;
        if (stopping_condition_holds)
        then begin
                STOP;
        end;
        T ← α *T;
        L_T ← γ * L_T;
end;
```

2.4 Output

X^*;

5.5.3 Helicopter Aggregation Algorithm

Another way to cope with the insufficiency of commercial optimizers in solving the mixed integer programming model in a reasonable time is to reduce the size of the problem by carrying out aggregations.

Number of helicopters can be very large in most of the real life cases. To reduce the dimensionality of the problem, and thus to enhance the solubility of the model, aggregating the helicopters is an efficient method in solving this MIP problem. An algorithm based upon the principle of aggregation is provided below:

STEP 1 – Input

number of locations;
number_of_helicopters;
initial_location_of_helicopter $\forall\ t$
C(t); // Capacity of Helicopter $\forall\ t$

STEP 2 – Initialization

i ←0;
t ←1; // t is the number of helicopters
t ←1;*
O(i) ←Φ; // the set of helicopters in location i
$\forall\ i$ *AC(* $t^{*}(i)$ *) ←0; // aggregated capacity of helicopter t in location i*

STEP 3 – Helicopter Aggregation

```
While( t < number of helicopters)
        do begin
                O(i) = O(i) ∪{t};
                t ←t+1;
        end;
While( i < number of locations)
        do begin
                if(O(i) ≠ Φ)
                        then begin
                                initial_location_of_helicopter(t*) = i;
                                AC(t*) = Σ_{t∈O(i)} C(t);
                                t* ← t*+1;
                        end;
        end;
number_of_helicopters* = t*;
```

$$AC(t^*) = \sum_{t \in O(i)} C(t)$$

Solve The Problem in the MIP Model for number_of_helicopters*, t* , initial_location_of_helicopter(t*), AC(t*)

STEP 4 – Helicopter Disaggregation

```
i ←1;
While (i < number_of_helicopters*)
        do begin
```

i = initial_location_of_helicopter t;*
run the minimum number of helicopter selection algorithm for O(i);

Output

Routes, Load and Unload decisions of helicopter

5.6 USER INTERFACES OF THE DECISION SUPPORT TOOL

At this stage, RTG SAS-045 attempted to design the user interface of Decision Support Tool (DST) for Helicopter Mission Planning (HELOMIP), which aims to determine the optimum routes for helicopters, load/ unload quantities, refueling activities and load configurations during material and human transfer between locations.

The first part of the HELOMIP interface includes basic information. The user may easily observe the title page of DST HELOMIP showing the main structure of the DST. The user will provide data by clicking on **INPUT DATA** button. The model parameters needed to run the model will be provided by **MODEL** button after the data entry. The output of the model will be presented by **OUTPUT DATA** button.

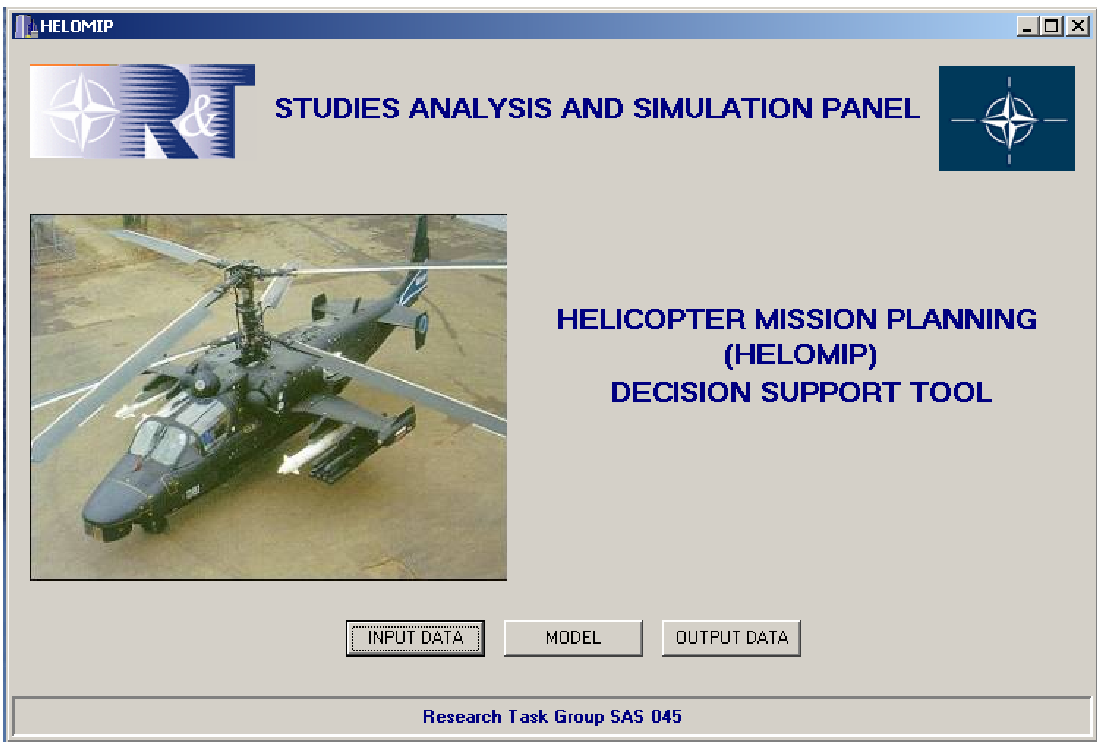


Figure 5.1: HELOMIP Initial Screen.

When the user selects **INPUT DATA**, he or she encounters two different sets of input data. The first button, which is called **NARAT MODEL DATA**, is needed for cases that use NARATs as task requests. The second button, which is called **GENERAL MODEL DATA**, is needed for the general case that can be treated in DST.

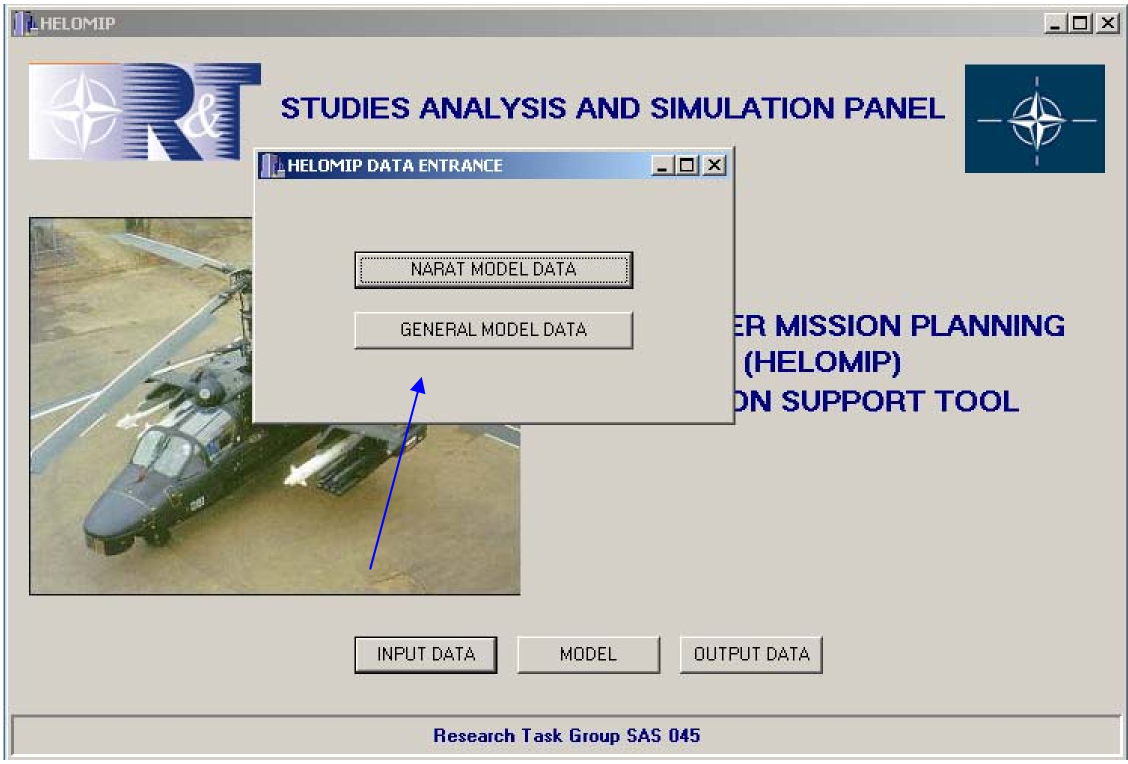


Figure 5.2: HELOMIP Data Entry (1).

The decision maker who selects **NARAT MODEL DATA** button will encounter the following screen. This will cover **NARAT NUMBER**, the authority who has requested the NARAT (**REQUESTED BY**) and information contained in standard NARAT format. These sections are **TRANSPORT OF PERSONNEL**, **TRANSPORT OF MATERIAL**, **TRANSPORT OF CASUALTIES**, **SCHEDULE**, **COMMUNICATIONS AND CONTACTS.**

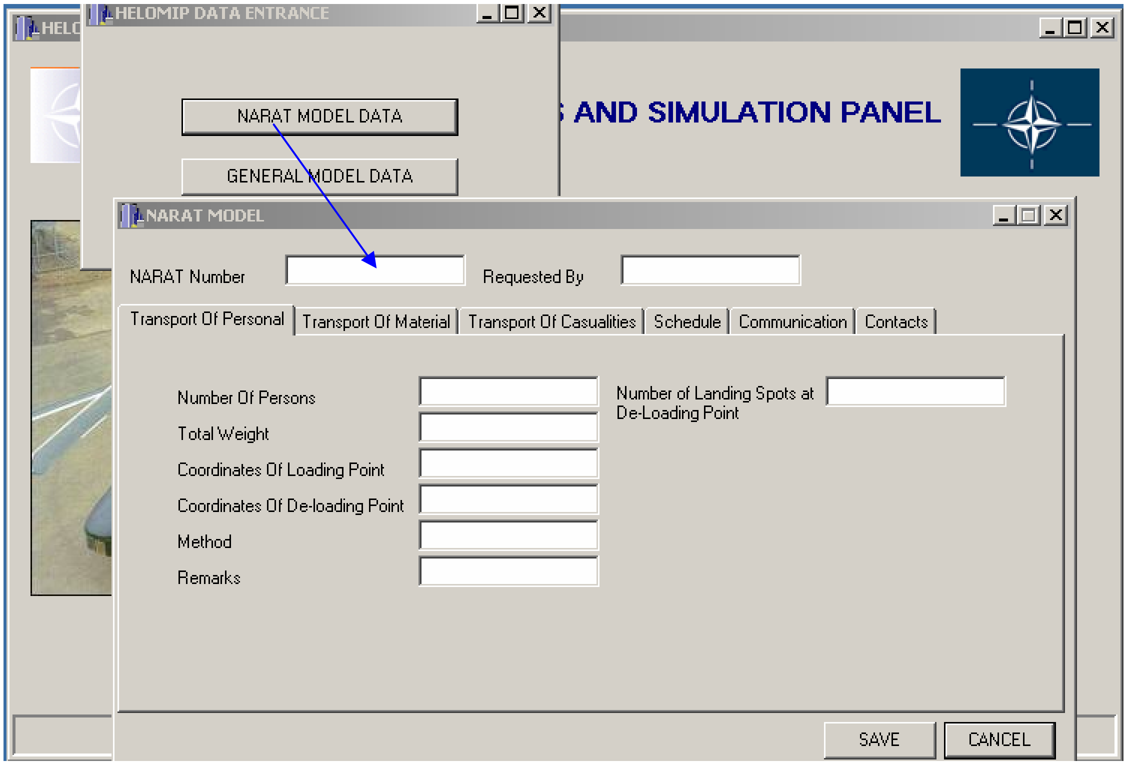


Figure 5.3: HELOMIP Data Entry (2).

TRANSPORT OF PERSONNEL section consists of number of persons, total weight of personnel, coordinates of loading and de-loading points, method, number of landing spots at de-loading point and remarks. The next section in this menu is **TRANSPORT OF MATERIAL** which contains information concerning total weight of material, description and priority of the cargo, weight, length and width of the load, number of material, coordinates of loading and de-loading point, referred delivery method, reason for delivery method, number of landing spots at de-loading point and remarks.

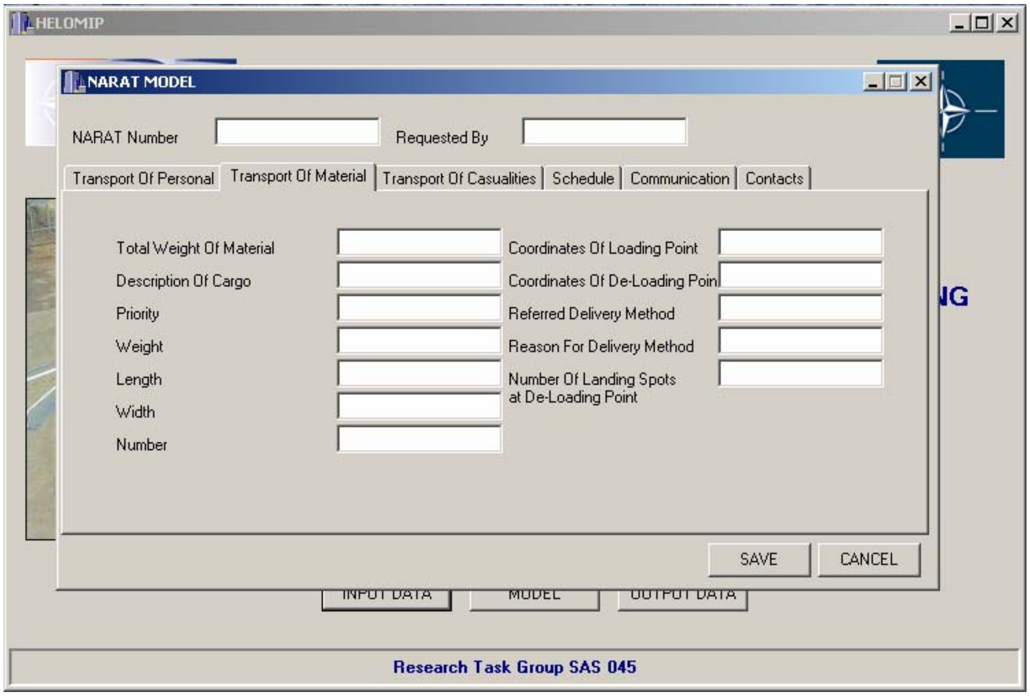


Figure 5.4: HELOMIP Data Entry (3).

SCHEDULE section consists of data about earliest start to move, earliest and latest delivery time at the landing site as it is shown in the following figure.

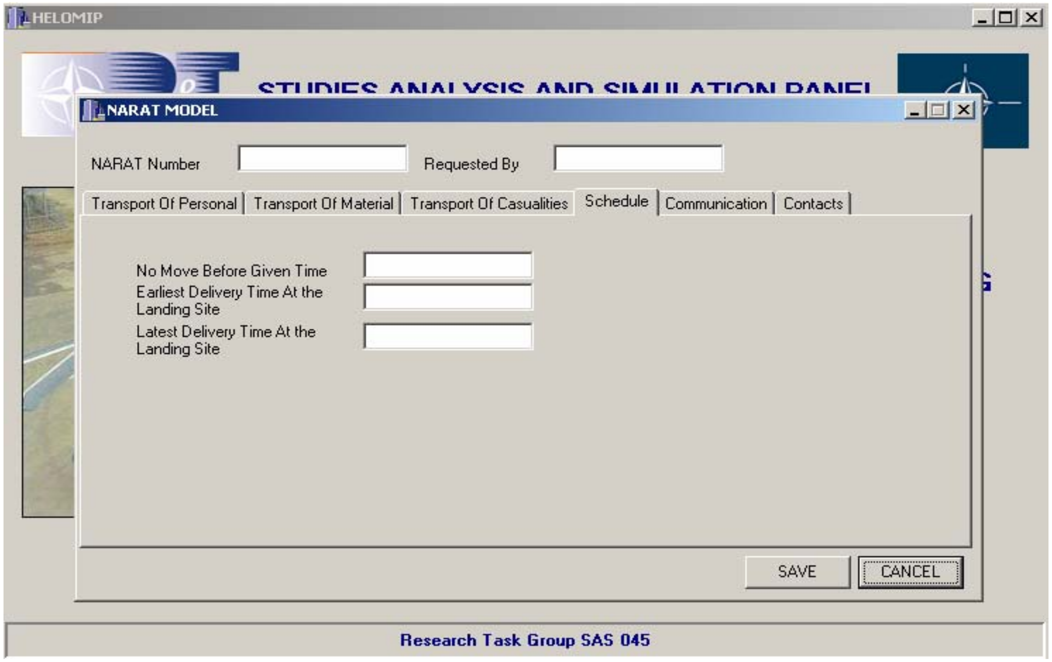


Figure 5.5: HELOMIP Data Entry (4).

Data about unit, call signs, frequencies and mode will be represented under **COMMUNICATION** section.

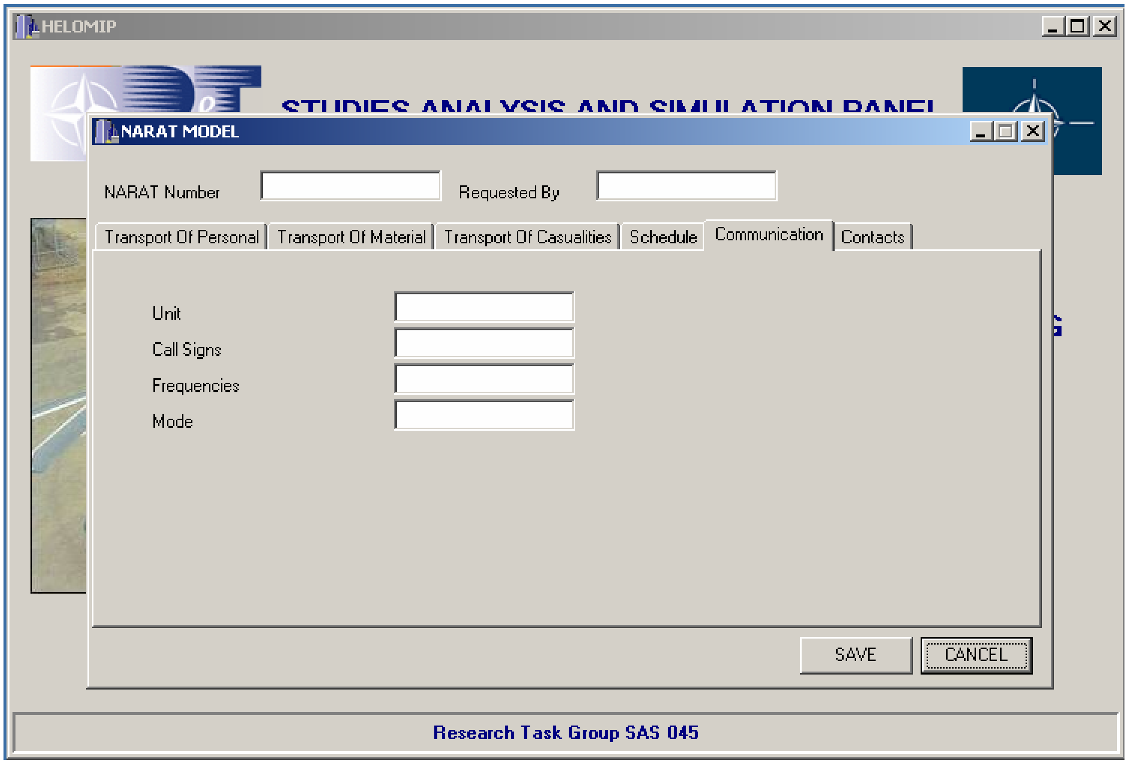


Figure 5.6: HELOMIP Data Entry (5).

CONTACTS will consist of type of the contact, location of the contact, staff information of contacts such as name, rank and phone as it is shown in the following figure.

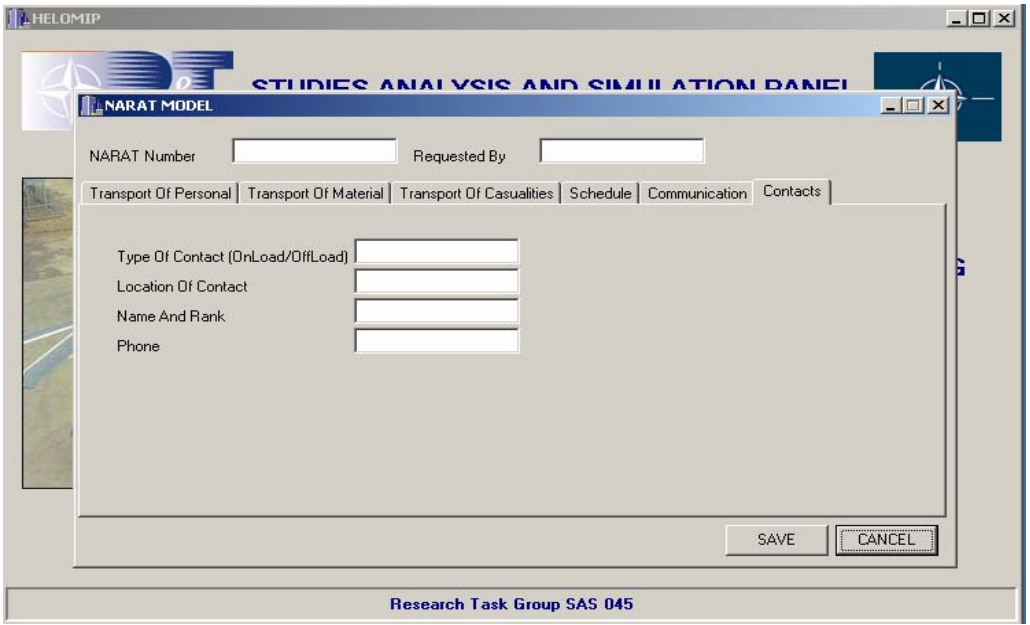


Figure 5.7: HELOMIP Data Entry (6).

The authority who selects **GENERAL MODEL DATA** will encounter the following menu which covers number of supply nodes, number of demand nodes and number of helicopter types. This information will help one to define the mission planning problem. Supply nodes are locations where resources are provided and demand nodes are the places for providing medical evacuation, rescue, etc.

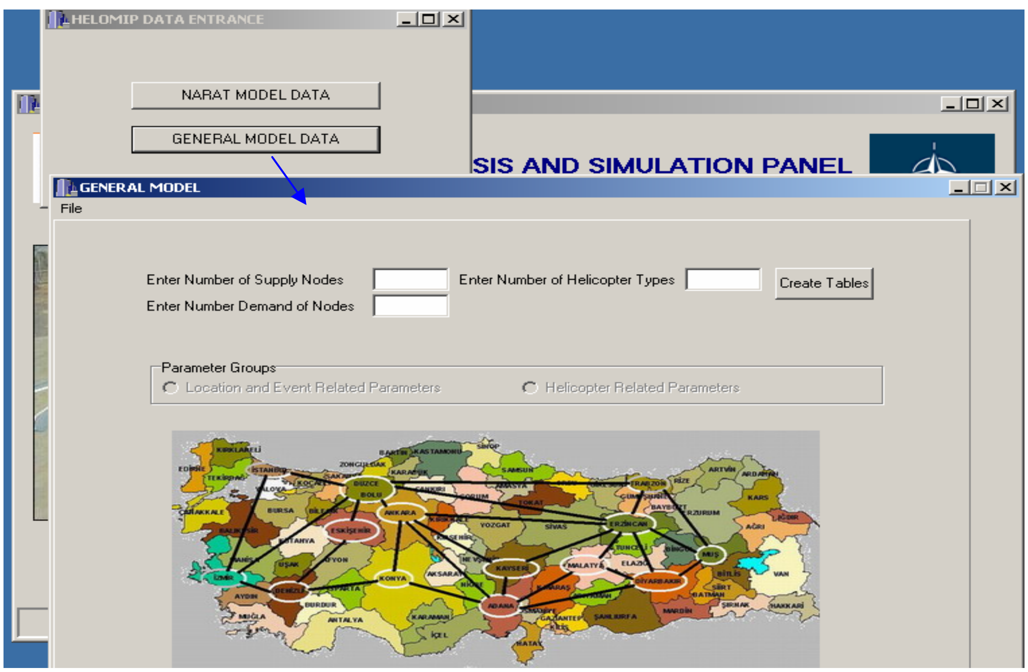


Figure 5.8: HELOMIP Data Entry (7).

Once the number of supply and demand nodes and helicopter type is provided, the user will be guided into an optional input mechanism. One option is Location and Event Related Parameters and the other is Helicopter Related Parameters. In this example, there are 2 supply nodes and 1 demand node. Two types of helicopters will be mobilized.

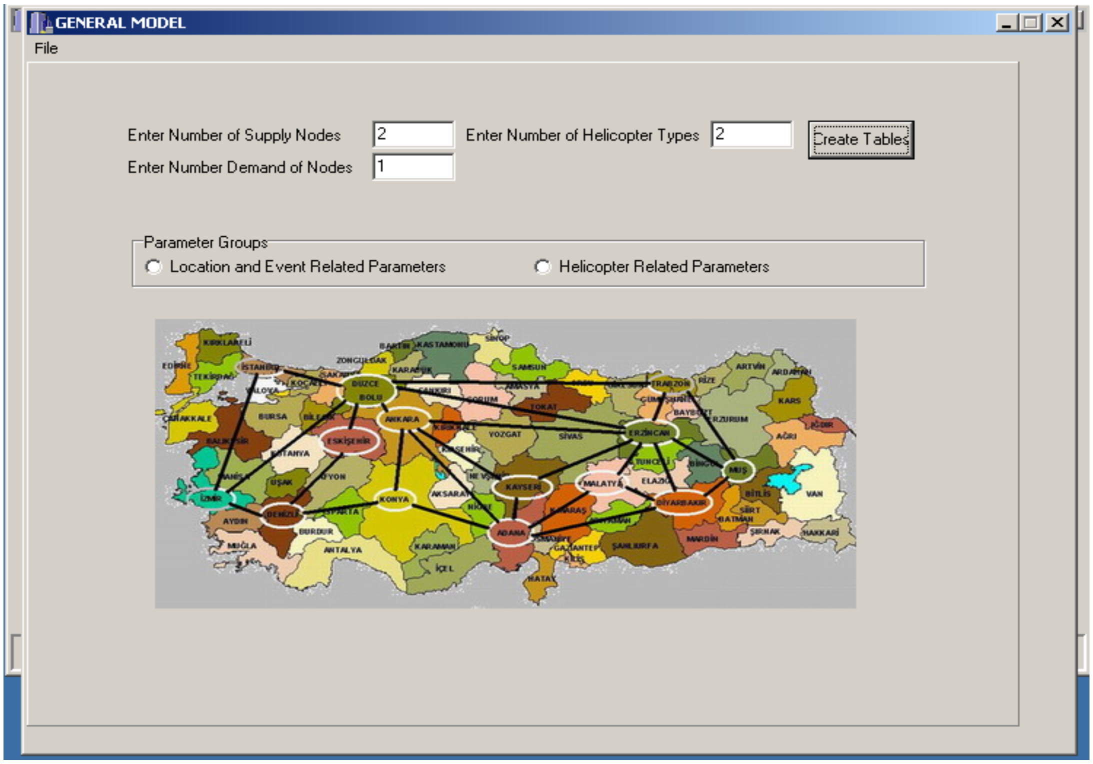


Figure 5.9: HELOMIP Data Entry (8).

After one inputs data for nodes and helicopter types, it is necessary to select **LOCATION AND EVENT RELATED PARAMETERS**. In this table, there are several tables, which represent Coordinates, Distance Matrix, Demand Quantities, Supply Quantities, Hospital Capacities and Refueling Nodes. In **COORDINATES** table, there are three coordinates: the first two belong to supply nodes, the other one belongs to demand nodes. Before proceeding to the next table, one should define the unit of measurement for distances between nodes. Kilometer is already selected in the following example.

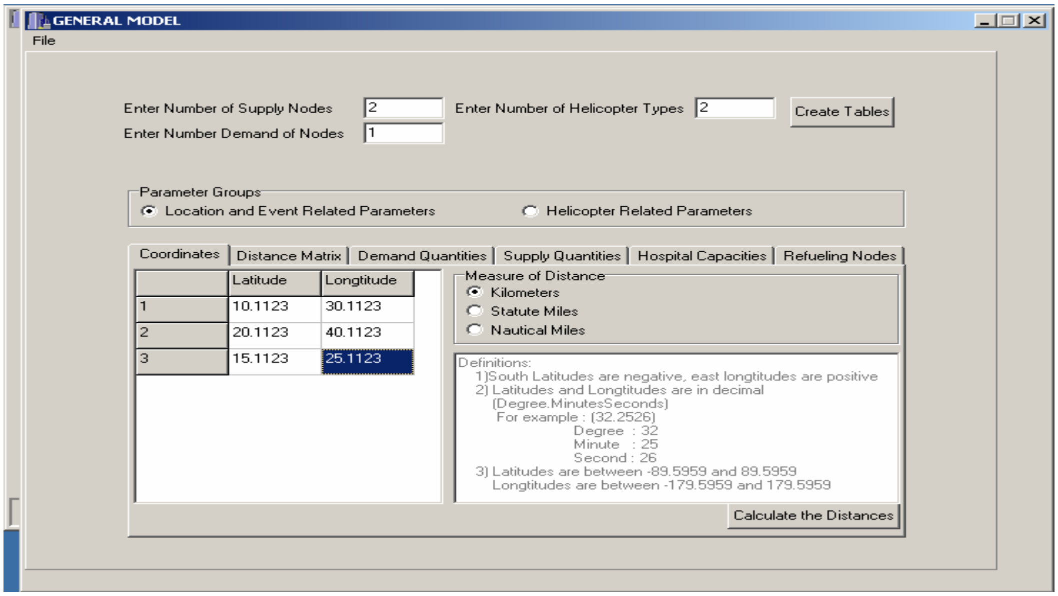


Figure 5.10: HELOMIP Data Entry (9).

When the user selects kilometers, the following **DISTANCE MATRIX** will be shown in the program.

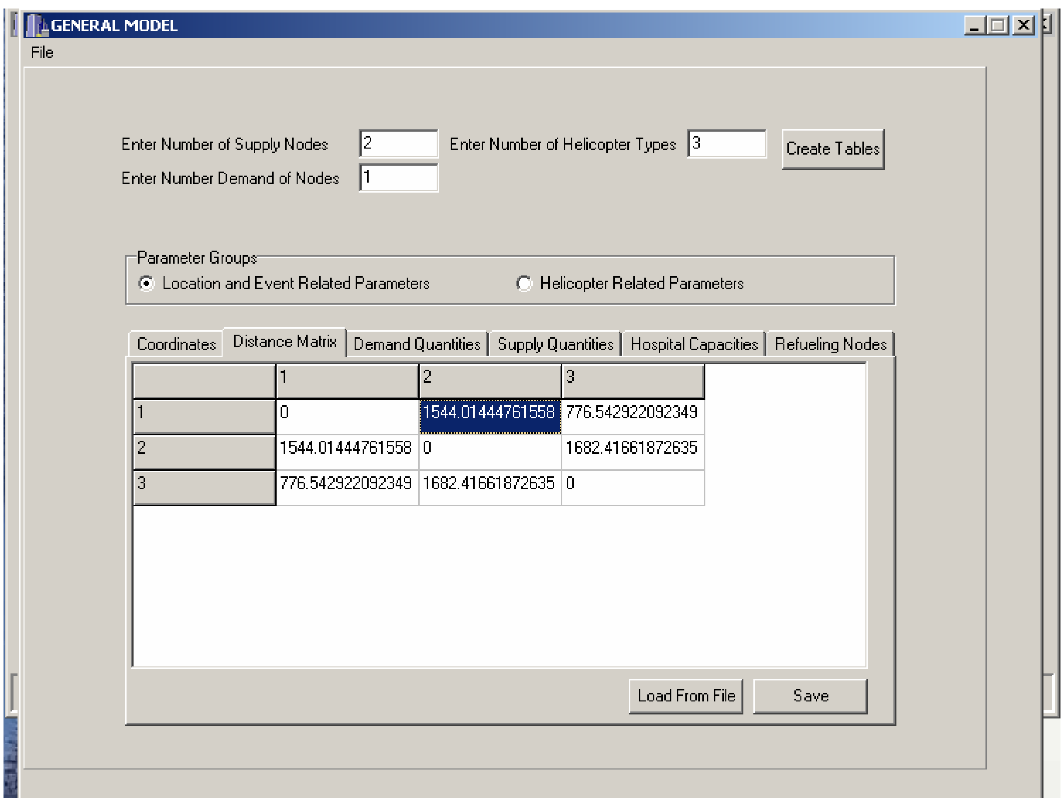


Figure 5.11: HELOMIP Data Entry (10).

Under the **DEMAND QUANTITIES,** the user may be able to see demand quantities in terms of passenger, evacuee and material. The passenger and evacuees will be in numbers. The materials will be in tons. As it can be seen in the following table, only the demand node is the third node, so it is necessary to fill in only the third column of the table.

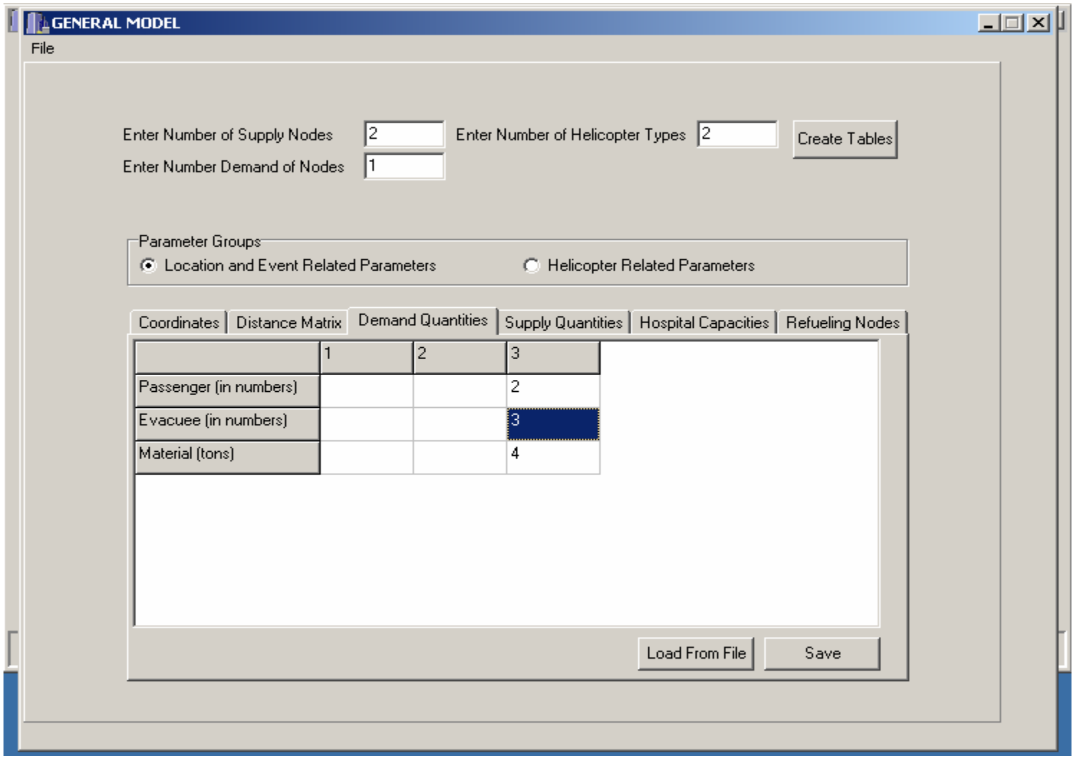


Figure 5.12: HELOMIP Data Entry (11).

SUPPLY QUANTITIES data will be filled in the same manner. The user must provide supply quantities in terms of passenger, evacuee and material. Supply nodes are the first and second nodes that are needed to be completed.

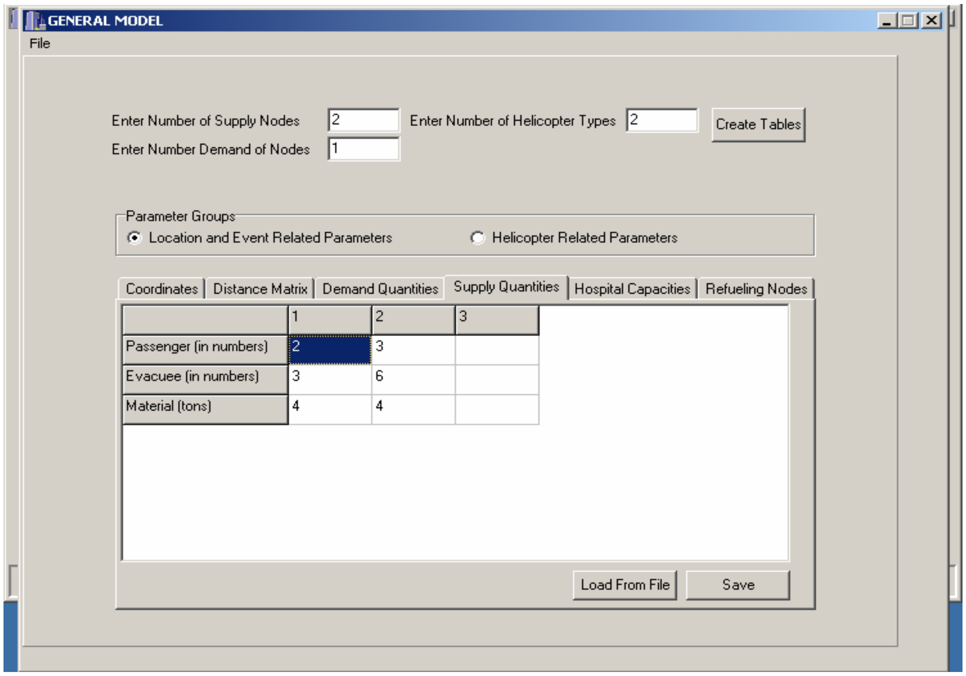


Figure 5.13: HELOMIP Data Entry (12).

Both supply and demand nodes may offer medical service, and accordingly hospital capacities must be provided under **HOSPITAL CAPACITIES**.

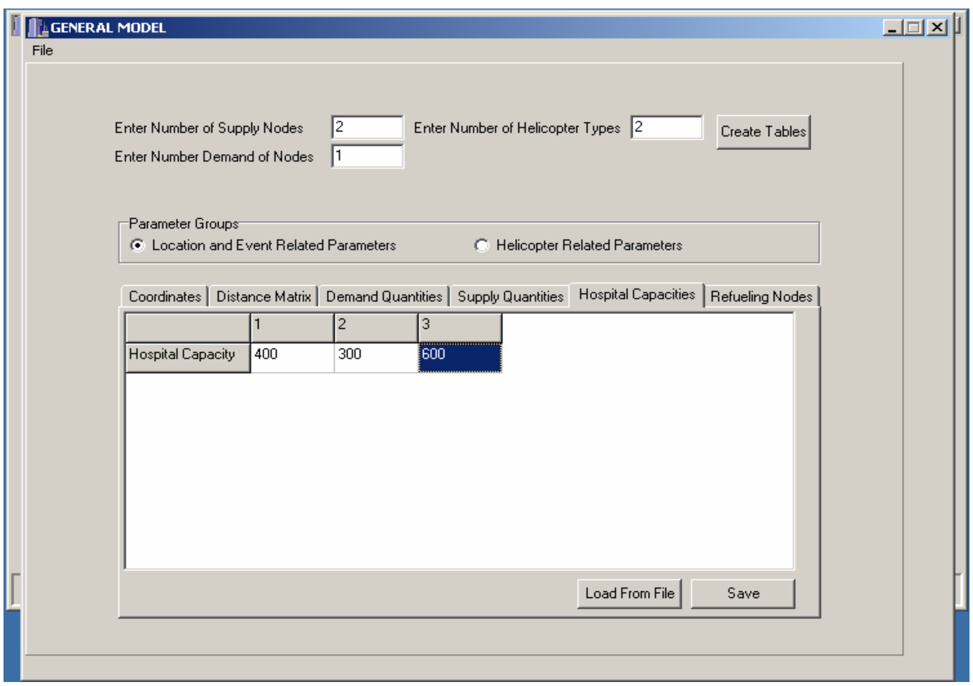


Figure 5.14: HELOMIP Data Entry (13).

Refueling capacities will be shown under **REFUELING NODES**. If a particular node provides refueling possibility, then this should indicated at this point by entering “1” for this node.

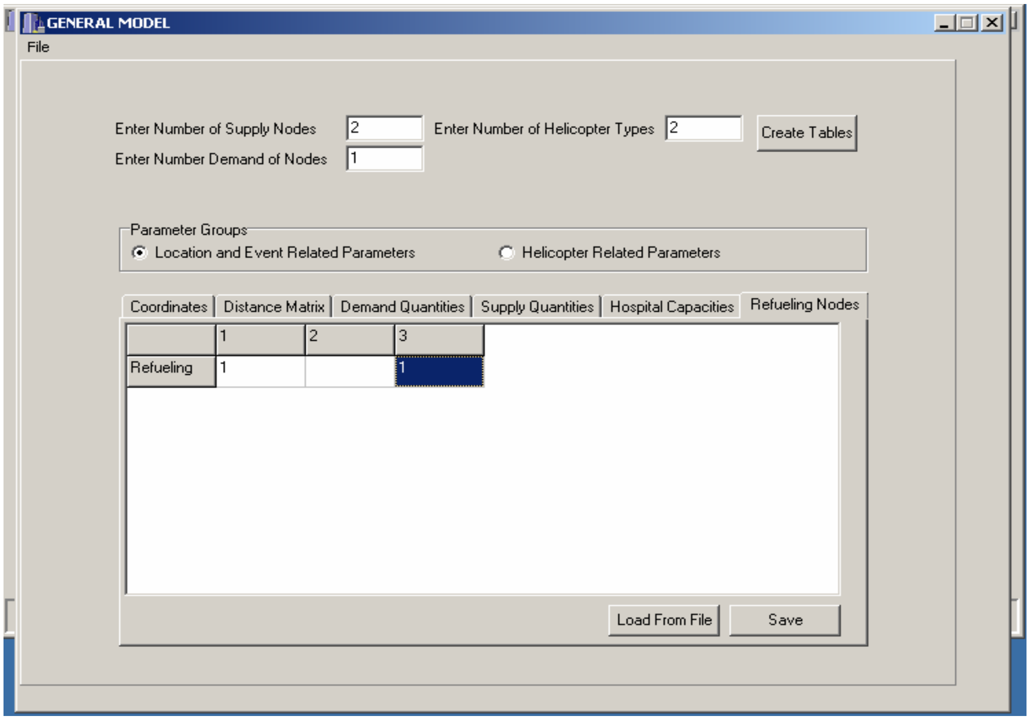


Figure 5.15: HELOMIP Data Entry (14).

To input helicopter related parameters, one needs to click on **HELICOPTER RELATED PARAMETERS.** This main table includes Availability of Nodes, Material and Personal Capacities, Technical Specifications and Load Configuration. There are two types of helicopters in this scenario, and **AVAILABILITY NODES** table shows the number of helicopters in each node.

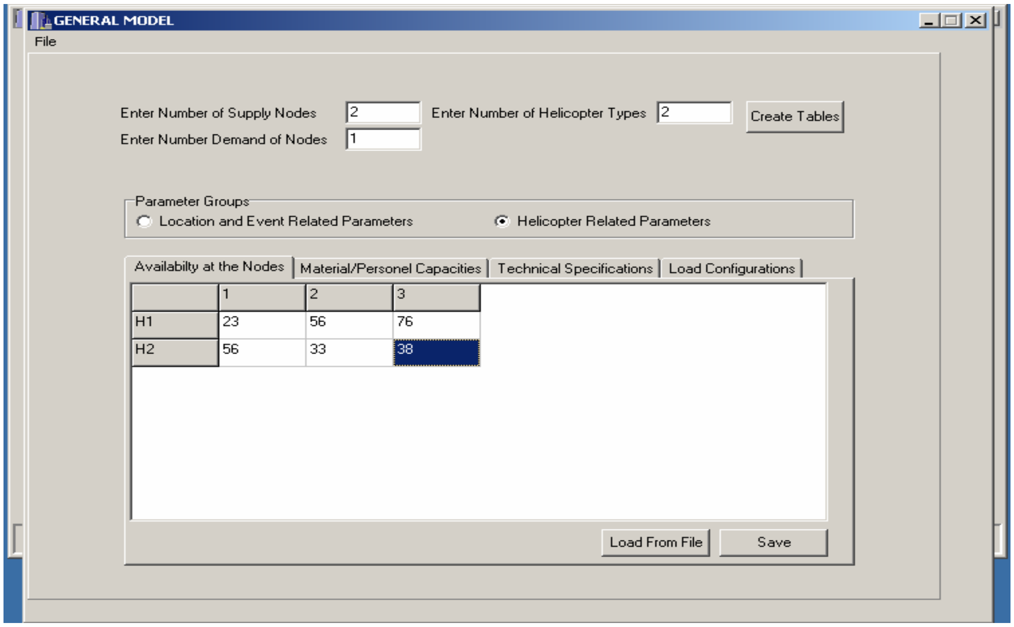


Figure 5.16: HELOMIP Data Entry (15).

The next table will indicate **MATERIAL AND PERSONNEL CAPACITIES** of each helicopter type.

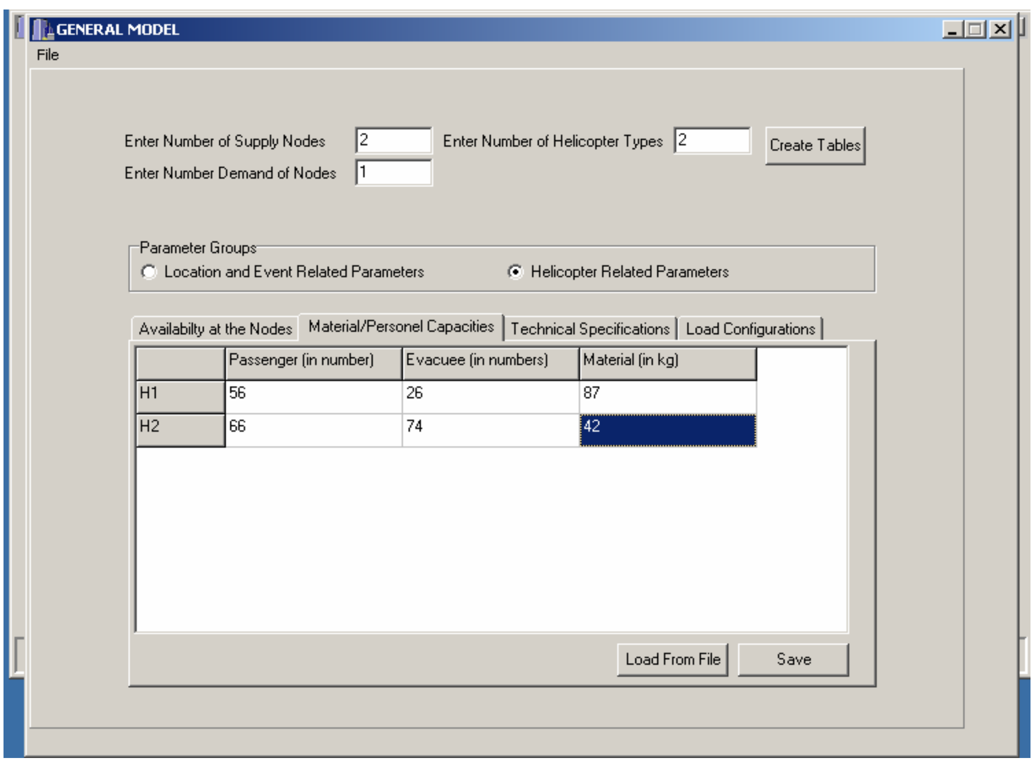


Figure 5.17: HELOMIP Data Entry (16).

TECHNICAL SPECIFICATIONS table will consist of total weight capacity in kg., fuel consumption rate and fuel tank capacities and speed.

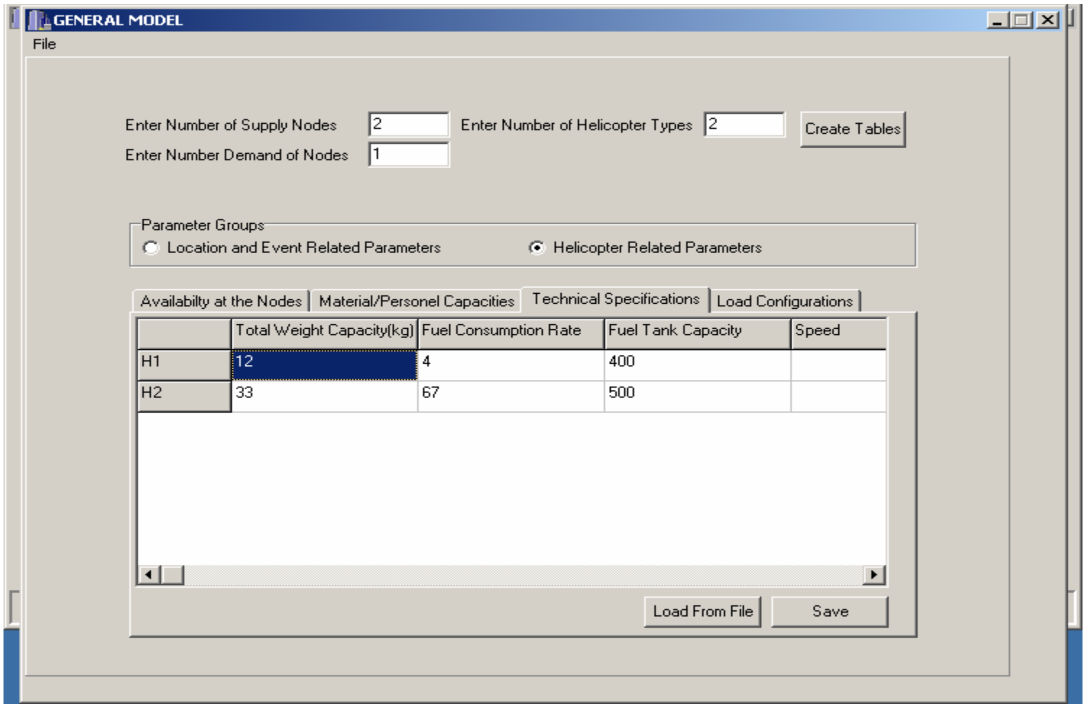


Figure 5.18: HELOMIP Data Entry (17).

After providing the input data, then it is necessary to click on **MODEL** which includes "Get Data File", "Get GAMS File" and "Find GAMS Path". If the user has already entered input data and saved it in a file, by typing the file name in the first box and clicking on "Get Data File", the user will be able to retrieve the previously stored data. The second box is needed to retrieve GAMS file and the third one is to detect the GAMS software. Tehn "Run GAMS" is activated to run the model HELOMIP.

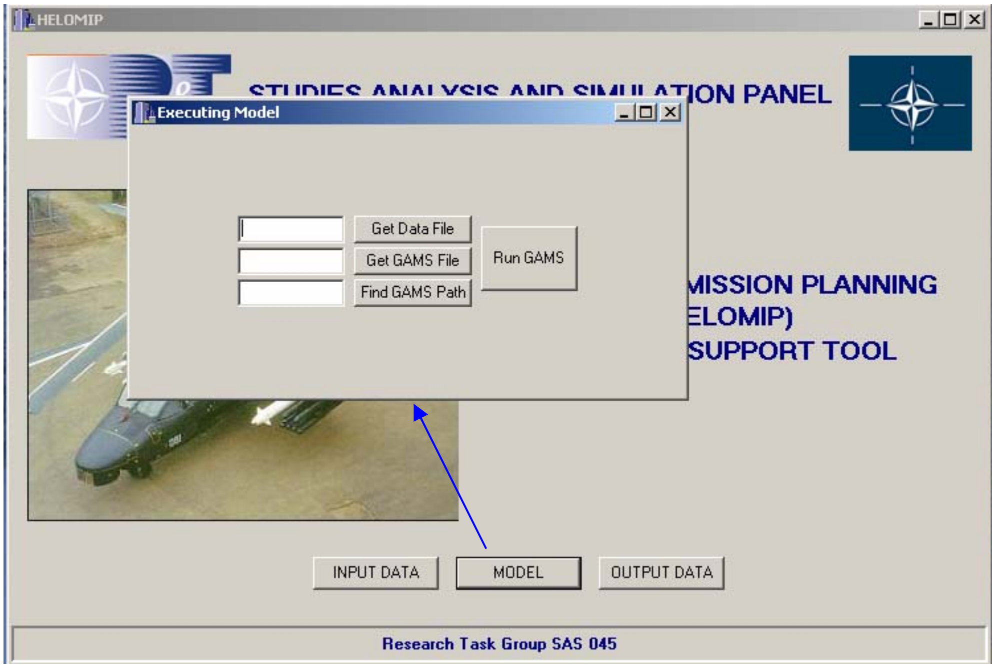


Figure 5.19: HELOMIP Execution Module.

After the user runs the model, the user may obtain results by using **OUTPUT DATA**. It is possible to reach outputs in different formats. One may get the output according to helicopter number, location, load type, helicopter type or NARAT Request.

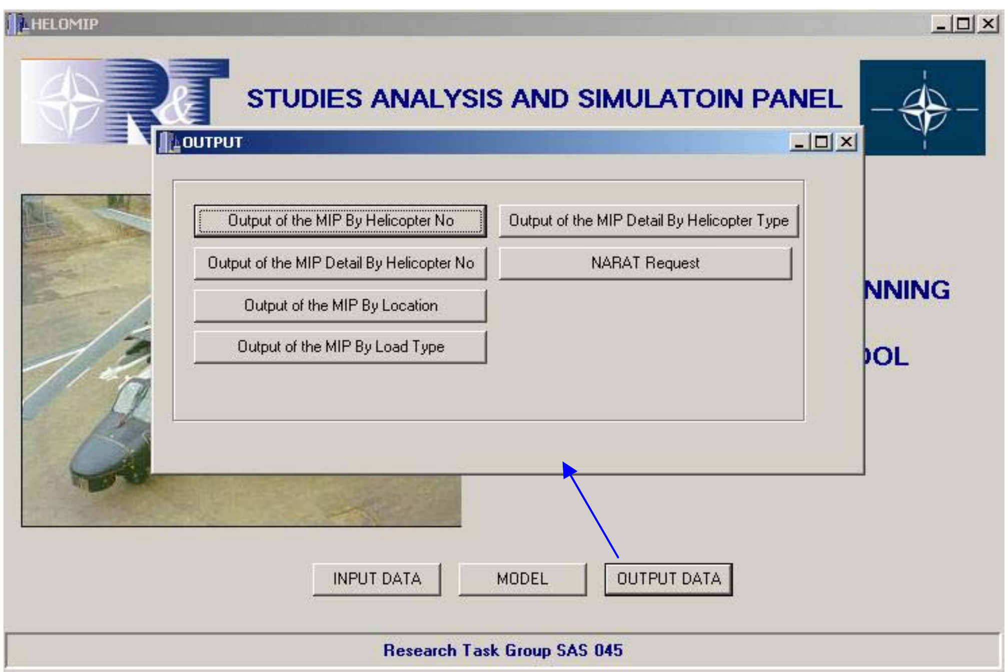


Figure 5.20: HELOMIP Output Display (1).

By clicking on **MIP BY HELICOPTER NUMBER**, the user will be able to see the location and the cargo of the defined helicopter (human, material, casualties-loaded and unloaded) and refueling amount.

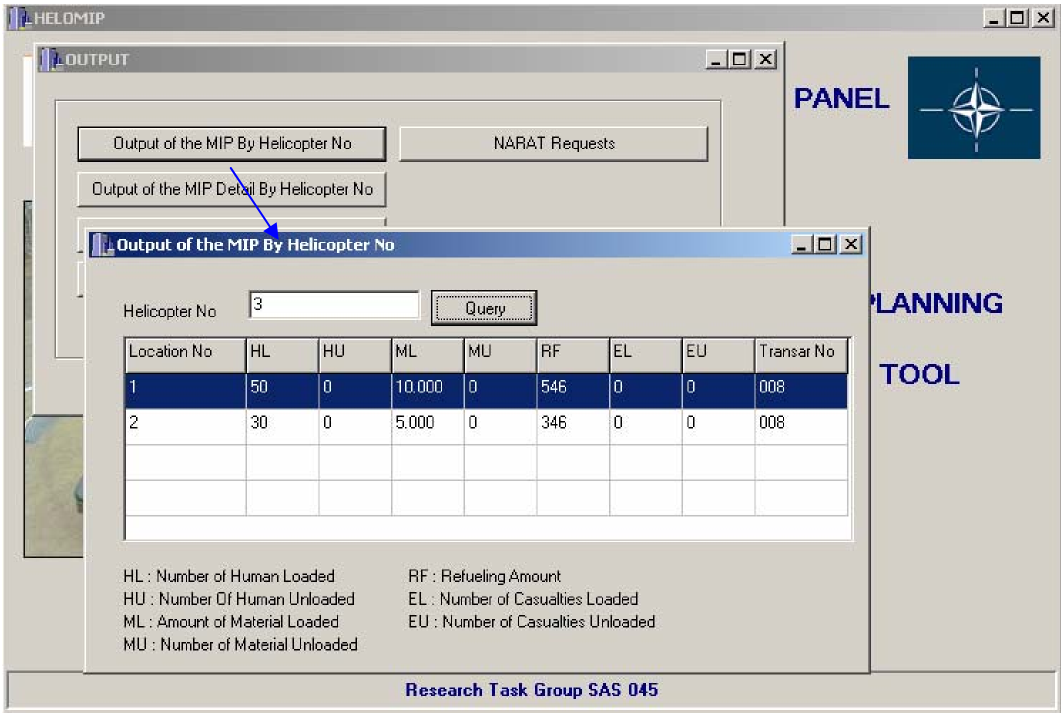


Figure 5.21: HELOMIP Output Display (2).

By clicking on **MIP DETAIL BY HELICOPTER NUMBER**, the user will be able to see the movement of the defined helicopter with the load configuration and refueling amount.

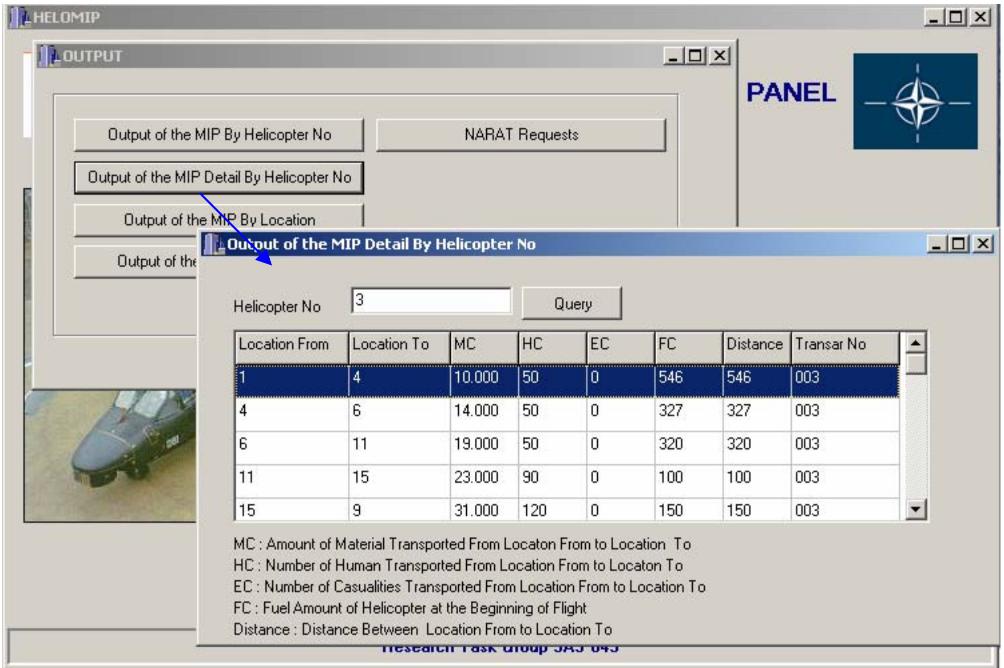


Figure 5.22: HELOMIP Output Display (3).

By clicking on **MIP BY LOCATION**, the user will be able to see the operation that will take place in the defined node. This table will give information regarding the type and cargo of the helicopters destined to arrive at this node.

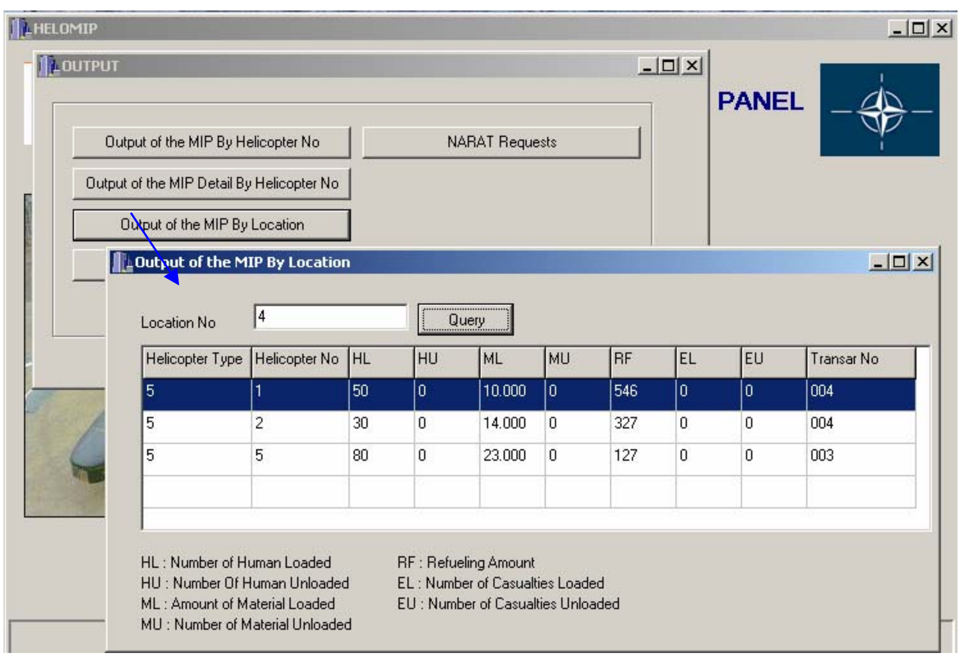


Figure 5.23: HELOMIP Output Display (4).

By clicking on **MIP BY LOAD TYPE**, the user will be able to see the load movement by load type. The table will give the user information about the type of helicopter, the amount of material that will be carried in this load type.

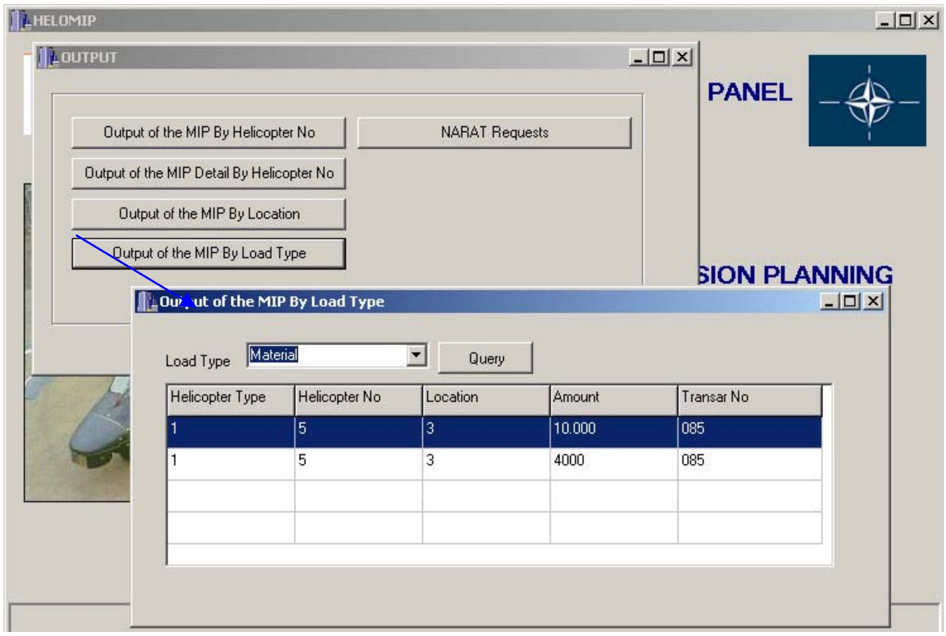


Figure 5.24: HELOMIP Output Display (5).

The user will be able to see which of the NARAT requests will be completed in this mission planning process.

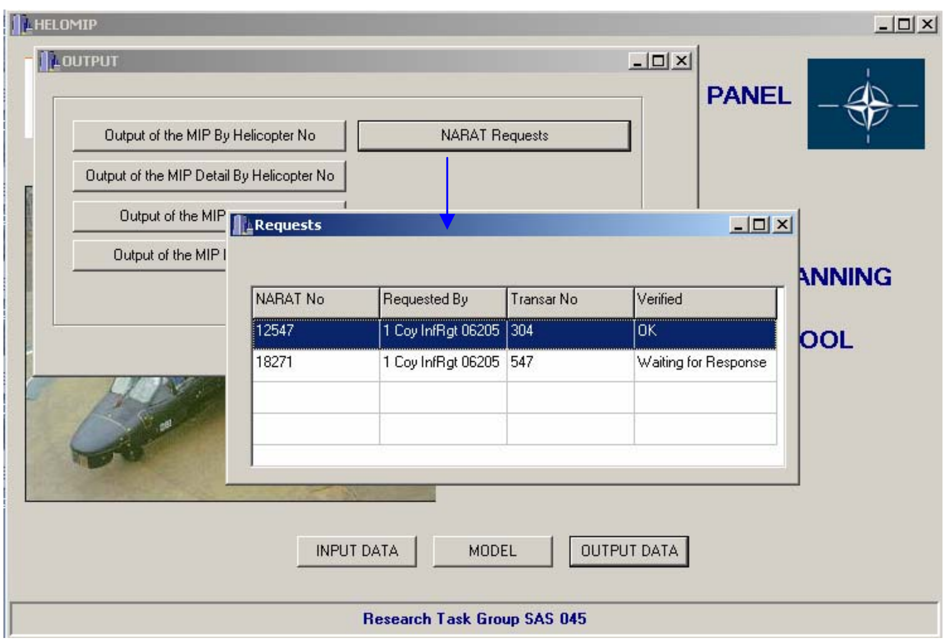


Figure 5.25: HELOMIP Output Display (6).

Chapter 6 – SCENARIO ANALYSIS

The decision support tool is desired to be used in military environments as well as in disaster relief and humanitarian aid environments. Therefore, two completely different scenarios are necessary to describe and to test the functions and the goals of the decision support tool (DST). As it is stated in Chapter 4, the intended DST will be used dynamically over a rolling horizon basis where the decision model equipped with all needed simulation infrastructure will be re-run as new data becomes available and existing databases are updated accordingly. At each iteration, the model will determine helicopter tasking based on the pending requests, the newly arriving requests and using the updated information about helicopters, crew, all other relevant resources and the already accomplished and non-accomplished tasks which have been previously planned. On the other hand, since the development of the prototype DST is not included within the scope of RTG SAS-045 and indeed it needs the establishment and the effort of another NATO RTG to be assigned by the SAS Panel, hopefully in near future, an operational DST is not available at the time of solving sample scenarios. Thus, one-time static solutions are obtained for both scenarios by considering the set of requests given in scenario descriptions as initial conditions.

6.1 DISASTER RELIEF AND HUMANITARIAN AID ENVIRONMENT

In disaster relief scenarios, different transportation tasks are to be executed, and the number of different helicopter types may be very high. The transportation tasks mostly required in such a scenario are:

- Air transport of medical and technical teams;
- Air transport of stocks;
- Air reconnaissance; and
- Casualty and medical evacuation.

The main problem in a disaster relief scenario is that the Air transport requests are always urgent. As new air transport requests arrive at the crisis management centers, re-planning of the tasking of the transport helicopters should be naturally done.

The following scenario is based on an earthquake disaster in TURKEY. The magnitude of the earthquake in the eastern part of TURKEY is recorded to be 7.2 on the Richter scale. Among the places most affected by the earthquake are ERZINCAN and SUSEHRI. It is reported that about 2000 buildings in ERZINCAN and about 500 buildings in SUSEHRI have collapsed and Erzincan Airport has become unusable. The casualties are estimated to be 3200 in ERZINCAN and 800 in SUSEHRI. A fleet of AS-532, UH-60, CH-47 and CH-53, in total 68 transport helicopters are available to execute multiple transportation tasks.

The government in TURKEY has formed a crisis center to coordinate disaster relief operations. The crisis center has decided to use helicopters as the main transportation asset to deploy personnel and material to and from ERZINCAN and SUSEHRI. In order to come up with a deployment plan, the crisis center has demanded information regarding casualties, rescue teams, medical people, military troops, food, medicine, blankets, tents, etc., from several governmental and non-governmental institutions and requested the OR team in the TGS HQ to determine a quick transportation plan. The OR team has gathered the information and summarized it to be used in deployment planning.

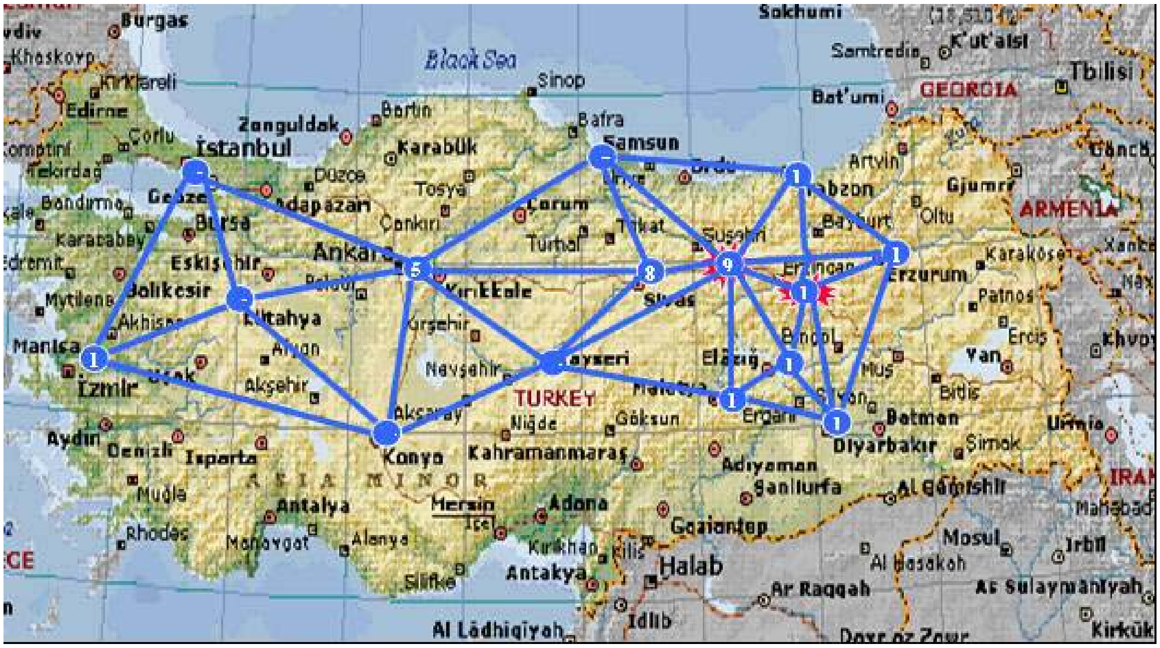


Figure 6.1: Operational Network on the Map.

6.1.1 Inputs of the Model

The availability of different types of helicopters at each node is given in Table 6.1. Each type of helicopter has known transportation capacity shown in Table 6.2.

Table 6.1: Number of Available Helicopters of Each Type at the Nodes

Number of Available Helicopters	NODE														
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
AS-532	5	5			5	2				3	2			4	
UH-60	3	2			5					2			2		1
CH-47	2	2			5					2				2	
CH-53	2	3			5					4					

Table 6.2: Personnel and Material Carrying Capacities of Each Type of Helicopter

Helicopter Type	Personnel and material to carry			Total Weight Capacity (kg.)
	Personnel		Material (Medicine, Food, blankets, tents, etc.) (in kgs)	
	Passenger (troop, rescue team member, etc.) (in numbers)	Casualty (in numbers)		
AS-532	17	6	2000	2000
UH-60	14	4	1800	1800
CH-47	40	24	7200	7200
CH-53	35	24	5500	5500

Table 6.3: Demand and Supply Quantities of Personnel and Material at the Nodes (A – sign shows the demand value at the node.)

Node	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
Passenger (in numbers)	50	70	0	0	100	50	20	0	– 120	– 400	100	20	20	60	30
Casualty (in numbers)									480	60					
Material (tons)	10	27	0	14	30	5	13	0	– 65	– 28	10	6	10	12	20

Table 6.4: Locations and Capacities of Hospitals at the Nodes

Node	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
Hospital Capacity					350	150		150		200	150		100	420	200

6.1.2 Scenario Solution for MODEL – C1

The solution is given in terms of the routes of the helicopters, where 24 helicopters are needed to complete this task. Helicopters depart from their initial locations and visit operation bases and demand nodes according to the assigned routes. Fuel consumption rates are taken as one per distance. The routes and transportation amounts are given in Figure 6.3 as it is explained in legend in Figure 6.2.

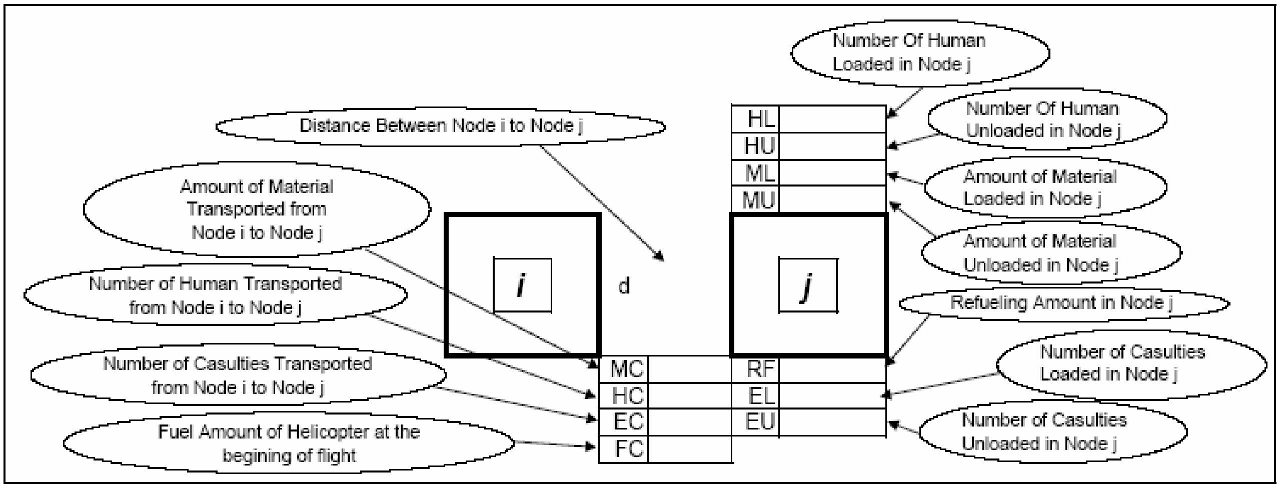


Figure 6.2: Legend for MODEL C1 Solutions.

TRANSAR NO:
NARAT NO:

Helicopter Type : AS-532 Number of Helicopters : 5
Initial Location : 5

	5	2	1	4	6	8	7	11	9	12	15	14	13
HL	4	2	2	0	2	0	1	0	4	2	0	0	0
HU	0	0	0	0	0	0	0	0	0	0	14	3	0
ML	500	500	250	100	125	50	100	125	125	125	0	0	0
MU	0	0	0	0	0	0	0	0	0	0	625	1,375	0
RF	453	565	546	327	194	1615	0	0	0	0	0	0	0
EL	0	0	0	0	0	0	0	0	0	0	0	0	0
EU	0	0	0	0		0	0	0	0	0	0	0	0

	5–2	2–1	1–4	4–6	6–8	8–7	7–11	11–9	9–12	12–15	15–14	14–13
d	453	565	546	327	194	340	232	230	150	188	324	151
MC	500	1,000	1,250	1,350	1,475	1,525	1,625	1,750	1,875	2,000	1,375	0
HC	4	6	8	8	10	10	11	11	15	17	3	0
EC	0	0	0	0	0	0	0	0	0	0	0	0
FC	453	565	546	327	194	340	232	230	150	188	324	151

TRANSAR NO:
NARAT NO:

Helicopter Type : UH-60 Number of Helicopters : 4
Initial Location : 5

	5	2	1	4	6	8	7	11	9	12	15	14	13
HL	3	2	2	0	1	1	1	0	3	2	0	0	0
HU	0	0	0	0	0	0	0	0	0	0	12	2	0
ML	450	450	225	100	113	45	90	112	113	112	0	0	0
MU	0	0	0	0	0	0	0	0	0	0	562	1,238	0
RF	453	565	546	327	194	1615	0	0	0	0	0	0	0
EL	0	0	0	0	0	0	0	0	0	0	0	0	0
EU	0	0	0	0	0	0	0	0	0	0	0	0	0

	5–2	2–1	1–4	4–6	6–8	8–7	7–11	11–9	9–12	12–15	15–14	14–13
d	453	565	546	327	194	340	232	230	150	188	324	151
MC	450	900	1,125	1,215	1,328	1,373	1,463	1,575	1,688	1,800	1,238	0
HC	3	5	7	7	8	8	9	9	12	14	2	0
EC	0	0	0	0	0	0	0	0	0	0	0	0
FC	453	565	546	327	194	340	232	230	150	188	324	151

TRANSAR NO:
NARAT NO:

Helicopter Type : CH-47 Number of Helicopters : 3
Initial Location : 5

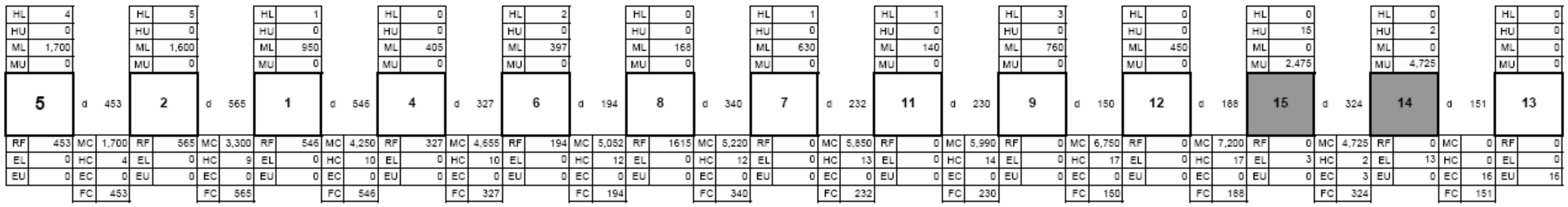
	5	2	1	4	6	8	7	11	9	12	15	14	13
HL	9	5	5	0	4	0	3	1	9	4	0	0	0
HU	0	0	0	0	0	0	0	0	0	0	34	6	0
ML	1,800	1,800	900	360	450	180	360	450	450	450	0	0	0
MU	0	0	0	0	0	0	0	0	0	0	2,250	4,950	0
RF	453	565	546	327	194	1,615	0	0	0	0	0	0	0
EL	0	0	0	0	0	0	0	0	0	0	3	13	0
EU	0	0	0	0	0	0	0	0	0	0	0	0	16

	5–2	2–1	1–4	4–6	6–8	8–7	7–11	11–9	9–12	12–15	15–14	14–13
d	453	565	546	327	194	340	232	230	150	188	324	151
MC	1,800	3,600	4,500	4,860	5,310	5,490	5,850	6,300	6,750	7,200	4,950	0
HC	9	14	19	19	23	23	26	27	36	40	6	0
EC	0	0	0	0	0	0	0	0	0	0	3	16
FC	453	565	546	327	194	340	232	230	150	188	324	151

Figure 6.3: Solution for MODEL – C1.

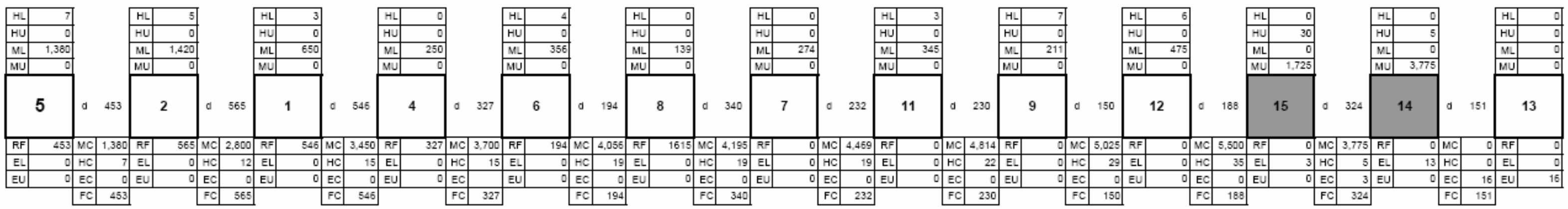
TRANSAR NO:
NARAT NO:

Helicopter Type : CH-47 Number of Helicopters : 2
Initial Location : 5



TRANSAR NO:
NARAT NO:

Helicopter Type : CH-53 Number of Helicopters : 5
Initial Location : 5



TRANSAR NO:
NARAT NO:

Helicopter Type : CH-47 Number of Helicopters : 5
Initial Location : 13

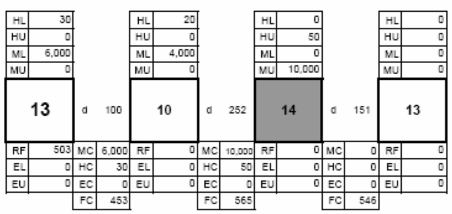


Figure 6.3: Solution for MODEL-C1 (Cont'd).

6.2 MILITARY ENVIRONMENT

In military environments, the tasks to be executed by transport helicopters are:

- Transport of Personnel;
- Transport of Material; and
- Casualty and Medical evacuation (CASEVAC and MEDEVAC).

In Joint and Combined NATO missions, air transport has to be requested by using the NARAT format as described in ATP 53. The reply of the approving authority is submitted in the TRANSAR format.

The following scenario is based on the exercise "EUROPEAN CHALLENGE". In this scenario, the Phase 2c of the military operation of the German Air Mobile Brigade 1 requires the air transport of two reinforced Infantry Coys and one Command Element. The troops are the Quick Reaction Force Tactical Reserve of the Air Mobile Brigade, one of which is equipped with light armoured vehicles. In Phase 2c of this scenario, their task is to gain and to secure 2 ELBE river crossings as precondition for the movement of the Main Forces (AUT and CZE Bde) to their AOR.

The Infantry Forces consist of 555 Soldiers and the following 99 light armored vehicles:

- 60 ESK;
- 13 Wiesel 2;
- 8 Wiesel 2 Mortar;
- 12 Wiesel 1 TOW; and
- 6 Wiesel 1 chain gun.

The troops are located in an Assembly Area 100 km south of BREMEN and need to be transported in one wave to their mission area at the river ELBE south of HAMBURG. The Airmobile Brigade is equipped with 32 NH-90 helicopters and is supported for this mission with one German battalion of 36 CH-53 helicopters, one Dutch battalion of 12 CH-47 helicopters and one battalion of 16 French PUMA helicopters. The decision support tool is expected to optimize the tasking of this multinational mixed helicopter fleet. The complete Air Transport Request is stated in 5 NARATs provided in Appendix 6.1. They are intended as the input for the decision support tool. For the sake of representational simplicity, the coordinates of the locations are labeled as given in Table 6.5.

Table 6.5: Location Legend

Location Label	Coordinates	
1	52 32 45 N	009 17 57 E
2	53 18 41 N	010 31 25 E
3	52 32 04 N	009 20 53 E
4	53 19 25 N	010 38 36 E
5	52 34 08 N	009 12 24 E
6	53 27 19 N	010 28 41 E
7	52 35 07 N	009 05 59 E
8	53 23 01 N	010 17 39 E
9	52 33 51 N	009 03 26 E
10	53 27 19 N	010 22 03 E
11	Assembly Area BREMEN	

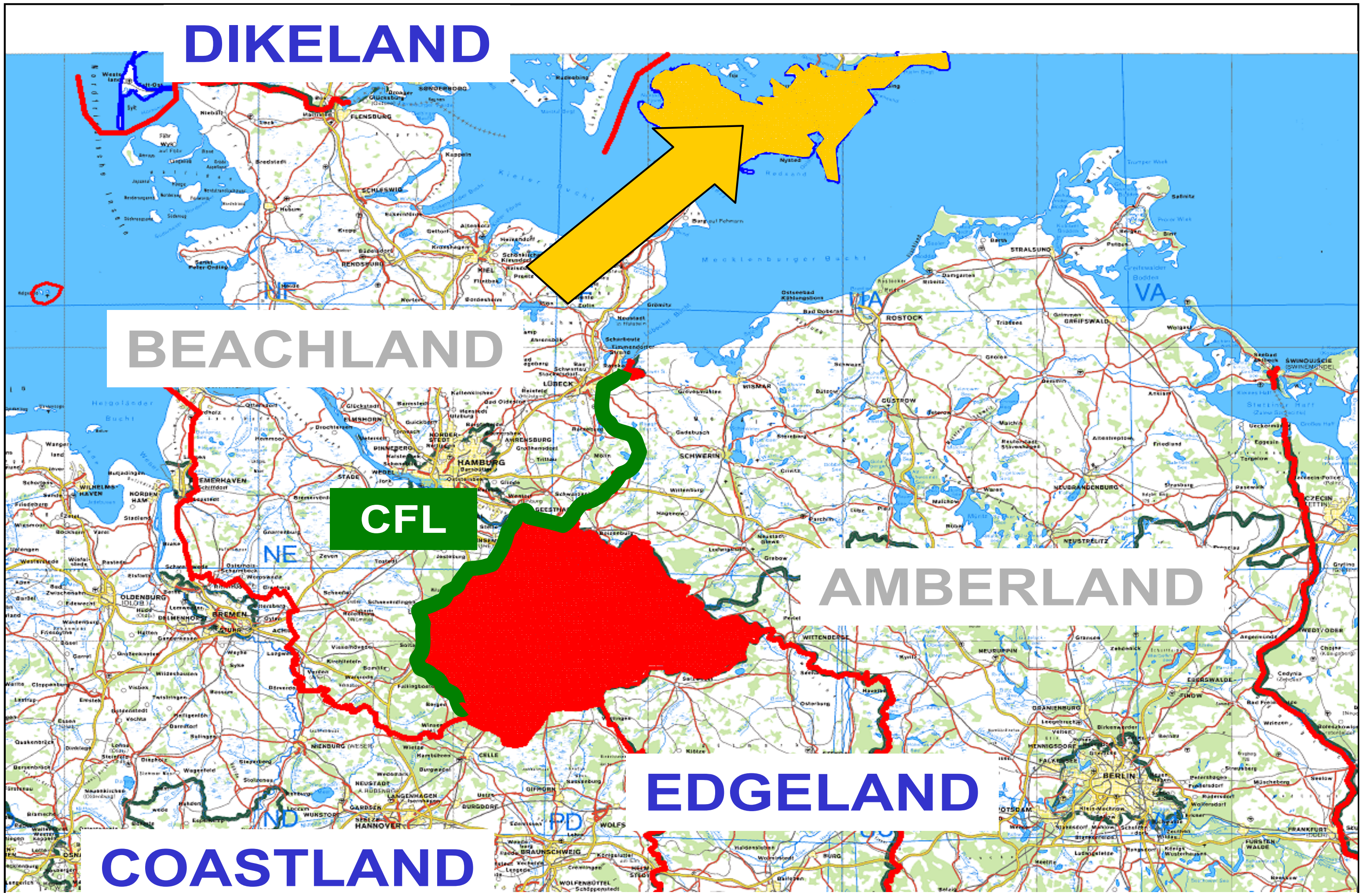


Figure 6.4: The Separation Map (The conflict parties Beachland and Amberland have to be separated by force. In the red area, a Demilitarized Zone has to be established by the forces. Coastland has offered Host Nation Support. Edgeland and Dikeland are neutral.)

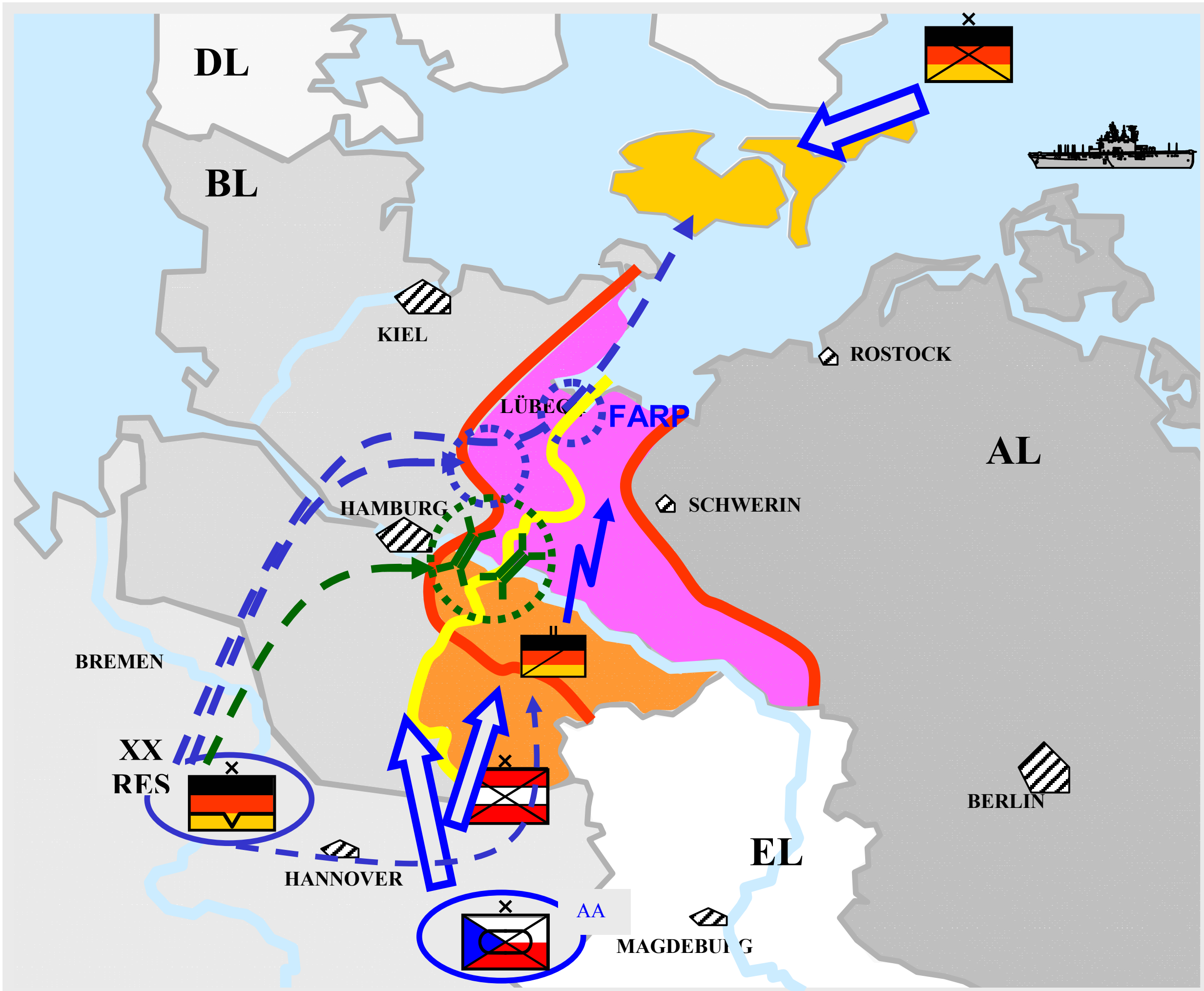


Figure 6.5: Phase 2c of the Operation (In Phase 2c of the whole Operation, the Forces of the German Airmobile Brigade 1 have to gain and to secure 2 ELBE river crossings as precondition for the AUT and the CZE BDE to reach their AOR.)

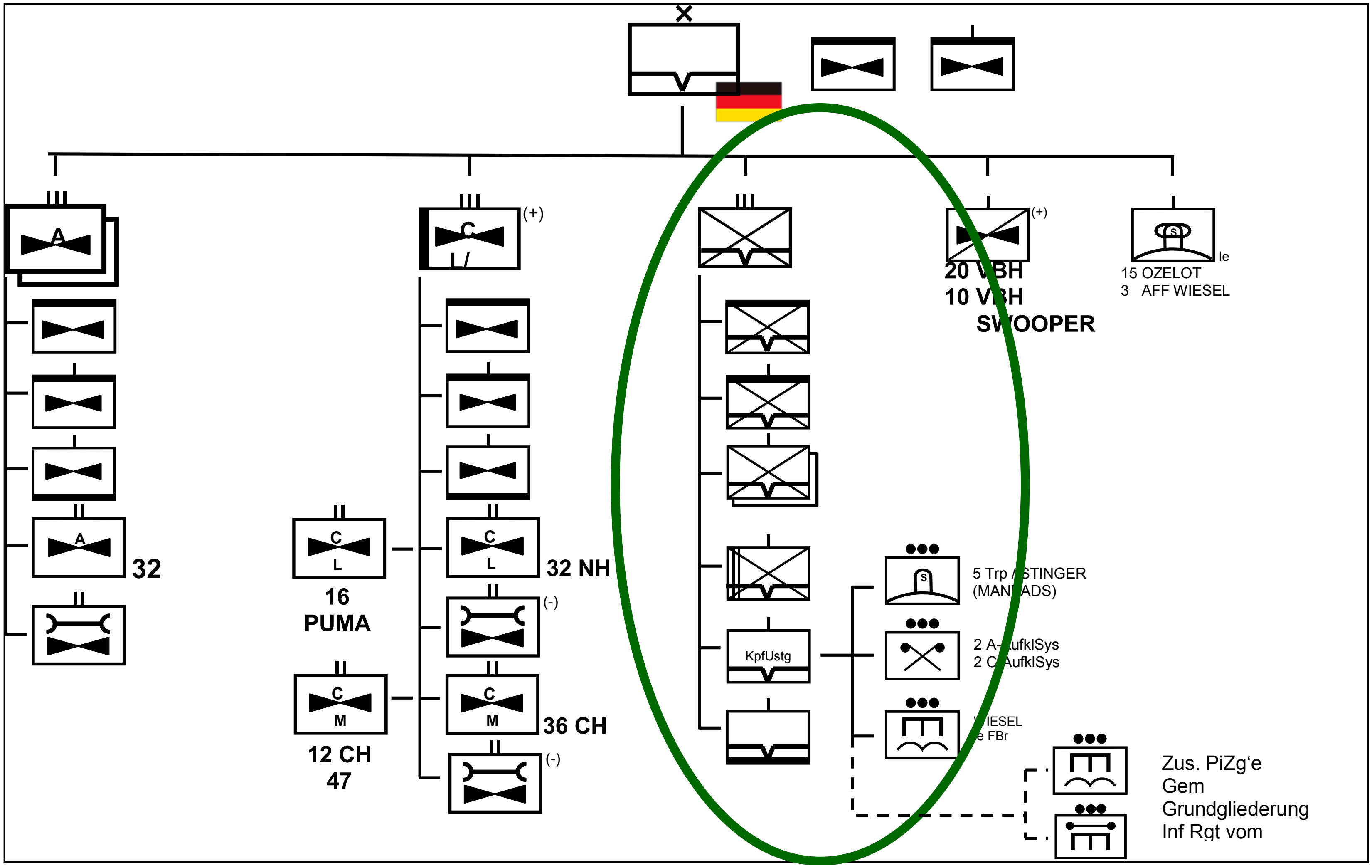


Figure 6.6: Elements of the InfRgt (Elements of the InfRgt are to be transported by helicopter to their mission south of Hamburg.)

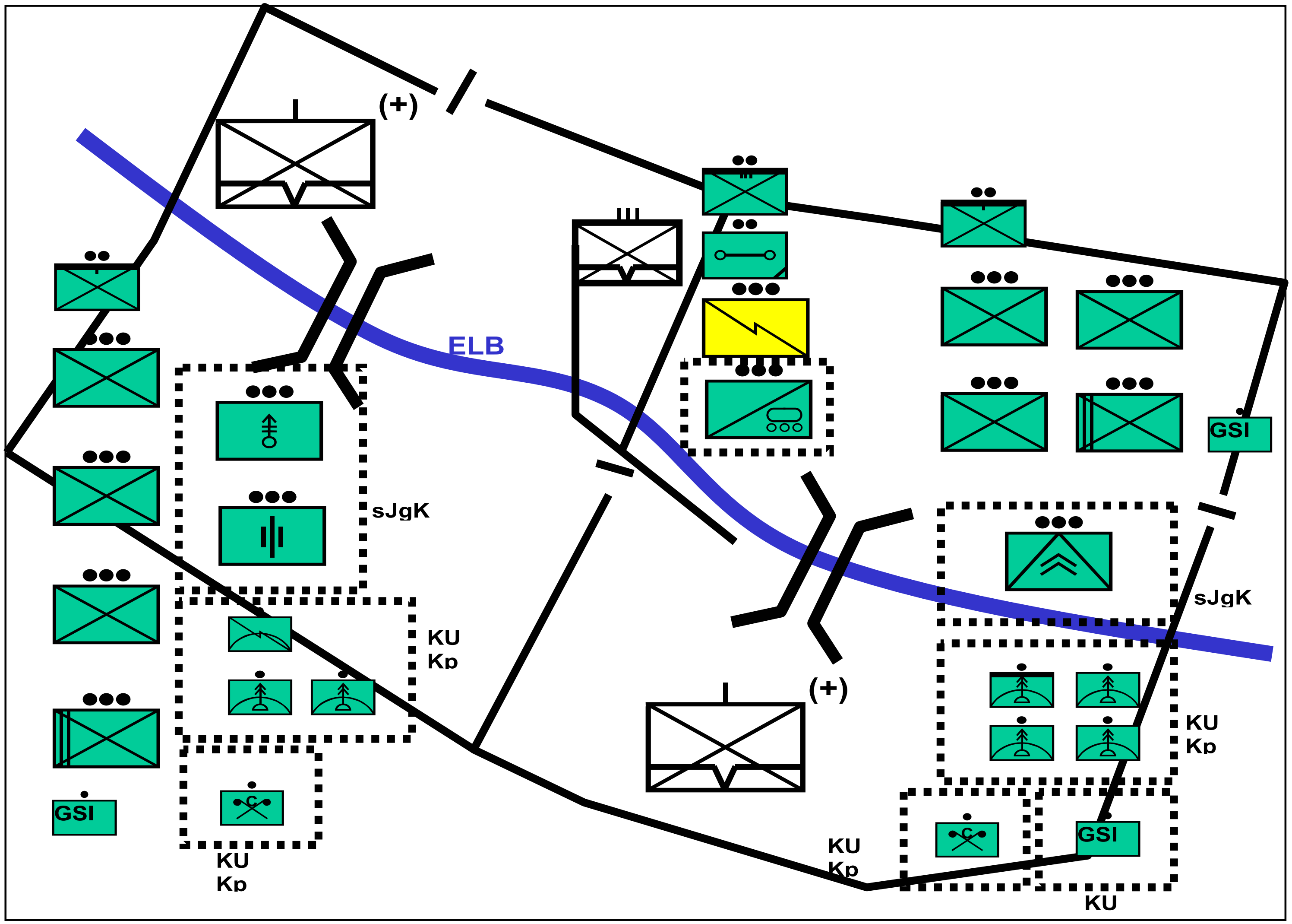


Figure 6.7: The Command Elements and Reinforced Coys (One Command Element and 2 reinforced Coys of the Infantry Regiment have to gain and to secure the ELBE river crossings.)

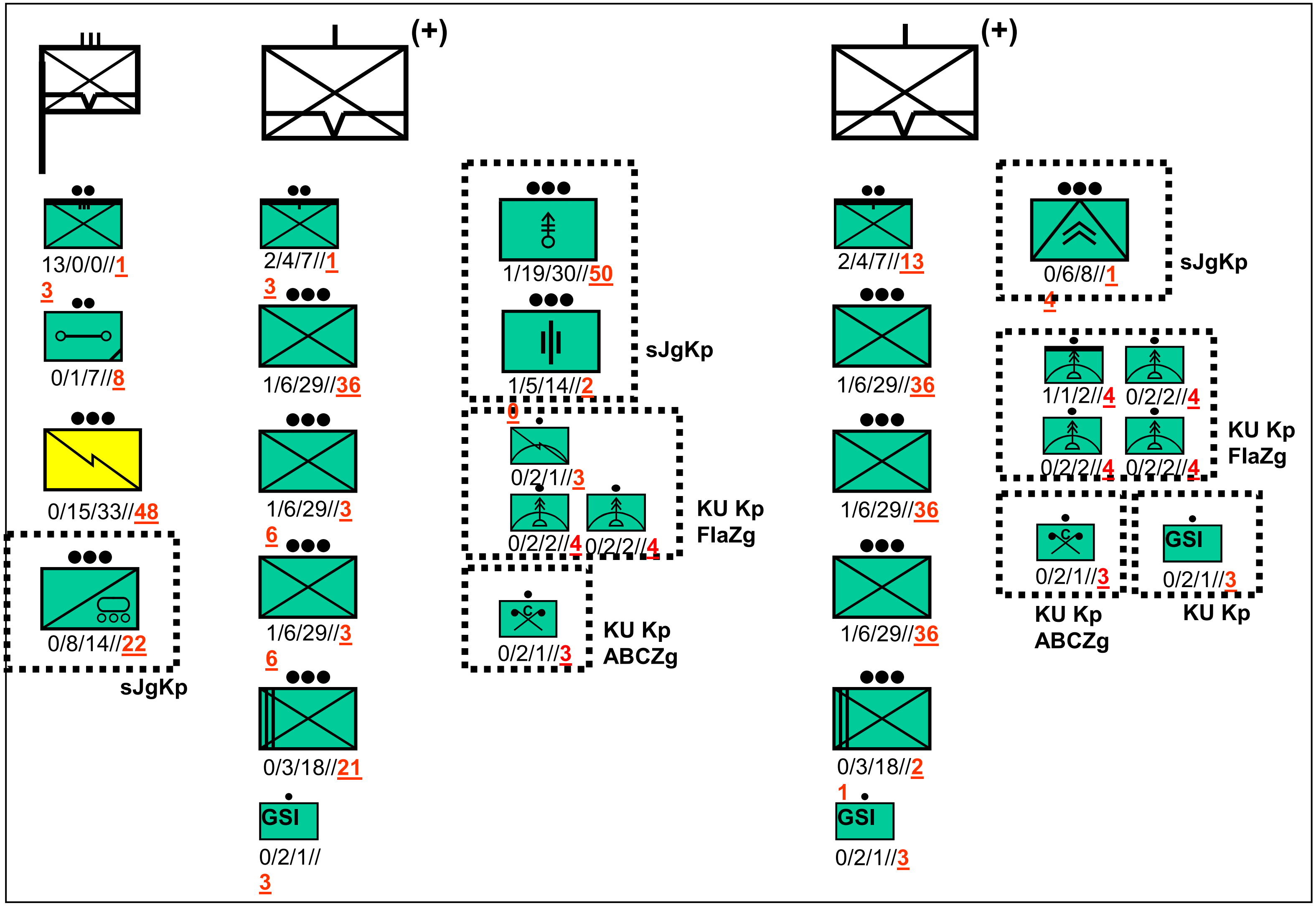


Figure 6.8: Soldiers to be Transported (Totally 555 Soldiers are to be transported.)

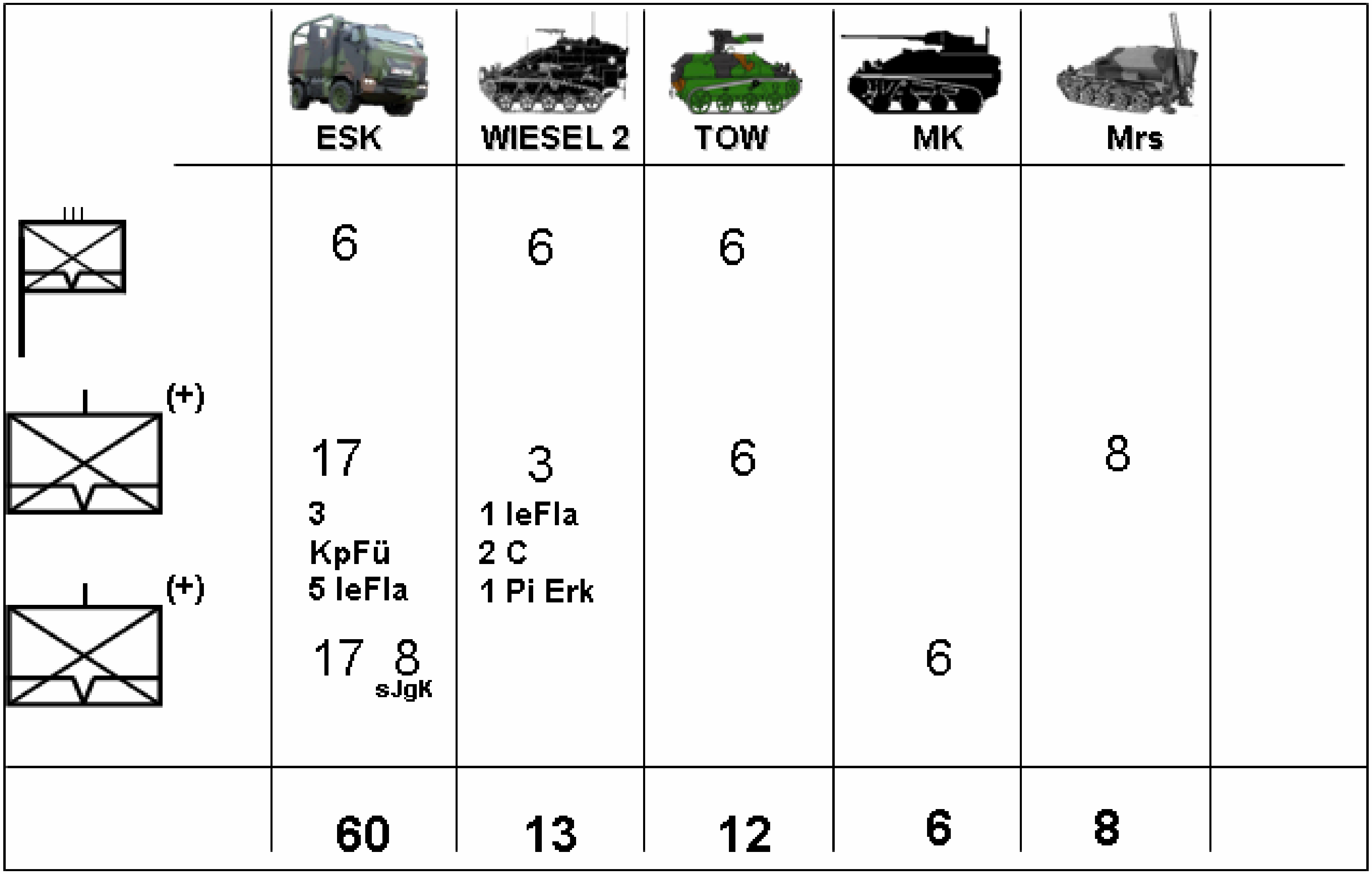


Figure 6.9: Light Armoured Vehicles (99 light armoured vehicles are to be transported in the first wave.)

The solution of the model provides the number of helicopters, their respective cargo quantities and the number sorties to execute the task orders given in NARATs where the objective function is selected to minimize the total number of sorties. The solution is given in Table 6.6, Figure 6.10 and Figure 6.11.

Table 6.6: Assignments to Helicopters

NARAT	Initial Node of N.	Final Node of N.	H. Type	# of H's	# of Sorties	Material Trans.	Human Trans.
1	1	2	NH-90	1	1	2,000	4
			CH-53	16	16	86,300	127
			PUMA	1	1	1,800	14
NARAT 1 – TOTAL				18	18	90,100	145
2	3	4	CH-53	5	5	27,500	36
			CH-47	9	12	86,300	-
NARAT 2 – TOTAL				14	17	113,800	36
3	5	6	NH-90	1	1	2,000	-
			CH-47	1	13	92,000	91
NARAT 3 – TOTAL				2	14	94,000	91
4	7	8	NH-90	28	28	56,000	-
			CH-53	1	1	5,500	-
			CH-47	1	4	28,600	145
NARAT 4 – TOTAL				30	33	90,100	145
5	9	10	NH-90	2	2	4,000	34
			CH-53	14	14	77,000	77
			CH-47	1	1	5,800	-
			PUMA	15	15	27,000	-
NARAT 5 – TOTAL				32	32	113,800	111

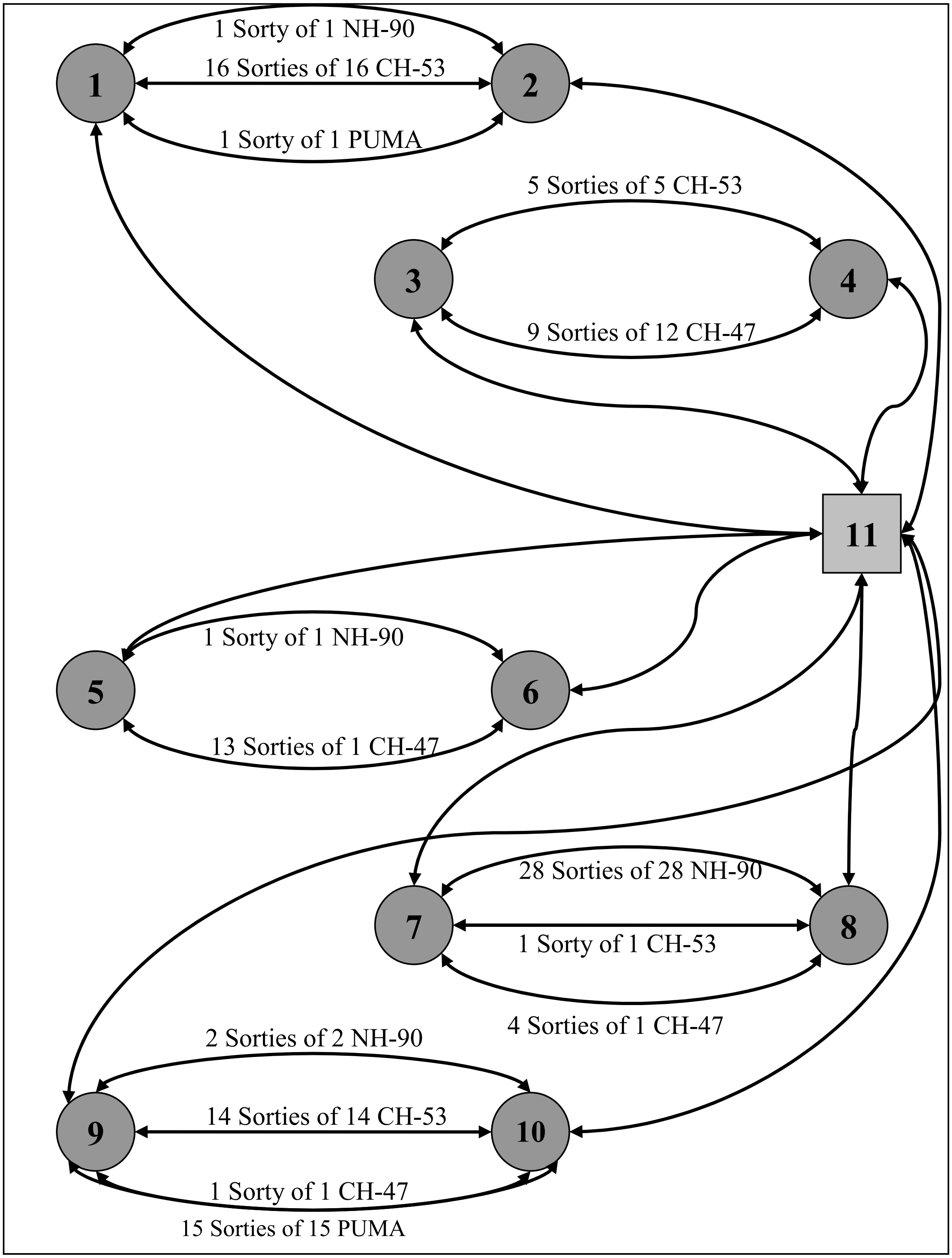


Figure 6.10: A General Overview of Helicopter Routes.

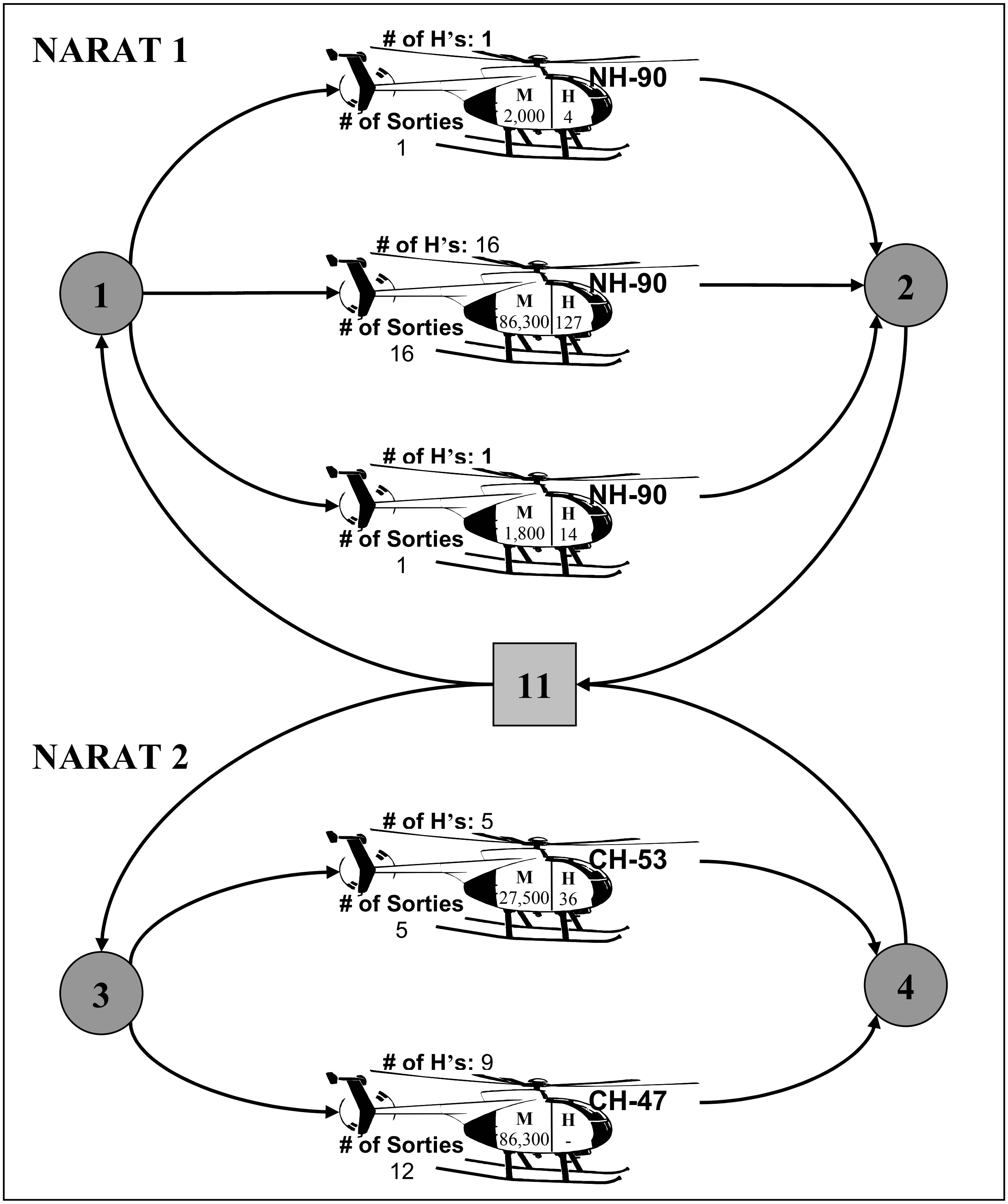


Figure 6.11: Routes and Transportation Details of Helicopters.

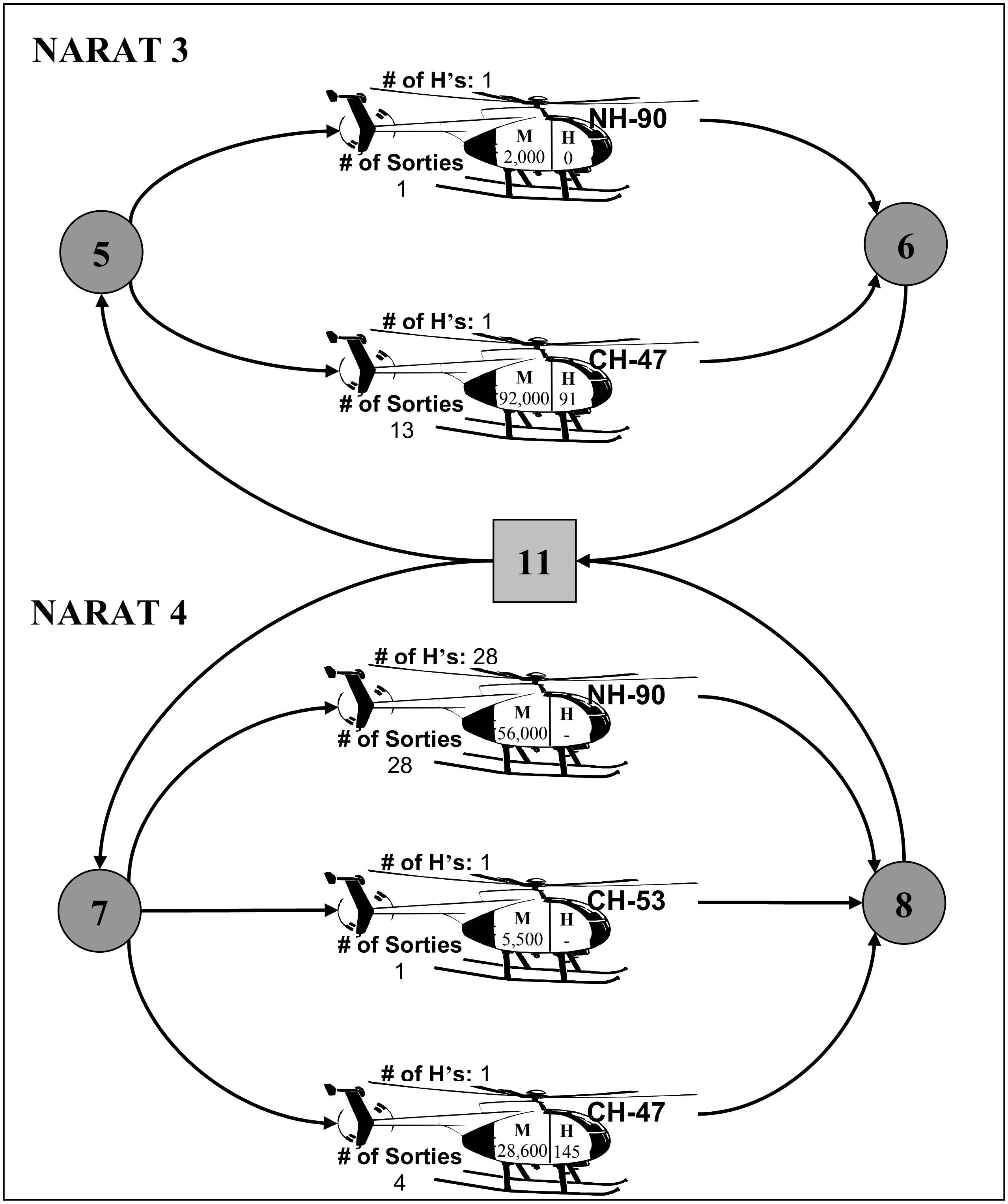


Figure 6.11: Routes and Transportation Details of Helicopters (cont'd).

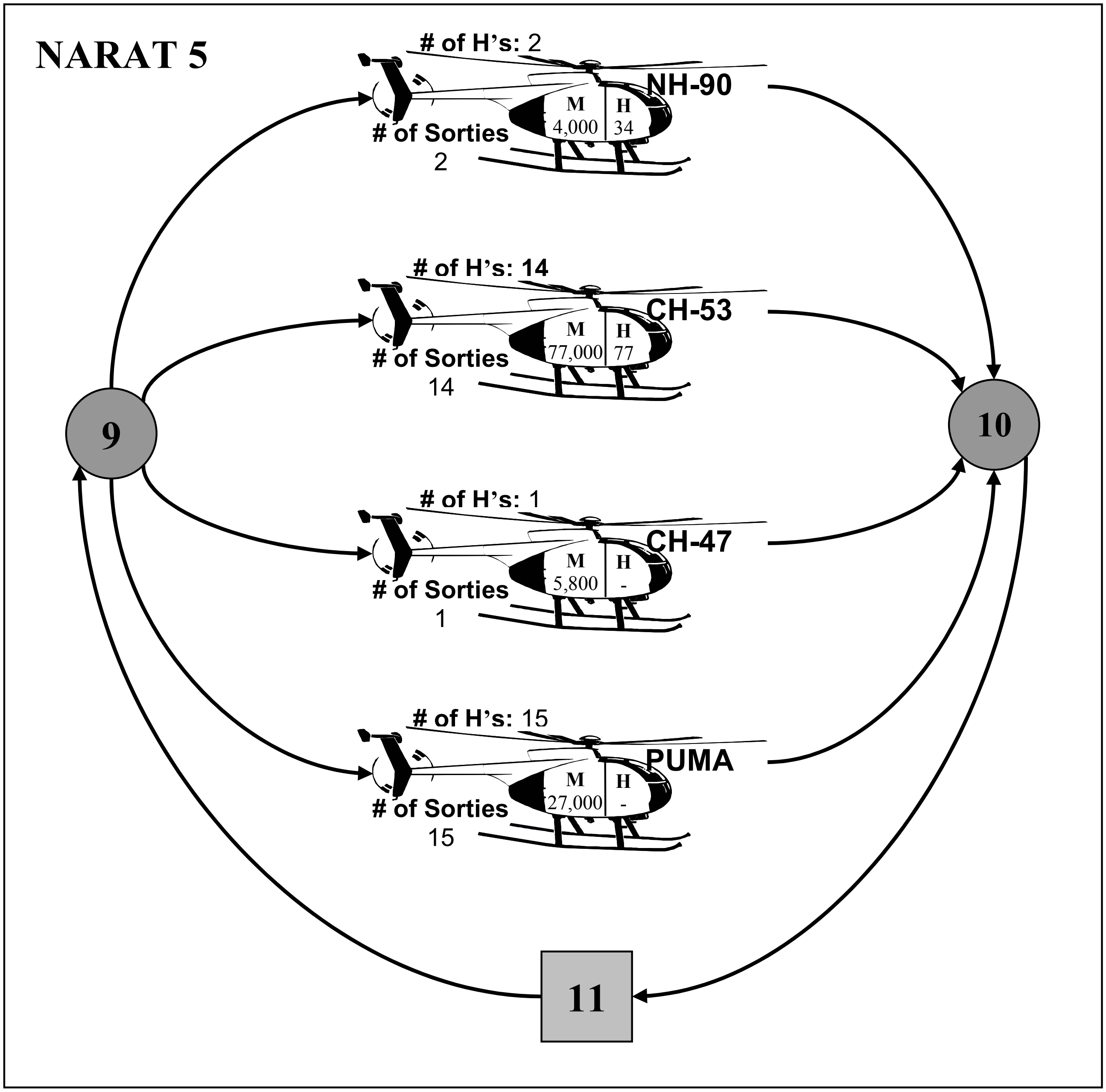


Figure 6.11: Routes and Transportation Details of Helicopters (cont'd).

Appendix 6.1 – NARATs

NARAT 1: 1 Coy InfRgt 062005

ONE		**Transport of Personal**
A.	145	Number of Persons
B.	21750 KGS	Total weight
C.	52° 35 07 N 009° 05 59 E	Coordinates of loading point
D.	53° 23 01 N 010° 17 39 E	Coordinates of de-loading point
E.	Land	Method
G.	10	Number of landing spots at de-loading point
H.	PAX are on the vehicles below	Remarks
TWO		**Transport of Material**
A.	90100 KGS	Total weight of Material
B.	NOST	Description of cargo
C.	1	Priority
F.	ESK: 5300 KGS	Weight
G.	ESK: 4,24 m	Length
H.	ESK: 1,85 m	Width
I.	ESK: 1,91 m	Height
J.	17 ESK	Number
	52° 35 07 N 009° 05 59 E	Coordinates of loading point
L.	53° 23 01 N 010° 17 39 E	Coordinates of de-loading point
O.	Land	Preferred Delivery Method
P.	TAC	Reason For Delivery Method
R.	10	Number of landing spots at de-loading point
THREE		**Transport of Casualties**
NIL		
FOUR		**Schedule**
A.	231945Z	No move before given time
B.	232030Z	Earliest delivery time at the landing site
C.	232115Z	Latest delivery time at the landing site
FIVE		**Communication**
A.	2 Coy InfRgt	Unit
B.	Tiger 2	Call signs
C.	121.375 P 221.00 S	Frequencies
D.	Voice	Mode

SIX		**Contacts**
	A. Onload	Type of Contact (Onload/Offload)
	B. HQ AMB	Location of Contact
	C. Cpt Meyer	Name and Rank
	D. 0221 9371 2266	Phone

NARAT 2:1 Coy InfRgt 052005

ONE		**Transport of Personal**
	F. 111	Number of Persons
	G. 16650 KGS	Total weight
	H. 52° 33 51 N 009° 03 26 E	Coordinates of loading point
	I. 53° 27 19 N 010° 22 03 E	Coordinates of de-loading point
	J. Land	Method
	I. 12	Number of landing spots at de-loading point
	J. PAX are on the vehicles below	Remarks

TWO		**Transport of Material**
	D. 113800 KGS	Total weight of Material
	E. NOST	Description of cargo
	F. 1	Priority
	K. ESK: 5300 KGS Wiesel 2: 4300 KGS Wiesel 2 MRS: 4300 KGS Wiesel 1 MK: 2800 KGS	Weight
	L. ESK: 4,24 m Wiesel 2: 4,153 m Wiesel 2 MRS: 4,153 m Wiesel 1 MK: 3,545 m	Length
	M. ESK: 1,85 m Wiesel 2: 1,852 m Wiesel 2 MRS: 1,852 m Wiesel 1 MK: 1,820 m	Width
	N. ESK: 1,91 m Wiesel 2: 1,752 m Wiesel 2 MRS: 1,752 m Wiesel 1 MK: 1,825 m	Height
	O. 11 ESK 1 Wiesel 2 8 Wiesel 2 MRS 6 Wiesel 1 MK	Number
	M. 52° 33 51 N 009° 03 26 E	Coordinates of loading point
	N. 53° 27 19 N 010° 22 03 E	Coordinates of de-loading point

Q.	Land	Preferred Delivery Method
R.	TAC	Reason For Delivery Method
S.	12	Number of landing spots at de-loading point

THREE		**Transport of Casualties**
NIL		

FOUR		**Schedule**
D.	231945Z	No move before given time
E.	232030Z	Earliest delivery time at the landing site
F.	232115Z	Latest delivery time at the landing site

FIVE		**Communication**
E.	1 Coy Support InfRgt	Unit
F.	Tiger 3	Call signs
G.	122.175 P 223.100 S	Frequencies
H.	Voice	Mode

SIX		**Contacts**
E.	Onload	Type of Contact (Onload/Offload)
F.	HQ AMB	Location of Contact
G.	Cpt Müller	Name and Rank
H.	0221 9371 2233	Phone

NARAT 3: 2 Coy InfRgt 062005

ONE		**Transport of Personal**
K.	145	Number of Persons
L.	21750 KGS	Total weight
M.	52° 32 45 N 009° 17 57 E	Coordinates of loading point
N.	53° 18 41 N 010° 31 25 E	Coordinates of de-loading point
O.	Land	Method
K.	9	Number of landing spots at de-loading point
L.	PAX are on the vehicles below	Remarks

TWO		**Transport of Material**
G.	90100 KGS	Total weight of Material
H.	NOST	Description of cargo
I.	1	Priority
P.	ESK: 5300 KGS	Weight
Q.	ESK: 4,24 m	Length
R.	ESK: 1,85 m	Width
S.	ESK: 1,91 m	Height
T.	17 ESK	Number
	52° 32 45 N 009° 17 57 E	Coordinates of loading point

O.	53° 18 41 N 010° 31 25 E	Coordinates of de-loading point
S.	Land	Preferred Delivery Method
T.	TAC	Reason For Delivery Method
T.	9	Number of landing spots at de-loading point

THREE		**Transport of Casualties**
NIL		

FOUR		**Schedule**
G.	231945Z	No move before given time
H.	232030Z	Earliest delivery time at the landing site
I.	232115Z	Latest delivery time at the landing site

FIVE		**Communication**
I.	2 Coy InfRgt	Unit
J.	Tiger 2	Call signs
K.	121.375 P 221.00 S	Frequencies
L.	Voice	Mode

SIX		**Contacts**
I.	Onload	Type of Contact (Onload/Offload)
J.	HQ AMB	Location of Contact
K.	Cpt Berger	Name and Rank
L.	0221 9371 2254	Phone

NARAT 4: 2 Coy Support InfRgt 072005

ONE		**Transport of Personal**
P.	36	Number of Persons
Q.	5400 KGS	Total weight
R.	52° 32 04 N 009° 20 53 E	Coordinates of loading point
S.	53° 19 25 N 010° 38 36 E	Coordinates of de-loading point
T.	Land	Method
M.	6	Number of landing spots at de-loading point
N.	PAX are on the vehicles below	Remarks

TWO		**Transport of Material**
J.	113800 KGS	Total weight of Material
K.	NOST	Description of cargo
L.	1	Priority
U.	ESK: 5300 KGS Wiesel 1 TOW: 2800 KGS	Weight
V.	ESK: 4,24 m Wiesel 1 TOW: 3,545 m	Length

W.	ESK: 1,85 m Wiesel 1 TOW: 1,820 m	Width
X.	ESK: 1,91 m Wiesel 1 TOW: 1,825 m	Height
Y.	7 ESK 12 Wiesel 1 TOW	Number
P.	52° 32 04 N 009° 20 53 E	Coordinates of loading point
Q.	53° 19 25 N 010° 38 36 E	Coordinates of de-loading point
U.	Land	Preferred Delivery Method
V.	TAC	Reason For Delivery Method
U.	6	Number of landing spots at de-loading point

THREE — **Transport of Casualties**

NIL

FOUR — **Schedule**

J.	231945Z	No move before given time
K.	232030Z	Earliest delivery time at the landing site
L.	232115Z	Latest delivery time at the landing site

FIVE — **Communication**

M.	2 Coy Support InfRgt	Unit
N.	Tiger 3	Call signs
O.	120.575 P 220.150 S	Frequencies
P.	Voice	Mode

SIX — **Contacts**

M.	Onload	Type of Contact (Onload/Offload)
N.	HQ AMB	Location of Contact
O.	Cpt Meier	Name and Rank
P.	0221 9371 2249	Phone

NARAT 5: Cmd Group InfRgt 032005

ONE — **Transport of Personal**

U.	91	Number of Persons
V.	13650 KGS	Total weight
W.	52° 34 08 N 009° 12 24 E	Coordinates of loading point
X.	53° 27 19 N 010° 28 41 E	Coordinates of de-loading point
Y.	Land	Method
O.	8	Number of landing spots at de-loading point
P.	PAX are on the vehicles below	Remarks

TWO			**Transport of Material**
	M.	94000 KGS	Total weight of Material
	N.	NOST	Description of cargo
	O.	1	Priority
	Z.	ESK: 5300 KGS Wiesel 2: 4300 KGS	Weight
	AA.	ESK: 4,24 m Wiesel 2: 4,153 m	Length
	BB.	ESK: 1,85 m Wiesel 2: 1,852 m	Width
	CC.	ESK: 1,91 m Wiesel 2: 1,752 m	Height
	DD.	8 ESK 12 Wiesel 2	Number
	R.	52° 34 08 N 009° 12 24 E	Coordinates of loading point
	S.	53° 27 19 N 010° 28 41 E	Coordinates of de-loading point
	W.	Land	Preferred Delivery Method
	X.	TAC	Reason For Delivery Method
	V.	8	Number of landing spots at de-loading point
THREE			**Transport of Casualties**
	NIL		
FOUR			**Schedule**
	M.	231930Z	No move before given time
	N.	232015Z	Earliest delivery time at the landing site
	O.	232100Z	Latest delivery time at the landing site
FIVE			**Communication**
	Q.	Cmd Group InfRgt	Unit
	R.	Tiger 1	Call signs
	S.	123.375 P 223.00 S	Frequencies
	T.	Voice	Mode
SIX			**Contacts**
	Q.	Onload	Type of Contact (Onload/Offload)
	R.	HQ AMB	Location of Contact
	S.	LTC Möllmann	Name and Rank
	T.	0221 9371 2233	Phone

Chapter 7 – CONCLUSION

As it is stated in the Terms of Reference and Programme of Work Research of NATO RTG SAS-045 "Computer-Aided Decision Support Tool for Mission Planning in Disaster Relief and Military Operation", the main goal of this research is to propose a framework for a generic and flexible decision support tool that can be used in effective management of helicopter missions both during humanitarian and military operations. The research procedure has consisted of conducting the problem analysis, investigating the concept of solutions and determining relevant technical requirements.

Since NATO's involvement in international disaster assistance has a long history, and the Euro-Atlantic Disaster Response Coordination Centre (EADRCC) and Euro-Atlantic Disaster Response Unit (EADRU) were established in 1998 to formalize NATO's role and responsibility in disaster assistance activities, it is well-known that the decision makers in command of controlling or managing a disaster relief mission need standard and interoperable procedures, guidelines, and regulations to respond quickly and effectively to an emergency situation. Thus, this study has aimed to design computer-based decision support tools (DSTs) which enhance the rapid and effective response capability of NATO commanders at operational and tactical levels during both Article V and Non-Article V Operations.

To achieve the objectives tasked, the work has been carried out in three phases which constitute the modules of this document.

First, the problem areas, processes and functions have been analyzed and defined, and operational description of the problem is presented mainly in three aspects: operational context (environment, desired capability and scope), mission types (key mission tasks and functions) and decision-making framework. A generic scenario is defined in the context of an emergency event in a known area which has been realized by NATO, and NATO has decided to respond to the mission, given the basic information about the number of operational bases, helicopters, crew, logistics (facilities, fuel, etc.), embarkation and debarkation nodes for traffic control, maintenance, refuelling, etc. NATO planners estimate the real time requirements based upon this data, and the requesting body within NATO generates the needed NATO Request for Air Transport Format (NARAT). Then, the Air Transport approving authority as the decision maker will receive transport requests in the form of a NARATs and is expected to generate a practical solution. Since both disaster relief and military operations pose quick response situations where time is the key limiting factor and helicopter moves should be planned and conducted very rapidly, the speed in generating good solutions on a quasi real time basis affects the overall performance of helicopter operations. At this point, the intended DST will help the decision maker find an execution plan of tasks with replanning capability.

The decision-making framework has been determined by analyzing the planning process involving both military and civil agents, procedures, and resources at the time of any crisis in the crisis centers both at the local (municipal) and central (governmental) levels.

Since the intended DST should satisfy all current information, communication and technical requirements needed for robustness, speed, open architecture, accessibility to accurate data on a real-time basis, standardization, inter- operability, flexibility, re-planning possibility, capability to deal with randomness, fuzziness and incompleteness, what-if analysis capability, technology surveys were conducted on decision support tool technologies, information technologies, and geographical information systems, digital maps and mission data compilation technologies. This investigation was completed by preparing a fourth report on existing models, data and knowledge management repositories and planning process in NATO organizations.

In the concept of solution phase, the required models have been developed and solution procedures and algorithms have been proposed to generate efficient and realistic plans. This module presents the mathematical modeling description (i.e. the inputs, constraints, objective functions and outputs), the resolution method (mixed integer programming, heuristics), and computational results on testing scenarios.

The technical requirement phase of this study contains in detail all relevant technical requirements that may directly lead to the development of such a system. In the technical requirement module, information management system, database interfacing module, and the protocols are described and the information support tool dependencies are defined.

The experience and the output of RTG SAS-045 clearly show that valuable expertise and know-how have been accumulated and necessary infrastructure has been planned so as to develop the intended prototype DST. Thus, it is strongly recommended that this work be extended to build a prototype and implementable system to be used during helicopter operations among NATO nations.

Chapter 8 – REFERENCES

Al-Mahmeed, A.S., 1996, "Tabu Search, Combination and Integration", in I.H. Osman and J.P. Kelly (Eds.), *Meta-Heuristics: Theory and Applications*, pp. 319-330, Kluwer Academic Publishers, Massachusetts.

Baptista, S., Oliveira, R.C. and Zuquete, E., 2002, "A Period Vehicle Routing Case Study", *European Journal of Operational Research,* Vol. 139, pp. 220-229.

Barbarosoglu, G. and Özgur, D., 1999, "A Tabu Search Algorithm for the Vehicle Routing Problem", *Computers and Operations Research*, Vol. 26, pp. 255-270.

Barbarosoglu, G., Özdamar, L. and Çevik, A., 2002, "An Interactive Approach for Hierarchical Analysis of Helicopter Logistics in Disaster Relief Operations" *European Journal of Operational Research,* Vol. 140, pp. 118-133.

Beullence, P., Muyldermans, L., Cattrysse, D. and Oudheusden, D.V., 2003, A Guided Local Search Heuristic for the Capacitated Arc Routing Problem, *European Journal of Operational Research,* Vol. 147, pp. 629-643.

Brandao, J. and Mercer, A., 1997, "A Tabu Search Algorithm for the Multi-Trip Vehicle Routing and Scheduling Problem", *European Journal of Operational Research,* Vol. 100, pp. 180-191.

Breedam, A.V., 2001, "Comparing Descent Heuristics and Metaheuristics for the Vehicle Routing Problem", *Computers and Operations Research*, Vol. 28, pp. 289-315.

Carvalho, J.M.V., 2002, "LP Models for Bin Packing and Cutting Stock Problems", *European Journal of Operational Research,* Vol. 141, pp. 253-273.

Charon, I. and Hudry, O., 1996, "Mixing Different Components of Metaheuristics", in I.H. Osman and J.P. Kelly (Eds.), *Meta-Heuristics: Theory and Applications*, pp. 589-603, Kluwer Academic Publishers, Massachusetts.

Destrochers, M., Jones, C.V., Lenstra, J.K., Savelsbergh, M.W.P. and Stougie, L., 1999, "Towards a Model and Algorithm Management System for Vehicle Routing and Scheduling Problems", *Decision Support Systems,* Vol.25, pp. 109-133.

Dimitrijevic, V. and Saric, Z., 1997, "An Efficient Transformation of the Generalized Traveling Salesman Problem into the Traveling Salesman Problem on Digraphs", Information Sciences, Vol. 102, pp. 105-110.

Ghiani, G., Guerriero, F., Laporte, G. and Musmanno, R., 2003, "Real-time Vehicle Routing: Solution Concepts, Algorithms and Parallel Computing Strategies", *European Journal of Operational Research,* (in press).

Glaab, H., 2002, "A New Variant of a Vehicle Routing Problem: Lower and Upper Bounds", *European Journal of Operational Research,* Vol. 139, pp. 557-577.

Glover, F., Gutin, G., Yeo, A. and Zverovich, A., 2001, "Construction Heuristics for the Asymmetric TSP", *European Journal of Operational Research,* Vol. 129, pp. 555-568.

Gutin, G. and Yeo, A., 2001, "TSP Tour Domination and Hamilton Cycle Decompositions of Regular Digraphs", Operations Research Letters, Vol. 28, pp. 107-111.

Hansen, G.W. and Hansen, J.V., 1996, *Database Management Design*, 2nd ed., Prentice-Hall, New Jersey.

Helsgaun, K., 2000, "An Effective Implementation of the Lin-Kernighan Traveling Salesman Heuristic", *European Journal of Operational Research,* Vol. 126, pp. 106-130.

Hwang, H.S., 2002, "An Improved Model for Vehicle Routing Problem with Time Constraint Based on Genetic Algorithm", *Computers and Industrial Engineering,* Vol. 42, pp. 361-369.

Kabadi, S.N., 2002, "New Polynomially Solvable Classes and a New Heuristic for the Traveling Salesman Problem and Its Generalization", *Discrete Applied Mathematics,* Vol. 119, pp. 149-167.

Kang, J. and Park, S., 2002, "Algorithms for the Variable Sized Bin Packing Problem", *European Journal of Operational Research,* (in press).

Kilby, P., Prosser, P. and Shaw, P., 2000, "A Comparison of Traditional and Constraint-based Heuristic Methods on Vehicle Routing Problems with Side Constraints", *Constraints*, Vol. 5, pp. 389-414.

Kim, J.U. and Kim, Y.D., 1999, "A decomposition Approach to a Multi-period Vehicle Scheduling Problem", *The International Journal of Management Science,* Vol. 27, pp. 421-430.

Laporte, G., Gendreau, M., Potvin, J.Y. and Semet, F., 2000, "Classical and Modern Heuristics for the Vehicle Routing Problem", *International Transactions in Operational Research*, Vol. 7, pp. 285-300.

Laporte, G., 1992, "The Vehicle Routing Problem: An Overview of Exact and Approximate Algorithms" *European Journal of Operational Research,* Vol. 59, pp. 345-358.

Lau, H.C., Sim, M. and Teo, K.M., 2003, "Vehicle Routing Problem with Time Windows and a Limited Number of Vehicles", *European Journal of Operational Research,* Vol. 148, pp. 559-569.

Lodi, A., Martello, S. and Monaci, M., 2002, "Two-Dimensional Packing Problems: A Survey", *European Journal of Operational Research,* Vol. 141, pp. 241-252.

Maurer, H., Scherbakov, N., Halim, Z. and Razak, Z., 1998, *From Databases to Hypermedia*, Springer-Verlag, Berlin.

Nikolakopoulou, G., Kortesis, S., Synefaki, A. and Kalfakakou, R., 2003, "Solving a Vehicle Routing Problem by Balancing the Vehicle Time Utilization", *European Journal of Operational Research* (in press).

Osman, I.H. and Wassan, N.A., 2002, "A Reactive Tabu Search Metaheuristics for the Vehicle Routing Problem with Back- Hauls", Journal of Scheduling, Vol. 5, pp. 263-285.

Rego, C. and Roucairol, C., 1996, "A Parallel Tabu Search Algorithm Using Ejection Chains for the Vehicle Routing Problem", in I.H. Osman and J.P. Kelly (Eds.), *Meta-Heuristics: Theory and Applications*, pp. 589-603, Kluwer Academic Publishers, Massachusetts.

Renaud, J. and Boctor, F.F., 2002, "A Sweep-Based Algorithm for the Fleet Size and Mix Vehicle Routing Problem", *European Journal of Operational Research,* Vol. 140, pp. 618-628.

Riccardi, G., 2001, *Principles of Database Systems with Internet and Java Applications*, Addison-Wesley, New York.

Salhi, S. and Sari, M., 1997, "A Multilevel Composite Heuristic for the Multi-Depot Vehicle Fleet Mix Problem", *European Journal of Operational Research,* Vol. 103, pp. 95-112.

Sierksma, G. and Tijssen, G.A., 1998, "Routing Helicopters for Crew Exchanges on Off-Shore Locations", *Annals of Operations Research*, Vol. 76, pp. 261-286.

Slater, A., 2002, "Specification for the Dynamic Vehicle Routing and Scheduling System", *International Journal of Transport Management,* Vol. 1, pp. 29-40.

Tarantilis, C.D. and Kiranoudis, C.T., 2002, "Using a Spatial Decision Support System for Solving the Vehicle Routing Problem", *Information and Management*, Vol. 39, pp. 359-375.

Toth, P. and Vigo, D., 1996, "Fast Local Search Algorithms for the Handicapped Persons Transportation Problem", in I.H. Osman and J.P. Kelly (Eds.), *Meta-Heuristics: Theory and Applications*, pp. 677-690, Kluwer Academic Publishers, Massachusetts.

Toth, P. and Vigo, D., 1999, "A Heuristic Algorithm for the Symmetric and Asymmetric Vehicle Routing Problems with Backhauls", *European Journal of Operational Research,* Vol. 113, pp. 528-543.

Toth, P. and Vigo, D., 2002, "An Overview of Vehicle Routing Problems", in P. Toth and D. Vigo (Eds.), *The Vehicle Routing Problem*, pp. 1-26, Society for Industrial and Applied Mathematics, Philadelphia.

Toth, P. and Vigo, D., 2002, "Models, Relaxations and Exact Approaches for the Capacitated Vehicle Routing Problem", *Discrete Applied Mathematics,* Vol. 123, pp. 487-512.

Tzoreff, T.E., Granot, D., Granot, F. and Sosic, G., 2002, "The Vehicle Routing Problem with Pickups and Deliveries on Some Special Graphs" *Discrete Applied Mathematics,* Vol. 116, pp. 193-229.

Voudouris, C. and Tsang, E., 1999, "Guided Local Search and Its Application to the Traveling Salesman Problem", *European Journal of Operational Research,* Vol. 113, pp. 469-499.

NATO RTG SAS-045
on
Computer Based Decision Support Tool for Helicopter Mission Planning in Disaster Relief and Military Operations

Annex A – TECHNICAL REPORT 1: OVERVIEW OF DECISION SUPPORT TOOL TECHNOLOGIES

Mickaël Fontaine
DSP/CAD, 16bis avenue Prieur de la cote d'Or, 94114 Arcueil, France
Tel: [33] (1) 42 31 90 58; Fax: [33] (1) 42 31 91 75

E-mail: mickael.fontaine@dga.defense.gouv.fr

A.1 INTRODUCTION

This document is prepared to fulfill the requirements of the Programme of Work (POW) of NATO SAS-045 RTG on "Computer Based Decision Support Tool for Helicopter Mission Planning in Disaster Relief and Military Operations".

The POW dictates to provide an a-priori analysis of modeling, computer (software engineering) and data collection technologies that are anticipated to support the development of NATO SAS-045 project. As it is stated in the Terms of Reference (TOR) of the afore-mentioned project, the main goal of the research is to provide the basis for developing a generic and flexible decision support tool for effective management of helicopter missions by conducting the problem analysis, investigating the concept of solutions and determining relevant technical requirements.

It is foreseen that different models, technologies and solution approaches will be utilized concurrently, and integration and interfacing will pose itself as an important technical issue within the scope of maintaining interoperability and standardization in NATO practices.

Thus, the POW states that during the Analysis Phase of the project technology surveys should be carried out on modeling, computer and software engineering and data collection technologies; geographical information systems, digital maps, mission data compilation systems; model, data, and knowledge management repositories in NATO nations. Then, the technology mapping and capability matrix can be developed using the identified current needs and capability gaps.

In this document, it is intended to provide a broad overview that will guide any potential research dealing with the design and development of a generic decision support tool within a similar NATO context, not limited to helicopter operations.

The aim of this document is not to cover in detail the optimization methods but rather to give a general idea of techniques and products. In this report, we provide information on Operations research techniques Simulation Data Analysis & Mining and Artificial intelligence in order to determine the design requirements for the helicopter mission planning decision support system. For more detailed operations research techniques' information see the references.

A.2 OPTIMIZATION OVERVIEW

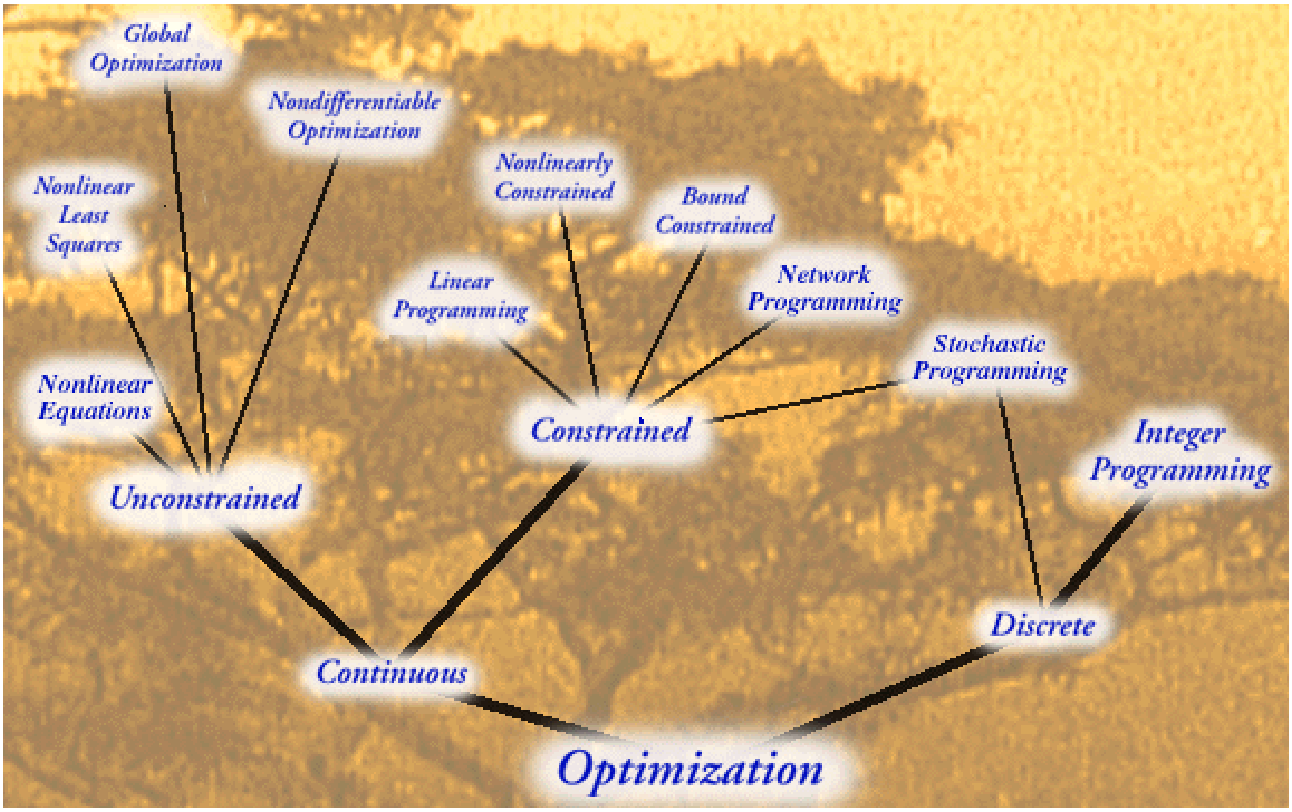


Figure A.1: A View of Different Subfields of Optimization.

This part introduces the different subfields of **optimization (or mathematical programming)** and includes outlines of the major algorithms in each area, with pointers to software packages where appropriate.

A.2.1 Introduction

Optimization problems are made up of three basic ingredients:

- An **objective function** we want to minimize or maximize. For instance, in a manufacturing process, we might want to maximize the profit or minimize the cost. In fitting experimental data to a user-defined model, we might minimize the total deviation of observed data from predictions based on the model. In designing an automobile panel, we might want to maximize the strength.
- A set of **unknowns** or **variables** affect the value of the objective function. In the manufacturing problem, the variables might include the amounts of different resources used or the time spent on each activity. In fitting-the-data problem, the unknowns are the parameters that define the model. In the panel design problem, the variables used define the shape and dimensions of the panel.
- A set of **constraints** that allow the unknowns to take on certain values but exclude others. For the manufacturing problem, it does not make sense to spend a negative amount of time on any activity, so we constrain all the "time" variables to be non-negative. In the panel design problem, we would probably want to limit the weight of the product and to constrain its shape.

The optimization problem is then:

Find values of the variables that minimize or maximize the objective function while satisfying the constraints.

Are All these ingredients necessary?

Objective Function

Almost all optimization problems have a single objective function. (When they don't they can often be reformulated so that they do!) The two interesting exceptions are:

- No objective function. In some cases (for example, design of integrated circuit layouts), the goal is to find a set of variables that satisfies the constraints of the model. The user does not particularly want to optimize anything so there is no reason to define an objective function. This type of problems is usually called a feasibility problem.
- Multiple objective functions. Often, the user would actually like to optimize a number of different objectives at once. For instance, in the panel design problem, it would be nice to minimize weight and maximize strength simultaneously. Usually, the different objectives are not compatible; the variables that optimize one objective may be far from optimal for the others. In practice, problems with multiple objectives are reformulated as single-objective problems by either forming a weighted combination of the different objectives or else replacing some of the objectives by constraints.

Variables

These are essential. If there are no variables, we cannot define the objective function and the problem constraints.

Constraints

Constraints are not essential. In fact, the field of unconstrained optimization is a large and important one for which a lot of algorithms and software are available. It's been argued that almost all problems really do have constraints. For example, any variable denoting the "number of objects" in a system can only be useful if it is less than the number of elementary particles in the known universe! In practice though, answers that make good sense in terms of the underlying physical or economic problem can often be obtained without putting constraints on the variables.

A.2.1.1 Mathematical Programming (Optimization) Theory

Many decision problems can be modeled using mathematical programs, which seek to maximize or minimize some objective which is a function of the decisions. The possible decisions are constrained by limits in resources, minimum requirements, etc. Decisions are represented by variables, which may be, for example, non-negative or integer. Objectives and constraints are functions of the variables, and problem data. Examples of problem data include unit costs, production rates, sales, or capacities.

Suppose the decisions are represented by the variables (x1, x2, ...xn). For example, x(i) can represent production of the i th of n products. The general form of a mathematical program is

Minimize $f(x_1, x_2, x_3, ..., x_n)$
subject to $g1(x_1, x_2, x_3, ..., x_n) <= 0$
…
$gm(x_1, x_2, x_3, ..., x_n) <= 0$
$x_1, x_2, x_3, ..., x_n$ in X

where X is a set that be, e.g. all non-negative real numbers. The constraints can be quite general, but linear constraints are sufficient in many cases to capture the essence of the model.

A.2.2 Linear Programming

A.2.2.1 Linear Programming Presentation

A Linear Program (LP) is a problem that can be expressed as follows (the so-called Standard Form):

minimize cx
subject to Ax = b
x > = 0

where x is the vector of variables to be solved for, A is a matrix of known coefficients, and c and b are vectors of known coefficients. The expression "cx" is called the objective function, and the equations "Ax = b" are called the constraints. All these entities must have consistent dimensions, of course, and you can add "transpose" symbols to taste. The matrix A is generally not square, hence you don't solve an LP by just inverting A. Usually A has more columns than rows, and Ax = b is therefore quite likely to be under-determined, leaving great latitude in the choice of x with which to minimize cx.

The word "Programming" is used here in the sense of "planning"; the necessary relationship to computer programming was incidental to the choice of name.

A.2.3 Non-Linear Programming

At least one of the functions (objective or constraint) is not affine (Usually, it is the rule of the function that classifies it as non-linear. In particular, a linear integer program is not generally considered an NLP.)

A.2.4 Integer Programming

Many times decision variables only have meaning when they are integers. Integer Linear Programs are similar to Linear Programs except that decision variables can only take on integer values. They are modeled like linear programs with additional constraints that variables assume integer values.

A.2.5 Mathematical Programming Products

A.2.5.1 CPLEX

The main module of CPLEX is the Linear Optimizer Base System. It is a linear programming environment which includes fast optimization algorithms as well as a full set of utilities to support solving linear programming problems. The Callable Library is an optimization library system created specifically for those with development or specialized application requirements. The Mixed Integer Solver is a high-performance optimizer for problems with integer variables while the Barrier/QP Solver is an optimizer for solving linear and quadratic programming problems available as an option with CPLEX optimization software. There are also Parallel Solvers taking full advantage of selected parallel computer architectures to deliver high solution performance.

OS: IBM-compatible PCs, Macintosh, various UNIX systems

Vendor: ILOG CPLEX Division http://www.cplex.com

A.2.5.2 XPRESS-MP

XPRESS-MP is a complete Linear and Integer Programming modelling and optimization system built around a state-of-the-art model builder and Simplex and Interior Point optimizers. XPRESS-MP is designed for the professional programmer. It helps you build, test and debug MP models on a personal computer, workstation, mini or mainframe faster than ever before. Because XPRESS-MP looks identical on a variety of hardware platforms, you can develop your application where it is most convenient for you and base your production system where it is most convenient for end users.

OS: Windows 95, Windows 98, Windows NT, UNIX, Linux, Solaris, AIX, etc.

Vendor: Dash Associates Ltd. http://www.dash.co.uk

Demo: http://www.dash.co.uk/evaluate.html

A.2.5.3 LINDO

LINDO is one of the most popular Linear Programming engines. It solves linear, integer, and quadratic optimization problems providing speedy problem entry, solution, and analysis. LINDO's straightforward, simple style makes it easy for beginners to learn. And, for developers, there's the flexibility of being able to link in your own routines, add custom commands and use the LINDO engine to run your own vehicle.

OS: Available for PC and (in Hyper and larger sizes) the following workstations: IBM, HP 9000, SunSPARC, Linux, Silicon Graphics, and DEC Alpha.

Vendor: LINDO Systems http://www.lindo.com

Demo: http://www.lindo.com/cgi-bin/frameset.cgi?leftdwnld.html;downloadf.html

A.2.5.4 OSL

OSL is high quality optimization software implementing fast, robust, state-of-the-art algorithms. The Optimization Solutions products provide stand alone solver capabilities for analyzing linear programming (including network programming) problems, quadratic programming problems, mixed integer programming problems, and stochastic programming problems.

These Solutions incorporate modules from the Optimization Library, and provide most of the functionality of the complete library without requiring the user to develop code. For linear programming and quadratic programming problems, both simplex solvers and interior point solvers are included.

The size of problems that the optimization solutions programs can handle is limited only by available memory. The Optimization Library includes about 70 user callable functions for manipulating models and analyzing the resulting mathematical problems. Besides the solvers, there are modules that analyze, simplify, and transform problems prior to solution, and others that assist with analyzing results.

These post solution analysis capabilities include: infeasibility, sensitivity, and parametrics. The modules of the Optimization Library include many program options, control variables, and user exit function calls that provide great flexibility in creating specialized applications. The Optimization Solutions modules provide access to the program options and the control variables.

OS: RISC System/6000 running AIX , PCs running Windows (NT or 95), and UNIX-based workstations from Hewlett Packard, Silicon Graphics, and Sun.

Vendor: IBM http://www6.software.ibm.com/es/oslv2/features/welcome.htm

Demo: http://www6.software.ibm.com/es/oslv2/features/download.htm

A.2.5.5 LP Solve (Non-Commercial Software)

The linear programming problem can be formulated as: Solve Ax – V_1 , with V_2 x maximal. A is a matrix, x a vector of (non-negative) variables, V_1 a vector called the right hand side, and V_2 a vector specifying the objective function.

Any number of the variables may be specified to be of type integer. This program solves problems of this kind. It is slightly more general than the above problem, in that every row of A (specifying one constraint) can have its own (in)equality, Ÿ, – or =. The result specifies values for all variables. LP Solve uses a ``Simplex'' algorithm and sparse matrix methods, for pure LP problems. If one or more of the variables is declared integer, the Simplex algorithm is iterated with a Branch-and-Bound algorithm, until the desired optimal solution is found.

OS: DOS, Windows 95

Contact: M.R.C.M. Berkelaar Eindhoven University of Technology Design Automation Section, P.O. Box 513, NL-5600 MB Eindhoven, The Netherlands, Phone: +31-40-2474792

Download: ftp://ftp.es.ele.tue.nl/pub/lp—solve

A.2.5.6 LINSOLVE (Non-Commercial Software)

LINSOLVE solves linear programming problems interactively. LINSOLVE can also be used to solve linear GOAL programming problems. The current maximum capacity of a registered program is 120 decision variables, 80 constraints, and 10 objectives. The public domain version will not handle more than 25 decision variables. LINSOLV40.EXE is a 40 column mode of LINSOLVE which can be used with an overhead projector in a classroom setting. You give your problem in an 'as is' facsimile format from a file or keyboard. The program will ask your choices for the options available. You have the choice of maximizing or minimizing, and even printing out the Simplex-tableau should you so wish. LINSOLVE optionally performs a sensitivity analysis on the coefficients of the objective function, and the right-hand sides of the constraints. LINSOLVE applies the two-stage Simplex-method.

OS: DOS

Contact: Timo Salmi, Professor of Accounting and Business Finance University of Vaasa, P.O. Box 297 SF-65101 Vaasa, Finland

Download: ftp://garbo.uwasa.fi/pc/ts/tslin35c.zip

A.2.6 Modeling Languages

A.2.6.1 Modeling Languages Presentation

Modeling systems are designed to help people formulate LPs and analyze their solutions. An LP modeling system takes as input a description of a linear program in a form that people find reasonably natural and convenient, and allows the solution output to be viewed in similar terms; conversion to the forms required by algorithmic codes is done automatically. The collection of statement forms for the input is often called a modeling language.

A.2.6.2 Modeling Languages Products

A.2.6.2.1 AMPL

AMPL, developed at AT&T's Bell Laboratories, is a powerful, yet easy-to-use, modeling environment for problems in linear, non-linear, network and integer programming. Users can formulate optimization models and analyze solutions using common algebraic notation; the computer manages the interface to advanced optimizers. AMPL is a computer language for describing production, distribution, blending, scheduling and many other kinds of problems known generally as large-scale optimization or mathematical programming. AMPL's familiar algebraic notation and interactive command environment are designed to help formulate models, communicate with a wide variety of solvers, and examine solutions.

OS: MS-DOS, Windows 3.1, Windows 95, Windows NT, SunOS 4.1.x (Solaris1.1), Solaris 2.3 or later, UNIX

Vendor: http://www.ampl.com/cm/cs/what/ampl/vendors.html

http://www.ampl.com

A.2.6.2.2 GAMS 2.50

The General Algebraic Modeling System (GAMS) is a high-level modeling system for mathematical programming problems. It consists of a language compiler and a stable of integrated high-performance solvers. GAMS is tailored for complex, large scale modeling applications, and allows you to build large maintainable models that can be adapted quickly to new situations.

OS: Windows 95, Windows 98, Windows NT, Windows 2000; UNIX: Sun Solaris, Compaq Digital UNIX, SGI irix, IBM AIX, HP hp/ux, Linux

Vendor: GAMS Development Corporation

http://www.gams.com

A.2.6.2.3 OPL Studio

ILOG OPL Studio delivers what you need to quickly and easily create advanced business optimization solutions. Its interactive graphical environment lets you develop high-level optimization applications without a detailed knowledge of computer programming. ILOG OPL Studio also makes the resources of the ILOG Optimization Suite available to you, letting you harness the power of the world's most advanced optimization tools to improve your enterprise's efficiency -and do it faster. ILOG OPL Studio will save you a lot of development time for your optimization applications, from modeling to deployment.

OS: Windows 95, Windows 98, Windows NT, Sun Solaris

Vendor: ILOG http://www.ilog.com

Demo: http://www.ilog.com/products/oplstudio/trial.cfm

A.2.6.3 Solver Interfaces with AMPL

The following table summarizes availability of solvers for which interfaces to AMPL have been constructed.

Algorithm types listed in the table are distinguished by the problems they solve and the methods they use, as follows:

- **Linear (simplex)**: Linear objective and constraints, by some version of the simplex method.
- **Linear (interior)**: Linear objective and constraints, by some version of an interior (or barrier) method.
- **Network**: Linear objective and network flow constraints, by some version of the network simplex method.
- **Quadratic**: Convex or concave quadratic objective and linear constraints, by either a simplex-type or interior-type method.
- **Non-linear convex**: Convex or concave but not all-linear objective, with linear (and possibly certain nonlinear) constraints, by an interior-type method.
- **Non-linear**: Continuous but not all-linear objective and constraints, by any of several methods including reduced gradient, quasi-newton, augmented lagrangian and interior-point.
- **Complementarity**: Linear or non-linear as above, with additional complementarity conditions.
- **Integer linear**: Linear objective and constraints and some or all integer-valued variables, by a branch-and-bound approach that applies a linear solver to successive sub-problems.
- **Integer non-linear**: Continuous but not all-linear objective and constraints and some or all integer-valued variables, by a branch-and-bound approach that applies a non-linear solver to successive sub-problems.

Table A.1: List of Mathematical Programming Products

Solver	Algorithm Types	Vendor or Download Site
BPMPD	Linear (interior)	Download from Netlib/opt
CONOPT	Non-linear	ARKI Consulting & Development A/S
CPLEX	Linear (simplex) Linear (interior) Network Quadratic Integer linear	ILOG
DONLP2	Non-linear	Download from the DONLP2 site
FortMP	Linear (simplex) Linear (interior) Quadratic Integer linear Integer quadratic	OptiRisk Systems
IPOPT	Non-linear	IPOPT homepage
KNITRO	Non-linear	KNITRO homepage
LAMPS	Linear (simplex) Integer linear	Advanced Mathematical Software
LANCELOT	Non-linear	LANCELOT site
LOQO	Linear (interior) Quadratic Non-linear	R.J. Vanderbei, Princeton Univ: LOQO homepage
LP_SOLVE	Linear (simplex) Integer linear	Download from Michel Berkelaar
MINOS	Linear (simplex) Non-linear	Stanford Business Software, Inc. (see also SOL Optimization Software)
MOSEK	Linear (simplex) Linear (interior) Quadratic, non-linear convex Integer linear Integer quadratic	MOSEK homepage
OSL	Linear (simplex) Linear (interior) Network Quadratic Integer linear	Optimal Solution Technologies (IBM)
PATH	Complementarity	CPNET Computer Sciences Dept, Univ of Wisconsin
SOPT	Linear (simplex) Quadratic Non-linear convex Integer linear	SAITECH
WSAT(OIP)	Integer linear constraints	Download from Joachim P. Walser

XA	Linear (simplex) Integer linear	Sunset Software Technology
XLSOL	Linear (simplex)	Frontline Systems, Inc.
LS-XLSOL	Quadratic Integer linear	
XPRESS	Linear (simplex) Linear (interior) Quadratic Integer linear	Dash Optimization

A.2.7 Stochastic Programming

A.2.7.1 Stochastic Programming Presentation

Stochastic programming supports decision making under uncertainty. It is a methodology for bringing uncertain future scenarios into the traditional decision making framework of linear programming. Just as linear programming models the optimal allocation of constrained resources to meet known demands, stochastic programming models the allocation of today's resources to meet tomorrow's unknown demands in such a way that the user can explore the trade offs with respect to expected risks and rewards and make informed decisions.

A.2.7.1.1 Stochastic Programs

Stochastic programs are mathematical programs where some of the data incorporated into the objective or constraints is uncertain. Uncertainty is usually characterized by a probability distribution on the parameters. Although the uncertainty is rigorously defined, in practice it can range in detail from a few scenarios (possible outcomes of the data) to specific and precise joint probability distributions. The outcomes are generally described in terms of elements w of a set W. W can be, for example, the set of possible demands over the next few months.

When some of the data is random, then solutions and the optimal objective value to the optimization problem are themselves random. A distribution of optimal decisions is generally unimplementable. Ideally, we would like one decision and one optimal objective value.

A.2.7.1.2 Recourse Models

One logical way to pose the problem is to require that we make one decision now and minimize the expected costs (or utilities) of the consequences of that decision. This paradigm is called the recourse model. Suppose x is a vector of decisions that we must take, and y(w) is a vector of decisions that represent new actions or consequences of x. Note that a different set of y's will be chosen for each possible outcome w. The Two-Stage formulation is

Minimize $f_1(x)$ + Expected Value[$f_2(y(w), w)$]
subject to $g_1(x) <= 0, \ldots g_m(x) <= 0$

h1(x, y(w)) <= 0 for all w in W
...
hk(x, y(w)) <= 0 for all w in W
x in X, y(w) in Y

The set of constraints $h_1 \dots h_k$ describe the links between the first stage decisions x and the second stage decisions y(w). Note that we require that each constraint hold with probability 1, or for each possible w in W. The functions f_2 are quite frequently themselves the solutions of mathematical problems. We don't want to make an arbitrary correction (recourse) to the first stage decision; we want to make the best such correction.

Recourse models can be extended in a number of ways. One of the most common is to include more stages. With a multistage problem, we in effect make one decision now, wait for some uncertainty to be resolved (realized), and then make another decision based on what's happened. The objective is to minimize the expected costs of all decisions taken.

A.2.7.2 Applications of Stochastic Programming

Stochastic programming has been applied to a wide variety of areas. Some of the specific problems are part of the Stochastic Programming test set. Other applications are listed below.

- Manufacturing production planning;
- Manufacturing production capacity planning;
- Electrical generation capacity planning;
- Machine scheduling;
- Freight scheduling;
- Dairy farm expansion planning;
- Macroeconomic modeling and planning;
- Timber management;
- Asset liability management;
- Portfolio selection;
- Traffic management;
- Optimal truss design; and
- Automobile dealership inventory management.

A.2.7.3 Stochastic Programming Products IBM Stochastic Solutions

IBM Stochastic Solutions offers new technology to help the user meet the demands of modeling stochastic programs. It has an easy-to-use solver that operates from the command line. It is the first commercial-grade optimization solution to implement the Stochastic Mathematical Programming System (SMPS) input format for multistage stochastic programs. The flexible decomposition solver is robust and fast, and with callable modules, the user has node-by-node access to data and solutions for efficient solution analysis. The components of IBM Stochastic Solutions are a solver, a suite of callable modules for advanced development, and the User's Guide and Reference containing documentation, problem examples, and sample drivers.

OS: Windows 95, Windows NT, AIX, HP-UX, Sun Solaris, SGI IRIX

Vendor: IBM http://www6.software.ibm.com/es/oslv2/startme.htm

A.2.7.3.1 Queue

Queue is an interactive program dedicated to basic queuing models. You can choose between the following queuing models:

> M/G/1, D/G/1, M/M/c, M/D/c, D/M/c, G/M/c queues, Finite-Capacity-Queues, Finite-Source-Queues and Transient M/M/1 Queues. (c = number of servers) In several queuing models, Erlangian and Coxian-2 services are assumed for arrivals. The information list defines the Erlangian and the Coaxian-2 distributions in detail: there is therefore also a pedagogical dimension to this program, over and above the simple solution of queuing problems.

OS: DOS

Vendor: Prof. Dr. H.C. Tijms http://www.econ.vu.nl/ectrie/nl/leden/htijms.html

A.2.8 Constraint Satisfaction Problem

A.2.8.1 Constraint Satisfaction Problem Presentation

Constraint Satisfaction Problems (CSP) have been a subject of research in Artificial Intelligence for many years. The pioneering works on networks of constraints were motivated mainly by problems arising in the field of picture processing [Waltz, Montanari]. AI research, concentrated on difficult combinatorial problems, is dated back to sixties and seventies and it has contributed to considerable progress in constraint-based reasoning. Many powerful algorithms were designed that became a basis of current constraint satisfaction algorithms.

A Constraint Satisfaction Problem (CSP) consists of:

- A set of *variables* $X = \{x_1, ..., x_n\}$;
- For each variable x_i, a finite set D_i of possible values (its *domain*); and
- A set of *constraints* restricting the values that the variables can simultaneously take.

Note that values need not be a set of consecutive integers (although often they are), they need not even be numeric.

A solution to a CSP is an assignment of a value from its domain to every variable, in such a way that every constraint is satisfied. We may want to find:

- Just one solution, with no preference as to which one;
- All solutions; and
- An optimal, or at least a good solution, given some objective function defined in terms of some or all of the variables.

Solutions to CSPs can be found by searching systematically through the possible assignments of values to variables. Search methods divide into two broad classes, those that traverse the space of partial solutions (or partial value assignments), and those that explore the space of complete value assignments (to all variables) stochastically.

The reasons for choosing to represent and solve a problem as a CSP rather than, say as a mathematical programming problem are two-fold.

Firstly, the representation as a CSP is often much closer to the original problem: the variables of the CSP directly correspond to problem entities, and the constraints can be expressed without having to be translated into linear inequalities. This makes the formulation simpler, the solution easier to understand, and the choice of good heuristics to guide the solution strategy more straightforward.

Secondly, although CSP algorithms are essentially very simple, they can sometimes find solution more quickly than if integer programming methods are used.

This tutorial is intended to give a basic grounding in constraint satisfaction problems and some of the algorithms used to solve them. In general, the tasks posed in the constraint satisfaction problem paradigm are computationally intractable (**NP-hard**).

A.2.8.2 Constraint Satisfaction Problem Techniques

A.2.8.2.1 Binarization of Constraints

A constraint can affect any number of variables form 1 to n (n is the number of variables in the problem). It is useful to distinguish two particular cases: unary and binary constraints. Since unary constraints are dealt with by pre-processing the domains of the affected variables, they can be ignored thereafter. If all the constraints of a CSP are binary, the variables and constraints can be represented in a constraint graph and the constraint satisfaction algorithm can exploit the graph search techniques. This is interesting because any constraint can be expressed in terms of binary constraints. Hence, binary CSPs are representative of all CSPs.

A.2.8.2.2 Systematic Search Algorithms

A CSP can be solved using generate-and-test paradigm (GT) that systematically generates each possible value assignment and then it tests to see if it satisfies all the constraints. A more efficient method uses the backtracking paradigm (BT) that is the most common algorithm for performing systematic search. Backtracking incrementally attempts to extend a partial solution toward a complete solution, by repeatedly choosing a value for another variable.

A.2.8.2.3 Consistency Techniques

The late detection of inconsistency is the disadvantage of GT and BT paradigms. Therefore various consistency techniques for constraint graphs were introduced to prune the search space. The consistency-enforcing algorithm makes any partial solution of a small sub-network extensible to some surrounding network. Thus, the inconsistency is detected as soon as possible. The consistency techniques range from simple node-consistency and the very popular arc-consistency to full, but expensive path consistency.

A.2.8.2.4 Constraint Propagation

By integrating systematic search algorithms with consistency techniques, it is possible to get more efficient constraint satisfaction algorithms. Improvements of backtracking algorithm have focused on two phases of the algorithm: moving forward (forward checking and look-ahead schemes) and backtracking (look-back schemes).

A.2.8.2.5 Variable and Value Ordering

The efficiency of search algorithms which incrementally attempts to extend a partial solution depends considerably on the order in which variables are considered for instantiations. Having selected the next variable to assign a value to, a search algorithm has to select a value to assign. Again, this ordering affects the efficiency of the algorithm. There exist various heuristics for dynamic or static ordering of values and variables.

A.2.8.2.6 Reducing Search

The problem of most systematic search algorithms based on backtracking is the occurrence of many "backtracks" to alternative choices which degrade the efficiency of the system. In some special cases, it is possible to completely eliminate the need for backtracking. Also, there exist algorithms which reduce the backtracking by choosing the special variable ordering.

A.2.8.2.7 Heuristics and Stochastic Algorithms

In the last few years, greedy local search strategies became popular, again. These algorithms incrementally alter inconsistent value assignments to all the variables. They use a "repair" or "hill climbing" metaphor to move towards more and more complete solutions. To avoid getting stuck at "local maxima" they are equipped with various heuristics for randomizing the search. Their stochastic nature generally voids the guarantee of "completeness" provided by the systematic search methods.

A.2.8.2.8 Benchmarking and Algorithm Analysis

It is possible to analyze the efficiency of the constraint satisfaction algorithms by traditional approaches that compute the worst-case, average, etc., complexity of the algorithm. However, the methodology for experimental evaluation of the algorithms was also developed. This methodology is based on analyzing populations of randomly-generated binary constraint satisfaction problems and it enables one to analyze the algorithm according to a given class of problems. It also helps to identify where the extremely hard problems occur.

A.2.8.3 Constraint Programming Products

The main contenders are Ilog Solver and CHIP. Ilog has a larger market share, probably because Cosytec were slow to offer CHIP as a \CPP/ library. ICL offer their CHIP-derived system, DecisionPower, to customers only as part of a consultancy package. IF/Prolog is also derived from CHIP, and both offer Boolean, rational, and finite domain (i.e. CSP) solvers. Both CHIP v5 and Ilog benefit from the addition of various sophisticated OR algorithms which allow powerful new symbolic constraints to be offered.

Ilog Solver/Scheduler: The Ilog optimization suite consists of three C++ and JAVA libraries:

- **Ilog Solver** is a basic CSP solver.
- **Ilog Scheduler** is built on top of Solver. It facilitates modelling activities and various resource types, and provides sophisticated algorithms to enforce capacity constraints.
- **Ilog Planner** uses a Simplex-like solver to facilitate production planning.

CHIP v5: Developed by the original Chip team. Prolog version or C library, C++ version available soon. No website, contact: COSYTEC.

IF/Prolog: Developed from Siemens SNI Prolog, hence another from the CHIP family.

Prolog IV: Constraint Solver for rationals, lists, and intervals. Unlike Prolog III which used the non-standard Marseille dialect of Prolog, Prolog IV is ISO compliant.

A.2.8.3.1 CHIP V5

CHIP stands for ``Constraint Handling In Prolog''. It is a new generation logic programming language combining the declarative aspects of Prolog with the efficiency of specialized constraint solving techniques. It brings together techniques from different paradigms in order to allow powerful symbolic and numerical constraint manipulation. CHIP V5, which is the second generation product in constraint programming technology, allows to develop decision support systems for solving challenging problems, in areas such as production scheduling, logistics or manpower planning. In most practical situations, these problems within a resource optimisation framework, are both very computationally complex and highly evolutive over time.

- **CHIP Technology**

In 1996, COSYTEC launched the second generation of Constraint Programming Tools with CHIP V5. The principal features of this tool – global constraints and open architecture – give COSYTEC a technological advantage acknowledged by the market.

A.2.8.3.2 Global Constraints

COSYTEC was the originator of this concept. It represents a major technological breakthrough, for several reasons:

- This level of abstraction makes it possible to model the problems in a much more concise and more realistic way;
- Calculation times and memory management are dramatically improved;
- Certain types of problem previously considered unsolvable are now solved extremely efficiently (e.g. simultaneous correction of shifts and timetables at LUFTHANSA, nuclear fuel disposal at EDF, procurement logistics for Sun Valley (UK) or scheduling of satellite missions for Alcatel ESPACE);
- Code size is considerably reduced, greatly facilitating evolutionary maintenance of the application; and
- The method for relaxation of a global constraint is defined when the constraint is set, making it possible to solve "over-constrained" problems.

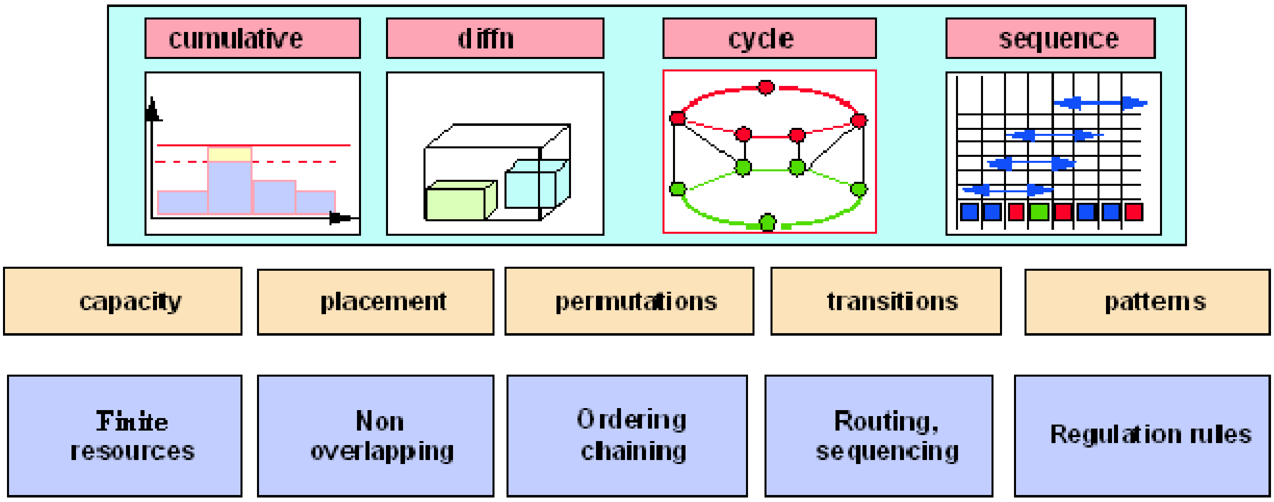


Figure A.2: The New Constraints in CHIP V5 and the Context for their Use.

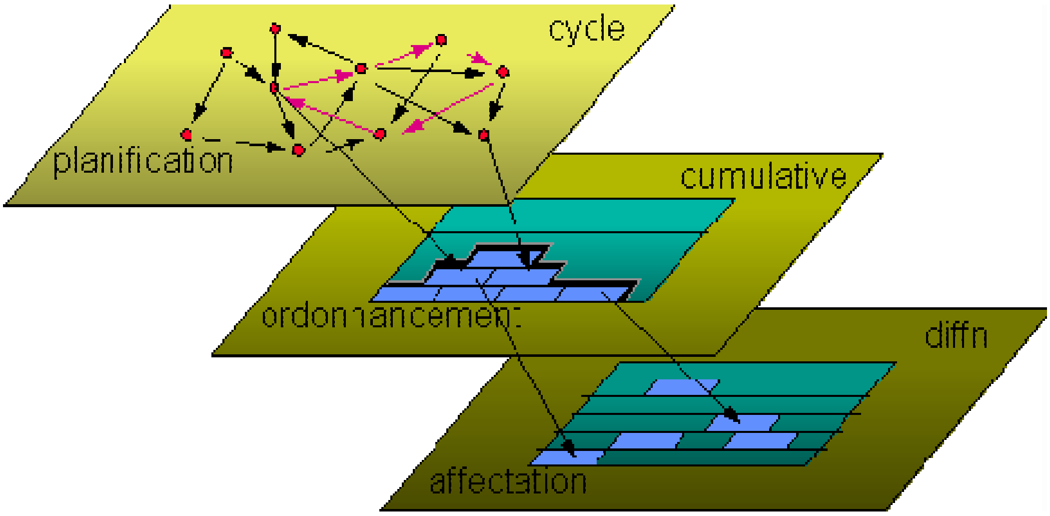


Figure A.3: Crossing of Global Constraints.

The example above illustrates how a concrete problem is represented using global constraints. By crossing them it is possible to maximize deduction and propagation, and quickly converge towards a solution.

For instance, let’s imagine a satellite planning problem:

> The **cumulative** constraint is used to model **producer/consumer** behavior, between the energy-producing resources (solar sensors), the storage facility (batteries) and the consumption of this energy; the **diffn** constraint expresses the concept of assignment (a photograph is assigned to a mission, or two photographs of neighboring regions must not overlap); finally, the **cycle** constraint expresses the time and space data concerning the missions (photographs above regions) or the transitions and chaining of activities.

- **Open Architecture**

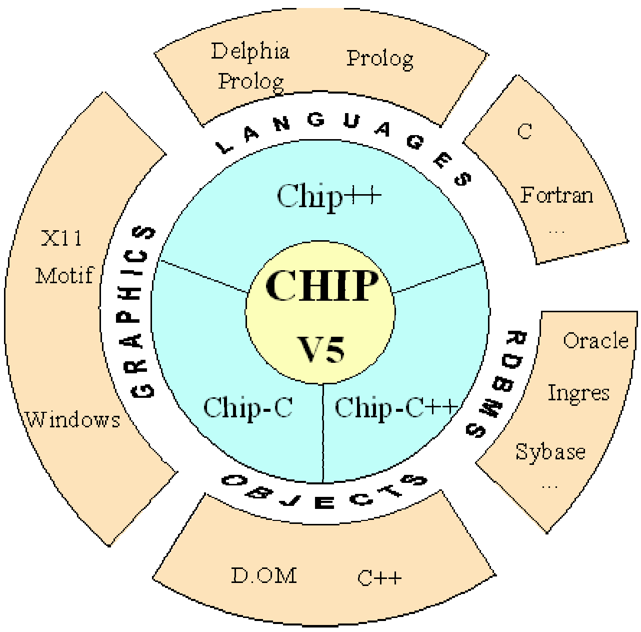


Figure A.4: CHIP Architecture.

CHIP V5's kernel is written entirely in ANSI C. The product's open architecture makes the constraint engine completely independent of the programming language (CHIP-C, CHIP-C++, CHIP++).

RAD tool (Rapid Application Development).

It is often difficult to express real-life optimization, scheduling or planning problems using a standard model, because they are complex and constantly changing.

Constraint Programming tackles and solves this problem efficiently, using an incremental approach in which the constraint model is gradually enriched. Rapid Application Development (RAD) is perfect for designing and validating mathematical models.

However to put this method into action you must have a sophisticated development environment offering fast, user-friendly debugging tools.

The XGIP and GAELL object-oriented graphic tools are linked to the solver. They further facilitate this rapid development process by enabling the developer to display the application's results in graphic form, as interactive Gantt diagrams, tables and curves.

Thanks to this tool, to create a graphic interactive mock-up of the application all you need to do is incorporate the customer's data.

This means a mock-up of the application can be created in a few days, and only two or three weeks are required to create a prototype including the solver.

This type of tool therefore makes it possible to implement a progressive approach during the project comprizing intermediate steps and successive software versions (prototype version 1, 2, etc.). This incremental methodology is ideal for setting up planning and scheduling applications, because this type of project requires user involvement at every stage, from the detailed specifications to development, experimentation and then implementation.

Different solvers included in CHIP.

The CHIP system comprizes two solvers: a solver for rational numbers and a solver on finite domains. These two solvers communicate with each other, enabling Mixed Programming.

The rational number solver in CHIP V5 is a complete solver on continuous domains, based on Gaussian elimination and the simplex method. It uses rational numbers and their exact arithmetic, and functions in incremental mode.

Because its application domain is so large, the solver on finite domains (comprising the global constraints) constitutes the central part of the CHIP V5 system.

A.2.8.3.3 Partial Search Techniques

The Partial Search Techniques recently implemented in CHIP V5 are Meta-Heuristics specially adapted to solving optimization problems. In most cases these techniques aim to find a correct solution every time (if one exists), and in a relatively short time period (e.g. less than one minute). They represent an effective alternative to proximity methods, such as taboo or simulated annealing, without any of the disadvantages (in the case of local optimums).

OS: UNIX, Open-VMS, Windows NT

Vendor: COSYTEC S.A. http://www.cosytec.fr

A.2.8.3.4 ILOG Solver and Scheduler

- **ILOG Solver is ILOG's Constraint-Based Optimization Engine**

A core engine of the ILOG Optimization Suite, this software component provides cutting-edge optimization technology for powering scheduling, sequencing, timetabling, configuration, dispatching and resource-allocation applications with logical constraints.

Available add-ons equip ILOG Solver for tackling a wide range of problems.

What is Constraint Programming? Constraint programming is a programming technology for solving complex combinatorial problems. Data representing a problem are described by domain variables. Each variable has an associated domain, which is the set of its potentially feasible values. Constraints describe the different relationships that must be met within a set of variables in order to solve a problem.

- **ILOG Scheduler Constraint-Based Scheduling**

ILOG Scheduler is a software component that supplements ILOG Solver by providing specialized modeling and algorithmic enhancements for solving problems that involve scheduling tasks over a given period of time:

  - Optimizes finite-capacity scheduling problems; and
  - Flexible, open and extensible.

A.2.9 Global Optimization

A.2.9.1 Global Optimization Presentation

Global optimization is the task of finding the absolutely best set of parameters to optimize an objective function. In general, there can solutions that are locally optimal but not globally optimal. Consequently, global optimization problems are typically quite difficult to solve exactly; in the context of combinatorial problems, they are often NP-hard. Global optimization problems fall within the broader class of non-linear programming (NLP).

Methods for global optimization problems can be categorized based on the properties of the problem that are used and the types of guarantees that the methods provide for the final solution. A NLP has the form

Minimize	$F(x)$		
subject to	$g_i(x) = 0$	for $i = 1, ..., m_1$	where $m_1 >= 0$
	$h_j(x) >= 0$	for $j = m_{1+1}, ..., m$	where $m >= m_1$

where F is a function of a vector of reals x that is subject to equality and inequality constraints. Some of the most important classes of global optimization problems are differential convex optimization, complementary problems, minimax problems, bi-linear and biconvex programming, continuous global optimization and quadratic programming.

Specific optimization methods have been developed for many classes of global optimization problems. Additionally, general techniques have been developed that appear to have applicability to a wide range of problems.

Combinatorial problems have a linear or non-linear function defined over a set of solutions that is finite but very large. There are a number of significant categories of combinatorial optimization problems, including network problems, scheduling, and transportation. If the function is piecewise linear, the combinatorial problem can be solved exactly with a mixed integer program method, which uses branch and bound. Heuristic methods like simulated annealing, tabu search and genetic algorithms have also been successfully applied to these problems to find approximate solutions.

General unconstrained problems have a non-linear function over reals that is unconstrained (or which have simple bound constraints). A variety of partitioning strategies have been proposed to solve this problem exactly. These methods typically rely on *a priori* knowledge of how rapidly the function can vary (e.g. the Lipshitz constant) or the availability of an analytic formulation of the objective function (e.g. interval methods). Statistical methods also use partitioning to decompose the search space, but they use *a priori* information (or assumptions) about how the objective function can be modeled. A wide variety of other methods have been proposed for solving these problems inexactly, including simulated annealing, genetic algorithms, clustering methods, and continuation methods, which first transform the potential function into a smoother function with fewer local minimizers, and then uses a local minimization procedure to trace the minimizers back to the original function (e.g. Moré and Wu).

General constrained problems have a non-linear function over reals that is constrained. These types of problems have not received as much attention as have general unconstrained problems. However, many of the methods for unconstrained problems have been adapted to handle constraints.

A.2.9.2 Global Optimization Techniques

- Branch and Bound;
- Mixed Integer Programming;
- Interval Methods;
- Clustering Methods;
- Evolutionary Algorithms;
- Hybrid Methods;
- Simulated Annealing;
- Statistical Methods; and
- Tabu Search.

A.2.10 Main Firms in Optimization

A.2.10.1 ILOG

Website: www.ilog.com

ILOG is the leading supplier of advanced software components. ILOG's customizable, pre-built C, C++, and Java components include optimization engines, business rule engines, and interactive user-interface engines. ILOG engines are used to cut development time of mission-critical applications while giving them new capabilities. Top-tier independent software vendors and leading corporations worldwide use ILOG products and services for Web, front-office, and back-office applications. Founded in 1987, ILOG is dually headquartered in Paris, France, and Mountain View, California, with more than 2,000 customers in 30 countries.

A.2.10.1.1 ILOG Product Overview

ILOG makes the world's most advanced software engines, including C++ and Java software components for optimization, visualization and business rules. Leading ISVs and corporations embed ILOG engines within business applications to add key functionality. ILOG technology leadership produces engines that promote faster development of smarter, more intuitive applications. ILOG is the technology and market leader in optimization engines. Major corporations and top ISVs use ILOG optimization engines within mission-critical applications that ensure the most efficient use of key resources like people, equipment, inventory and time.

- **ILOG Visualization Suite**

ILOG is the technology and market leader in advanced user-interface engines. ILOG user-interface engines are embedded in applications that manage complex systems – such as telecom networks or supply chains – as well as in distributed applications like workflow. ILOG technology speeds development time by up to 80%, and greatly increases user productivity.

- **ILOG Rules**

ILOG is the technology leader in high-performance rule engines that automate decision-making within many e-business applications. In today's constantly changing, short-turnaround Web environments, using ILOG components ensures that applications can adapt quickly, evolving ahead of the competition. ILOG Optimization SuiteILOG supplies the world's most powerful and comprehensive components for developing optimization applications.

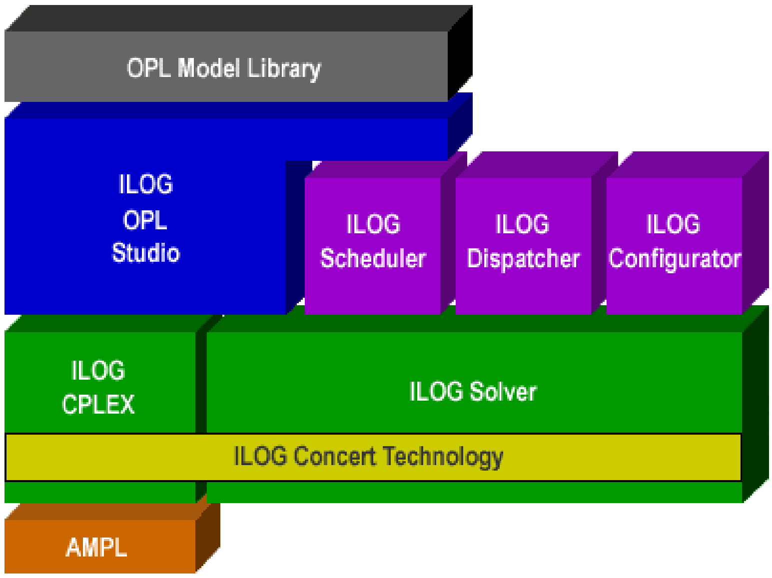


Figure A.5: ILOG Optimization Suite.

- **ILOG Optimization Suite**

Core optimization engines:

- ILOG CPLEX Mathematical programming engine
- ILOG Solver Constraint programming engine
- ILOG Concert Technology: Both ILOG CPLEX and ILOG Solver include ILOG Concert Technology

Vertical engine extensions:

- ILOG SchedulerConstraint-based scheduling
- ILOG DispatcherVehicle routing and personnel dispatching
- ILOG ConfiguratorConfiguration of products and services

Modeling tools:

- ILOG OPL StudioModeling environment for rapid development of optimization applications
- ILOG OPL Model LibraryOPL models to jump-start development
- AMPL CPLEX SystemModeling support integrated with ILOG CPLEX

- **ILOG CPLEX**

World's Leading Mathematical Programming Optimizers: ILOG CPLEX delivers high-performance, robust, flexible optimizers for solving linear, mixed-integer and quadratic programming problems in mission-critical resource allocation applications. Virtually every leading end-user and software provider in supply chain planning, telecommunication network design and transportation logistics depends on the unequaled solving power of ILOG CPLEX.

Robust algorithms for demanding problemsILOG pioneered development of the world's fastest, most robust implementations of the fundamental algorithms needed to solve today's most demanding mathematical optimization problems. ILOG CPLEX has solved problems with millions of constraints and variables, and consistently sets new records for mathematical programming software performance. Leading hardware vendors use ILOG CPLEX to measure the computational performance of their latest CPUs.

Industry-leading supportNo other vendor can match ILOG's reputation in the industry for performance, reliability, flexibility and support. ILOG's ongoing commitment to pushing the performance envelope ensures ILOG customers that their investment in ILOG CPLEX is protected well into the future.

Features

High performanceILOG CPLEX optimizers deliver the speed needed to solve huge, real-world optimization problems, as well as the quick solutions that e-commerce applications require. ILOG CPLEX is the de facto standard for mathematical optimization, with thousands of users worldwide.

Robust and reliableILOG CPLEX has been fully proven in the most demanding real-world applications. Only ILOG CPLEX provides the accuracy and dependability that key business applications demand.

Flexible ILOG CPLEX Component Libraries empower developers with the ability to embed the ILOG CPLEX engine seamlessly and efficiently in their applications. Only ILOG CPLEX provides the full range of functionality and flexibility that optimization applications require. ILOG CPLEX can be accessed from most programming environments and used on a wide variety of platforms, providing true portability.

Leaders in supply chain planning, telecommunication network design and transportation logistics depends on ILOG CPLEX optimizers for solving linear, mixed-integer and quadratic programming problems in resource-allocation applications.

- **ILOG Solver Constraint Programming Engine**

ILOG Solver is ILOG's constraint-based optimization engine.

A core engine of the ILOG Optimization Suite, this software component provides cutting-edge optimization technology for powering scheduling, sequencing, timetabling, configuration, dispatching and resource-allocation applications with logical constraints.

Available add-ons equip ILOG Solver for tackling a wide range of problems.

What is Constraint Programming? Constraint programming is a programming technology for solving complex combinatorial problems. Data representing a problem are described by domain variables. Each variable has an associated domain, which is the set of its potentially feasible values. Constraints describe the different relationships that must be met within a set of variables in order to solve a problem.

- **ILOG Scheduler Constraint-Based Scheduling**

ILOG Scheduler is a software component that supplements ILOG Solver by providing specialized modeling and algorithmic enhancements for solving problems that involve scheduling tasks over a given period of time:

- Optimizes finite-capacity scheduling problems; and
- Flexible, open and extensible.

Quick, intuitive modelingILOG Scheduler's modeling objects lets users quickly and intuitively model resources and activities, as well as temporal and capacity constraints. Its state-of-the-art algorithms rapidly and accurately compute schedules that satisfy business objectives and constraints.

Increased efficiency, reduced cost Scheduling is at the center of many strategic corporate problems, and is often the area where companies find the greatest savings. ILOG Scheduler users typically report cost savings of 20%, and a 75% reduction in operational planning time. Application development is also shortened dramatically, thanks to ILOG Scheduler's powerful, easy-to-use application program interface (API).

- **ILOG OPL Studio Rapid Development and Deployment for Optimization Applications**

ILOG OPL Studio is a complete platform for leveraging valuable resources with optimization technology. Powered by ILOG's leading optimization engines, this comprehensive modeling system expedites development and deployment.

Fast development Minimal computer programming is required to develop optimization applications, thanks to ILOG OPL Studio's intuitive optimization programming language (OPL):

- Reduce development time for a wide range of optimization problems, ranging from short-term operational scheduling to long-term economic planning;
- Experiment with both constraint programming and mathematical programming to determine the most effective method for solving optimization problems; and
- Descriptive OPL syntax produces substantially simpler code than traditional programming languages, reducing development time from weeks to days.

Smooth deployment Optimization projects advance directly from concept to implementation, drawing on OPL Component Libraries:

- Embed OPL models directly into business applications after analysis and testing;
- Connect optimization systems directly to data sources, reading business information and storing optimized solutions; and
- Solve large or complex optimization problems, utilizing support for powerful computation servers.

A.2.10.2 Dash Optimization

Website: www.dashoptimization.com

Dash develops and markets Xpress-MP, the world's leading software product for modeling and optimization. Xpress-MP solves large-scale optimization problems and enables better business decisions and resulting financial benefits in areas such as supply chain management, operations, logistics and asset management. It has been applied in sectors as diverse as manufacturing, processing, distribution, retailing, transport, finance and investment.

Dash is completely focused on optimization software and works in close partnership with its customers and partners, enabling them to get the best possible performance from Xpress-MP. Xpress-MP is embedded in many software products and solutions as a component, making leading edge optimization accessible to a wide range of customers and applications. Through expertise in the product and its application, excellent customer support and fast track product development, Dash continues to maintain Xpress-MP at the cutting edge.

A.2.10.2.1 Dash Optimization Products

- Xpress-MP – the fast track from formulation to solution
- Xpress-MP is the preferred choice for end-user applications
- Powerful yet flexible model building tools
- The fastest optimizers around
- Able to solve large problems
- Rapid development
- Xpress-MP – the superior optimization software component
- Xpress-MP is the preferred optimization component for software product developers
- Easy to integrate and Reliable
- Powerful, flexible, industry standard interfaces
- Delivers state-of-the-art technology

The Xpress-Optimizer

The Xpress-Optimizer features three optimization algorithms which enable the user to solve linear programming problems (LP), mixed integer programming problems (MIP), quadratic programming problems (QP), and mixed integer quadratic programming problems (MIQP).

The simplex optimizer, which includes primal and dual methods, solves LP problems, and is also used within a branch-and-bound framework to solve MIP and MIQP problems**. The Newton barrier optimizer** is an interior point method for solving LP and QP problems. Xpress-MP uses ultra-efficient sparse matrix handling to allow it to solve the largest problems in record time. A presolve procedure reduces the size of the problem before it is solved, sometimes by an order of magnitude. Xpress-MP is also noted for its ability to solve numerically hard or unstable problems, which is one of the reasons Xpress-MP is the clear market leader in the process industries.

The MIP/MIQP optimizer uses a sophisticated branch-and-bound algorithm to solve MIP and MIQP problems, and is particularly known for its ability to find high quality solutions fast. MIP problems can have an exponential number of possible solutions, and the essential property of the Xpress MIP optimizer is its ability to cut down the number of solutions to a manageable size, and then to navigate through them so it can find good ones quickly.

Some of the more sophisticated techniques include various classes of cutting planes, which are generated automatically during the optimization to improve the quality of bounds and reduce the size of the search (so the MIP algorithm is really called “branch-cut-cut”). The presolve is particularly effective on MIP problems, as it is able to tighten the formulation, which improves the quality of initial solutions and enables better cutting planes to be generated.

The Hyper32 edition of Xpress-MP has no internal limits on problem size, allowing the user to solve any problem which can physically be accommodated within the limits of 32-bit memory on their computer. And if that isn’t big enough, Hyper64 unleashes the power of 64-bit computing. Two restricted size editions, Extended and Professional, are also available. When it is important to solve MIP problems in the shortest possible time, or to obtain solutions for the hardest MIP problems, Xpress-Parallel is the ideal solution. Operating on multi-processor machines or across a network of computers, it enables the user to harness all of the computing power at their disposal to solve MIP problems in parallel.

A.2.10.2.2 Xpress-MP Architecture

Xpress-MP is a suite of optimization software, used to solve optimization problems. It is used in three different ways:

- As a component by ISVs to embed optimization functionality within their own products;
- By consultants to offer optimization solutions to their clients; and
- By business analysts and other end-users within large organizations to solve their own optimization problems directly.

It is used by thousands of companies worldwide to solve all sorts of problems, ranging from supply chain planning to portfolio management, from process industry refining to e-commerce.

Xpress-MP features:

- Different optimization algorithms for solving different problem types;
- Different capacities for solving problems of various sizes;
- Different modeling interfaces which allow the user to define their problem in different ways various software products and components, suitable for using Xpress-MP in an embedded system or as a stand-alone application; and
- Available on all common computer platforms.

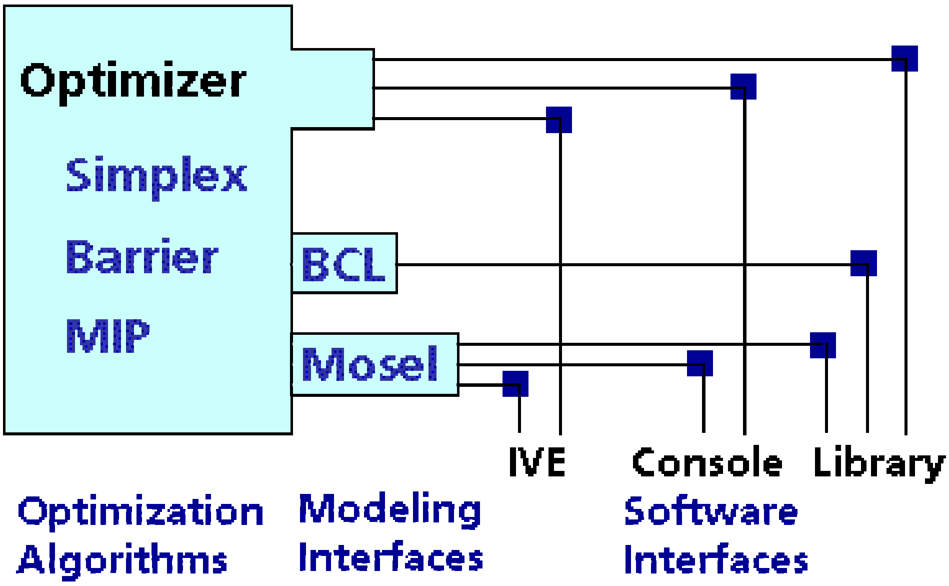


Figure A.6: Xpress-MP Architecture.

A.2.11 Decision Theoretic Approaches

A.2.11.1 Multi Criteria Analysis

- **MCA Introduction**

There are a number of methods which have been developed to provide a rational model for making decisions. The purpose of these techniques is to provide a decision maker with a rational perspective for selecting the "best" path. These techniques help a decision maker frame the known criteria and alternatives and deal with uncertainty where it may be present. The term for the body of these techniques is Decision Analysis (DA). These techniques are higher level of analysis than those oriented exclusively toward finance such as return on investment (ROI) or payback period. The objective of development in decision analysis has been to improve the ability of the human decision maker to make more timely and better quality decisions. Toward this end, extensive algorithmic techniques have been developed for properly framing the decision environment. Research has shown that the more sophisticated techniques tend to produce results that decision makers prefer – but those same techniques require considerable time to understand and to utilize. Therefore they are under utilized. At the same time education in this area needs to be expanded. Many decision makers continue to rely upon faulty, personal, inductive methods and are unaware of the efficacy of decision science methods. Fortunately packaged software is now available to help in applying these methods and for easily gathering opinion and for weighing alternatives.

- **Multi Criteria Decision Analysis Methodology**

A decision analysis technique may contain some or all of the following elements.

Table A.2: Decision Analysis Checklist

Element	Purpose is to:
Goal/Challenge	Succinctly state the ultimate end point of the challenge
Alternatives/Options/Objectives	Help in eliciting and describing the list of alternative measures or objectives to be met in attacking the challenge
Criteria/Objectives/Attributes	List such items as cost, technical, social, and political factors to be considered in the decision
Uncertainty/Preference	Provide a method for attaching information relating to the probability or expected value of an occurrence
Measurement Scales	Specifically value a criteria or alternative
Synthesis Technique	Provide for combining all the elicited and evaluated data to produce a specific answer or ranking in the alternatives
Decision Makers	Poll and apply values from multiple participants

- **MCA Techniques**

A.2.11.1.1 The Performance Matrix

A standard feature of multi-criteria analysis is a performance matrix, or consequence table, in which each row describes an option and each column describes the performance of the options against each criterion. The individual performance assessments are often numerical, but may also be expressed as 'bullet point' scores, or colour coding.

A.2.11.1.2 Scoring and Weighting

MCA techniques commonly apply numerical analysis to a performance matrix in two stages:

- **Scoring:** the expected consequences of each option are assigned a numerical score on a strength of preference scale for each option for each criterion. More preferred options score higher on the scale, and less preferred options score lower. In practice, scales extending from 0 to 100 are often used, where 0 represents a real or hypothetical least preferred option, and 100 is associated with a real or hypothetical most preferred option. All options considered in the MCA would then fall between 0 and 100.
- **Weighting:** numerical weights are assigned to define, for each criterion, the relative valuations of a shift between the top and bottom of the chosen scale.

A.2.11.1.3 Direct Analysis of the Performance Matrix

A limited amount of information about options' relative merits can be obtained by direct inspection of the performance matrix. An initial step can be to see if any of the options are dominated by others.

Dominance occurs when one option performs at least as well as another on all criteria and strictly better than the other on at least one criterion. In principle, one option might dominate all others, but in practice this is unlikely. When it does occur, it is helpful to ask if there is some advantage of the dominated option that is not represented by the criteria; this may reveal new criteria that have been overlooked. Dominance is more likely just to enable the decision-making team to eliminate dominated options from further consideration.

Once any dominance analysis has been concluded, the next stage is for the decision-making team to determine whether trade-offs between different criteria are acceptable, so that good performance on one criterion can in principle compensate for weaker performance on another. Most public decisions admit such trade-offs, but there may be some circumstances, perhaps where ethical issues are central, where trade-offs of this type are not acceptable. If it is not acceptable to consider trade-offs between criteria, then there are a limited number of non-compensatory MCA techniques available.

Where compensation is acceptable, most MCA methods involve implicit or explicit aggregation of each option's performance across all the criteria to form an overall assessment of each option, on the basis of which the set of options can be compared. The principal difference between the main families of MCA methods is the way in which this aggregation is done.

A.2.11.1.3.1 Multi-Attribute Utility Theory

There is no normative model of how individuals should make multi-criteria choices that is without critics. The one that comes closest to universal acceptance is based on multi-attribute utility theory and derives from the work of von Neumann and Morgenstern, and of Savage, in the 1940s and 1950s.

While this work provided powerful theoretical insights, it does not directly help decision makers in undertaking complex multi-criteria decision tasks. The breakthrough in this respect is the work of Keeney and Raiffa, published in 1976. They developed a set of procedures, consistent with the earlier normative foundations, which would allow decision makers to evaluate multi-criteria options in practice.

There are three building blocks for their procedures. First is the performance matrix and the second is procedures to determine whether criteria are independent of each other or not. The third consists of ways of estimating the parameters in a mathematical function which allow the estimation of a single number index, U, to express the decision maker's overall valuation of an option in terms of the value of its performance on each of the separate criteria.

The Keeney and Raiffa approach to decision support has been applied to many real decisions, in both the private and public sectors. Although well-regarded and effective, in its most general form it is relatively complex and best implemented by specialists on major projects where time and expertise are both necessary and available.

What makes the Keeney and Raiffa model potentially demanding to apply is firstly that it takes uncertainty formally into account, building it directly into the decision support models and secondly that it allows attributes to interact with each other in other than a simple, additive fashion. It does not assume mutual independence of preferences. In certain circumstances, it can be important to build into the analysis one or both of these factors, but often in practice it may be better to ignore them in order to allow a simpler and more transparent decision support to be implemented more quickly, by a wider range of users and for a larger set of problem types.

A.2.11.1.3.2 Linear Additive Models

If it can either be proved, or reasonably assumed, that the criteria are preferentially independent of each other and if uncertainty is not formally built into the MCA model, then the simple linear additive evaluation model is applicable. The linear model shows how an option's values on the many criteria can be combined into one overall value. This is done by multiplying the value score on each criterion by the weight of that criterion, and then adding all those weighted scores together. However, this simple arithmetic is only appropriate if the criteria are mutually preference independent. Most MCA approaches use this additive model, and it is the basis of the MCDA model.

Models of this type have a well-established record of providing robust and effective support to decision-makers working on a range of problems and in various circumstances. They will form the foundation for the more detailed recommendations we shall give later.

However, as was argued earlier, the variety of circumstances in which decision support has been sought has led to the development of a range of different decision support models. A number of these will now be described and related to the basic MCDA model.

A.2.11.1.3.3 The Analytical Hierarchy Process

We will describe in some detail appropriate methods to assess the scores that are the basis of the performance matrix and for judging the weights in the linear additive model.

The Analytic Hierarchy Process (AHP) also develops a linear additive model, but, in its standard format, uses procedures for deriving the weights and the scores achieved by alternatives which are based, respectively, on pairwise comparisons between criteria and between options. Thus, for example, in assessing weights, the decision maker is asked a series of questions, each of which asks how important one particular criterion is relative to another for the decision being addressed.

The strengths and weaknesses of the AHP have been the subject of substantial debate among specialists in MCA. It is clear that users generally find the pairwise comparison form of data input straightforward and convenient. This feature is exploited in MCDA by the MACBETH approach to scoring and weighting and the REMBRANDT approach. On the other hand, serious doubts have been raised about the theoretical foundations of the AHP and about some of its properties. In particular, the rank reversal phenomenon has caused concern This is the possibility that, simply by adding another option to the list of options being evaluated, the ranking of two other options, not related in any way to the new one, can be reversed. This is seen by many as inconsistent with rational evaluation of options and thus questions the underlying theoretical basis of the AHP.

A.2.11.1.3.4 Outranking Methods

A rather different approach from any of those discussed so far has been developed in France and has achieved a fair degree of application in some continental European countries. It depends upon the concept of outranking. The methods that have evolved all use outranking to seek to eliminate alternatives that are, in a particular sense, 'dominated'. However, unlike the straightforward dominance idea, dominance within the outranking frame of reference uses weights to give more influence to some criteria than others.

One option is said to outrank another if it outperforms the other on enough criteria of sufficient importance (as reflected by the sum of the criteria weights) and is not outperformed by the other option in the sense of recording a significantly inferior performance on any one criterion. All options are then assessed in terms of the extent to which they exhibit sufficient outranking with respect to the full set of options being considered as measured against a pair of threshold parameters. An explanation of precisely how outranking can identify a preferred option, or a set of preferred options for further investigation.

An interesting feature of outranking methods is that it possible, under certain conditions, for two options to be classified as 'incomparable' ('difficult to compare' is probably a better way to express the idea). Incomparability of two options is not the same as indifference between two options and might, for example, be associated with missing information at the time the assessment is made. This is not an unlikely occurrence in many decision making exercises. Building this possibility into the mathematical structure of outranking allows formal analysis of the problem to continue while neither imposing a judgement of indifference which cannot be supported nor dropping the option entirely, simply because information is not to hand.

The main concern voiced about the outranking approach is that it is dependent on some rather arbitrary definitions of what precisely constitutes outranking and how the threshold parameters are set and later manipulated by the decision maker.

The outranking concept does, however, indirectly capture some of the political realities of decision making. In particular it downgrades options that perform badly on any one criterion (which might in turn activate strong lobbying from concerned parties and difficulty in implementing the option in question). It can also be an effective tool for exploring how preferences between options come to be formed.

A.2.11.1.3.5 Procedures that Use Qualitative Data Inputs

The view taken in this manual is that reliable and transparent support for decision making is usually best achieved using numerical weights and scores on a cardinal scale. There are some exceptions, for example application of dominance and use of models that approximate the linear additive model but are based on ranking of weights. However, in general, it is a fair generalisation that the less precise the data inputs to any decision support procedure, the less precise and reliable will be the outputs that it generates.

Nonetheless, it is the case that decision makers working in government are frequently faced with circumstances where the information in the performance matrix, or about preference weights, consists of qualitative judgements. A number of methods exist to respond to this.

One group revolves around approximation to the linear additive model. In this respect they are relatively transparent, although they may involve significant amounts of data processing and, consistent with the fact that imprecise inputs rarely generate precise outputs to appraisal processes, usually require some extra assumptions to be made if, say, a single preferred option is to be identified, or even a ranking of options.

An alternative approach, largely developed in the Netherlands, has instead sought to develop procedures which amend outranking models in order to allow them to process imprecise, qualitative data inputs. They share many of the characteristics of (cardinal scale) outranking methods and have achieved only a limited degree of application, most often in urban and regional planning.

A.2.11.1.4 MCA Methods Based on Fuzzy Sets

A different response to the imprecision that surrounds much of the data on which decision making is based has been to look to the newly developing field of fuzzy sets to provide a basis for decision making models. However, methods of this type are not yet widely applied.

Fuzzy sets attempt to capture the idea that our natural language in discussing issues is not precise. Options are ‘fairly attractive’ from a particular point of view or ‘rather expensive’, not simply ‘attractive’ or ‘expensive’. Fuzzy arithmetic then tries to capture these qualified assessments using the idea of a membership function, through which an option would belong to the set of, say, ‘attractive’ options with a given degree of membership, lying between 0 and 1.

Building on assessments expressed in this way, fuzzy MCA models develop procedures for aggregating fuzzy performance levels using weights that are sometimes also represented as fuzzy quantities. However, these methods tend to be difficult for non-specialists to understand, do not have clear theoretical foundations from the perspective of modelling decision makers’ preferences and have not yet established that they have any critical advantages that are not available in other, more conventional models.

A.2.11.1.4.1 Other MCA Methods

The preceding sections have outlined some of the main types of MCA model that have been proposed as potentially applicable to public sector decision making. There are many others, some of which have a record of application, but many others which have not advanced significantly beyond the conceptual phase. Categories that have not been explicitly discussed but which are referred to in the MCA literature include methods based on Rough Sets, or on Ideal Points and several methods that are heavily dependent on interactive development, using specially constructed computer packages.

A.2.11.2 Decision Support System

A.2.11.2.1 Decision Support Systems Presentation

Decision Support Systems are computer-based systems that help decision makers make better decisions. DSS support individual and group, one-time or re-occurring, decisions that address semi-structured and unstructured problems requiring human judgement. DSS are interactive systems that incorporate data and models. Decision makers and professionals at all levels of the organization use one or more types of DSS including Ad-hoc DSS, Organizational DSS, Group DSS, and Executive Information System.

A.2.11.2.2 Decision Support Systems and Multi Criteria Decision Analysis Products

A.2.11.2.2.1 ELECTRE IS

ELECTRE IS is a generalization of ELECTRE I. It is a multicriteria method which enables to use pseudo-criteria (criteria with thresholds). Given a finite set of alternatives evaluated on a consistent family of criteria, ELECTRE IS supports the user in the process of selecting one alternative or a subset of alternatives. The method consists of two parts:

- Construction of one crisp outranking for modelling the decision-maker's preferences; and
- Exploitation of the graph corresponding to this relation.

The subset searched is the kernel of the graph.

Information: www.dauphine.lamsade.fr

A.2.11.2.2.2 ELECTRE II-IV

ELECTRE III starts with a finite set of actions evaluated on a consistent family of pseudo-criteria and aggregates these partial preferences into a fuzzy outranking relation. ELECTRE IV builds several non-fuzzy outranking relations when criteria cannot be weighted.

Two complete pre-orders are then obtained through a "distillation" procedure, either from the fuzzy outranking relation of ELECTRE III, or from the non-fuzzy outranking relations provided by ELECTRE IV. The intersection of these pre-orders indicates the most reliable part of the global preference.

Information: www.dauphine.lamsade.fr

A.2.11.2.2.3 ELECTRE TRI

ELECTRE TRI is a multiple criteria decision aiding tool designed to deal with sorting problems (segmentation problems). A segmentation problem consist in analysing the intrinsic value of each alternative (file, candidate, project, ...) in order to propose an appropriate recommendation for each of them.

Starting from a finite set of alternative evaluated on a family of quantitative and/or qualitative criteria and from a set of ordered categories corresponding to pre-defined recommendations (for example: very good,

good,… , mediums, bad), ELECTRE TRI provides the users two different procedures that assign each alternative to the categories.

In opposition with the classical procedures based on the weighted sum (possibility of compensation), the two proposed procedures in the ELECTRE TRI method refuse such possibility of total compensation among the evaluations of alternatives according to the various criteria. The assignment of an alternative is grounded on the comparison of the alternative to reference alternatives by mean of an outranking relation. These two procedures differ according to their behaviour (either pessimistic or optimistic) on the way to deal with situations in which certain alternatives are incomparable with some reference alternatives.

ELECTRE TRI Assistant is a new module in the software that enables the user to calibrate the ELECTRE TRI model indirectly, i.e. fix the model parameters by giving some examples of assignments (corresponding to desired assignments or past decisions). The values of parameter are inferred through a certain form of regression on assignment examples. Hence, ELECTRE TRI Assistant reduces the cognitive effort required from the decision maker to elicit the preferential parameters.

Information: www.dauphine.lamsade.fr

A.2.11.2.2.4 UTA PLUS

The method can be used to solve the problems of multicriteria choice and ranking on a set A of alternatives. It constructs an additive utility function from a preference weak order defined by the user on a subset Aþ of reference alternatives. The construction, based on a principle of ordinal regression, consists of solving a small LP-problem. The software proposes marginal utility functions in piecewise linear form as compatible as possible with the given weak order. It allows the user to modify interactively the marginal utility functions within limits following from a sensitivity analysis of the ordinal regression problem. For these modifications, the user is helped by a friendly graphical interface.

Information: www.dauphine.lamsade.fr

A.2.11.2.2.5 Analytica

Analytica is a visual software tool for creating, analyzing, and communicating quantitative models. Analytica provides an alternative to the spreadsheet, providing graphical influence diagrams to show qualitative structure of models, hierarchical models to organize complicated models into manageable modules, and intelligent arrays with the power to scale simple models up to handle large problems.

OS: Windows 95, Windows 98, Windows NT 4.0, Macintosh OS

Vendor: Lumina Decision Systems Inc. http://www.lumina.com

Demo: http://www.lumina.com/reg/trialanalytica.htm

A.2.11.2.2.6 Criterium DecisionPlus

Criterium DecisionPlus supports users in structuring and analyzing complex decisions between alternatives and involving multiple criteria.

Two leading methodologies for multi-criteria analysis are combined with uncertainty analysis: the Analytic Hierarchy Process and Simple Multi-Attribute Rating Technique. Based on these methodologies, the software aids the decision maker in consistently assigning relative importance to criteria, and rating alternatives against those criteria. The weighted score of each alternative against those criteria indicates how well it meets the decision criteria. Graphic analysis windows aid the user in analyzing the decision for reasonableness, robustness and tradeoffs.

Minimum requirements may be captured using the software's rule syntax, e.g. "applicants experience must be at least 3 years". By assigning probability distributions to capture uncertain data, the user knows not only which alternative best meets their criteria, but how likely that alternative is to be truly the best choice. The outcomes based on the assigned uncertainties are overlaid on the results based on the single-valued estimates to give a powerful visual representation of the impact of uncertain data on the decision process. The software analyzes which information's uncertainty should be reduced to best clarify the decision.

OS: Windows 3.1, Windows 95, Windows 98, Windows NT

Vendor: InfoHarvest, Inc. http://www.infoharvest.com

Demo: http://www.infoharvest.com/infoharv/CDPFreeDownloads.htm

A.2.11.2.2.7 DPL

Key features: DPL (Decision Programming Language) is decision analysis software developed to meet the requirements of decision-makers in business and government. DPL offers an advanced synthesis of the two major decision-making tools, influence diagrams and decision trees, which assist in structuring complete and focused analyses. DPLs powerful solution algorithms and many graphical outputs provide comprehensive and insightful results. DPL is currently being used by over 400 companies, government agencies, universities and research institutes in 31 countries.

OS: Windows 95, Windows NT

Vendor: Applied Decision Analysis. http://www.adainc.com

A.2.11.2.3 Expert Choice

Expert Choice for Windows is a multi-criteria decision support software tool based on the world's most popular decision-making methodology: the Analytic Hierarchy Process (AHP). With Expert Choice, defining your goals, identifying the criteria and alternatives, and evaluating key tradeoffs is a straight-forward and thorough process. It assists you in building a model for your decision, then leads you in judging, via pair-wise comparisons, the relative importance of the variables. Expert Choice then synthesizes your judgments to arrive at a conclusion and allows you to examine how changing the weighting of your criteria affects your outcome.

OS: Windows 3.11, Windows 95, Windows 98, Windows NT

Vendor: Palisade Corporation http://www.palisade.com

A.3 SIMULATION

Simulation is a powerful tool for those who want to analyze, design, and operate complex systems. It lets users create models of real-world processes which are too complex to be analyzed by spreadsheets or flowcharts. It is a cost-effective means of exploring new processes, without having to resort to pilot programs. And it is an efficient communication tool, showing how an operation works while stimulating creative thinking about how it can be improved. Simulation is used in industry, government, and educational institutions to shorten the design cycle, reduce costs, and enhance knowledge.

A.3.1 Introduction

Computer system users, administrators, and designers usually have a goal of highest performance at lowest cost. Modeling and simulation of system design trade off is good preparation for design and engineering decisions in real world jobs.

A computer simulation can be effectively used in a particular scenario. In addition to its use as a tool to better understand and optimize performance and/or reliability of systems, simulation is also extensively used to verify the correctness of designs. Most if not all digital integrated circuits manufactured today are first extensively simulated before they are manufactured to identify and correct design errors. Simulation early in the design cycle is important because the cost to repair mistakes increases dramatically the later in the product life cycle that the error is detected. Another important application of simulation is in developing "virtual environments", e.g. for training. Analogous to the holodeck in the popular science-fiction television program Star Trek, simulations generate dynamic environments with which users can interact "as if they were really there." Such simulations are used extensively today to train military personnel for battlefield situations, at a fraction of the cost of running exercises involving real tanks, aircraft, etc.

Dynamic modeling in organizations is the collective ability to understand the implications of change over time. This skill lies at the heart of successful strategic decision process. The availability of effective visual modeling and simulation enables the analyst and the decision-maker to boost their dynamic decision by rehearsing strategy to avoid hidden pitfalls.

System Simulation is the mimicking of the operation of a real system, such as the day-to-day operation of a bank, or the value of a stock portfolio over a time period, or the running of an assembly line in a factory, or the staff assignment of a hospital or a security company, in a computer. Instead of building extensive mathematical models by experts, the readily available simulation software has made it possible to model and analyze the operation of a real system by non-experts, who are managers but not programmers.

A simulation is the execution of a model, represented by a computer program that gives information about the system being investigated. The simulation approach of analyzing a model is opposed to the analytical approach, where the method of analyzing the system is purely theoretical. As this approach is more reliable, the simulation approach gives more flexibility and convenience. The activities of the model consist of events, which are activated at certain points in time and in this way affect the overall state of the system. The points in time that an event is activated are randomized, so no input from outside the system is required. Events exist autonomously and they are discrete so between the execution of two events nothing happens.

In the field of simulation, the concept of "principle of computational equivalence" has beneficial implications for the decision-maker. Simulated experimentation accelerates and replaces effectively the "wait and see" anxieties in discovering new insight and explanations of future behavior of the real system.

A.3.2 Topics in Descriptive Simulation Modeling

A.3.2.1 Modeling and Simulation

Simulation in general is to pretend that one deals with a real thing while really working with an imitation. In operations research the imitation is a computer model of the simulated reality. A flight simulator on a PC is also a computer model of some aspects of the flight: it shows on the screen the controls and what the "pilot" (the youngster who operates it) is supposed to see from the "cockpit" (his armchair).

A.3.2.1.1 Why to Use Models?

To fly a simulator is safer and cheaper than the real airplane. For precisely this reason, models are used in industry commerce and military: it is very costly, dangerous and often impossible to make experiments with real systems. Provided that models are adequate descriptions of reality (they are valid), experimenting with them can save money, suffering and even time.

A.3.2.1.2 When to Use Simulation?

Systems that change with time, such as a gas station where cars come and go (called dynamic systems) and involve randomness. Nobody can guess at exactly which time the next car should arrive at the station, are good candidates for simulation. Modeling complex dynamic systems theoretically need too many simplifications and the emerging models may not be therefore valid. Simulation does not require that many simplifying assumptions, making it the only tool even in absence of randomness.

A.3.2.1.3 How to Simulate?

Suppose we are interested in a gas station. We may describe the behavior of this system graphically by plotting the number of cars in the station; the state of the system. Every time a car arrives the graph increases by one unit while a departing car causes the graph to drop one unit. This graph (called sample path), could be obtained from observation of a real station, but could also be artificially constructed. Such artificial construction and the analysis of the resulting sample path (or more sample paths in more complex cases) consists of the simulation.

Types of Simulations:

- **Discrete event.** The above sample path consisted of only horizontal and vertical lines, as car arrivals and departures occurred at distinct points of time, what we refer to as events. Between two consecutive events, nothing happens – the graph is horizontal. When the number of events are finite, we call the simulation “discrete event.”

In some systems the state changes all the time, not just at the time of some discrete events. For example, the water level in a reservoir with given in and outflows may change all the time. In such cases “continuous simulation” is more appropriate, although discrete event simulation can serve as an approximation.

Further consideration of discrete event simulations.

A.3.2.1.4 How is Simulation Performed?

Simulations may be performed manually. Most often, however, the system model is written either as a computer program or as some kind of input into simulator software.

System terminology:

- **State**: A variable characterizing an attribute in the system such as level of stock in inventory or number of jobs waiting for processing.
- **Event**: An occurrence at a point in time which may change the state of the system, such as arrival of a customer or start of work on a job.
- **Entity**: An object that passes through the system, such as cars in an intersection or orders in a factory. Often an event (e.g. arrival) is associated with an entity (e.g. customer).

- **Queue**: A queue is not only a physical queue of people, it can also be a task list, a buffer of finished goods waiting for transportation or any place where entities are waiting for something to happen for any reason.
- **Creating**: Creating is causing an arrival of a new entity to the system at some point in time.
- **Scheduling**: Scheduling is the act of assigning a new future event to an existing entity.
- **Random variable**: A random variable is a quantity that is uncertain, such as inter-arrival time between two incoming flights or number of defective parts in a shipment.
- **Random variate**: A random variate is an artificially generated random variable.
- **Distribution**: A distribution is the mathematical law which governs the probabilistic features of a random variable.

A.3.2.2 Development of Systems Simulation

Discrete event systems (DES) are dynamic systems which evolve in time by the occurrence of events at possibly irregular time intervals. DES abound in real-world applications. Examples include traffic systems, flexible manufacturing systems, computer-communications systems, production lines, coherent lifetime systems, and flow networks. Most of these systems can be modeled in terms of discrete events whose occurrence causes the system to change from one state to another. In designing, analyzing and operating such complex systems, one is interested not only in performance evaluation, but also in sensitivity analysis and optimization.

A typical stochastic system has a large number of control parameters that can have a significant impact on the performance of the system. To establish a basic knowledge of the behavior of a system under variation of input parameter values and to estimate the relative importance of the input parameters, sensitivity analysis applies small changes to the nominal values of input parameters. For systems simulation, variations of the input parameter values cannot be made infinitely small. The sensitivity of the performance measure with respect to an input parameter is therefore defined as (partial) derivative.

Sensitivity analysis is concerned with evaluating sensitivities (gradients, Hessian, etc.) of performance measures with respect to parameters of interest. It provides guidance for design and operational decisions and plays a pivotal role in identifying the most significant system parameters, as well as bottleneck subsystems. I have carried out research in the fields of sensitivity analysis and stochastic optimization of discrete event systems with an emphasis on computer simulation models. This part of lecture is dedicated to the estimation of an entire response surface of complex discrete event systems (DES) from a single sample path (simulation), such as the expected waiting time of a customer in a queuing network, with respect to the controllable parameters of the system, such as service rates, buffer sizes and routing probabilities. With the response surfaces at hand, we are able to perform sensitivity analysis and optimization of a DES from a single simulation, that is, to find the optimal parameters of the system and their sensitivities (derivatives), with respect to uncontrollable system parameters, such as arrival rates in a queuing network. We identified three distinct processes. Descriptive Analysis includes: Problem Identification and Formulation, Data Collection and Analysis, Computer Simulation Model Development, Validation, Verification and Calibration, and finally Performance Evaluation. Prescriptive Analysis: Optimization or Goal Seeking. These are necessary components for Post-prescriptive Analysis: Sensitivity, and What-If Analysis. The prescriptive simulation attempts to use simulation to prescribe decisions required to obtain specified results. It is subdivided into two topics- Goal Seeking and Optimization. Recent developments on “single-run” algorithms for the needed sensitivities (i.e. gradient, Hessian, etc.) make the prescriptive simulation feasible.

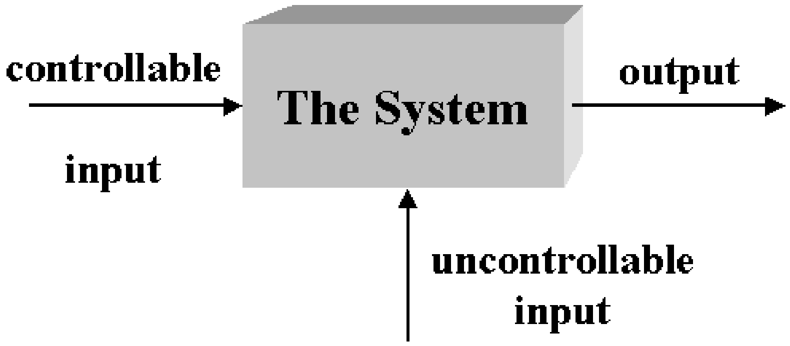


Figure A.7: Generic Design of DES.

A.3.2.2.1 Problem Formulation

Identify controllable and uncontrollable inputs. Identify constraints on the decision variables. Define measure of system performance and an objective function. Develop a preliminary model structure to interrelate the inputs and the measure of performance.

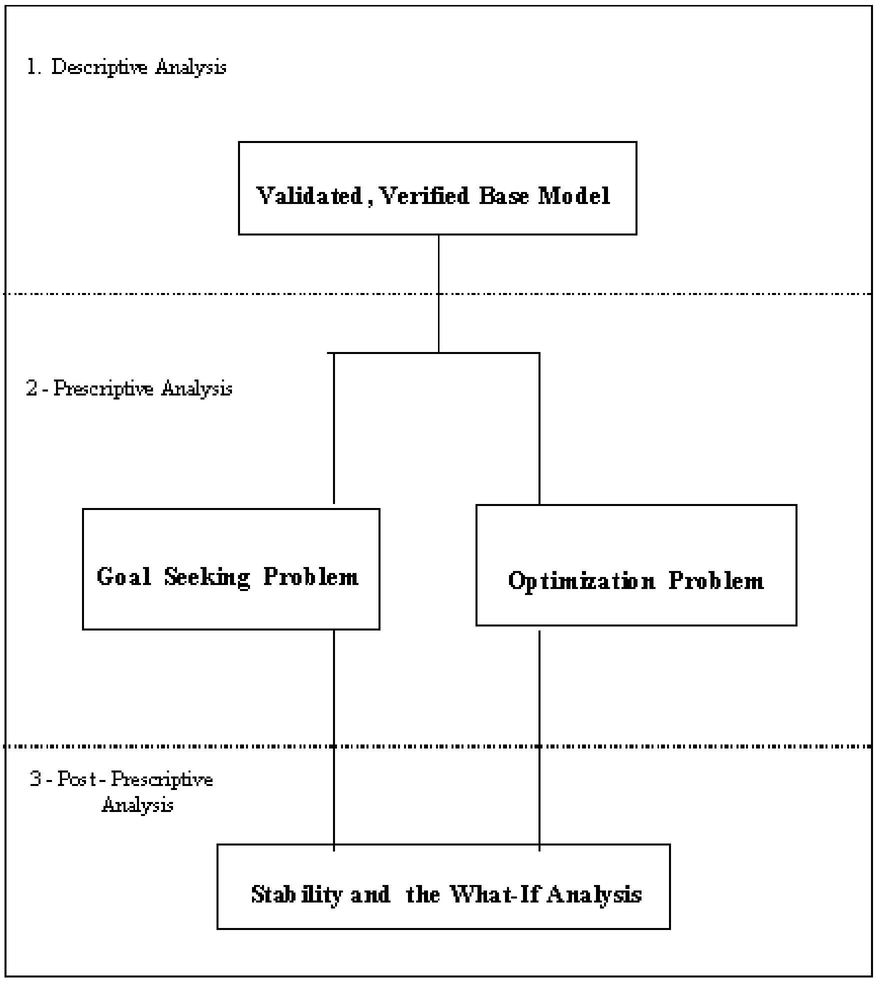


Figure A.8: A Development Process for Systems Simulation.

A.3.2.2.2 Data Collection and Analysis

Regardless of the method used to collect the data, the decision of how much to collect is a trade-off between cost and accuracy.

A.3.2.2.3 Simulation Model Development

Acquiring sufficient understanding of the system to develop an appropriate conceptual, logical and then simulation model is one of the most difficult tasks in simulation analysis.

A.3.2.2.4 Model Validation, Verification and Calibration

In general, verification focuses on the internal consistency of a model, while validation is concerned with the correspondence between the model and the reality. The term validation is applied to those processes which seek to determine whether or not a simulation is correct with respect to the "real" system. More prosaically, validation is concerned with the question "Are we building the right system?". Verification, on the other hand, seeks to answer the question "Are we building the system right?" Verification checks that the implementation of the simulation model (program) corresponds to the model. Validation checks that the model corresponds to reality. Calibration checks that the data generated by the simulation matches real (observed) data.

Validation: The process of comparing the model's output with the behavior of the phenomenon. In other words: comparing model execution to reality (physical or otherwise).

Verification: The process of comparing the computer code with the model to ensure that the code is a correct implementation of the model.

Calibration: The process of parameter estimation for a model. Calibration is a tweaking/tuning of existing parameters and usually does not involve the introduction of new ones, changing the model structure. In the context of optimization, calibration is an optimization procedure involved in system identification or during experimental design.

A.3.2.2.5 Input and Output Analysis

Discrete-event simulation models typically have stochastic components that mimic the probabilistic nature of the system under consideration. Successful input modeling requires a close match between the input model and the true underlying probabilistic mechanism associated with the system. The input data analysis is to model an element (e.g. arrival process, service times) in a discrete-event simulation given a data set collected on the element of interest. This stage performs intensive error checking on the input data, including external, policy, random and deterministic variables. System simulation experiment is to learn about its behavior. Careful planning, or designing, of simulation experiments is generally a great help, saving time and effort by providing efficient ways to estimate the effects of changes in the model's inputs on its outputs. Statistical experimental-design methods are mostly used in the context of simulation experiments.

Performance Evaluation and What-If Analysis: The `what-if' analysis is at the very heart of simulation models.

A.3.2.2.6 Sensitivity Estimation

Users must be provided with affordable techniques for sensitivity analysis if they are to understand which relationships are meaningful in complicated models.

A.3.2.2.7 Optimization

Traditional optimization techniques require gradient estimation. As with sensitivity analysis, the current approach for optimization requires intensive simulation to construct an approximate surface response function. Incorporating gradient estimation techniques into convergent algorithms such as Robbins-Monroe type algorithms for optimization purposes, will be considered.

A.3.2.2.8 Gradient Estimation Applications

There are a number of applications which measure sensitivity information, (i.e. the gradient, Hessian), Local information, Structural properties, Response surface generation, Goal-seeking problem, Optimization, What-if Problem, and Meta-modelling.

A.3.2.2.9 Report Generating

Report generation is a critical link in the communication process between the model and the end user.

A.3.2.3 Simulation Software Selection

The vast amount of simulation software available can be overwhelming for the new users. The following are only a random sample of software in the market today:

> ACSL, APROS, ARTIFEX, Arena, AutoMod, C++SIM, CSIM, Call$im, FluidFlow, GPSS, Gepasi, JavSim, MJX, MedModel, Mesquite, Multiverse, NETWORK, OPNET Modeler, POSES++, Simulat8, Powersim, QUEST, REAL, SHIFT, SIMPLE++, SIMSCRIPT, SLAM, SMPL, SimBank, SimPlusPlus, TIERRA, Witness, and javasim.

There are several things that make an ideal simulation package. Some are properties of the package, such as support, reactivity to bug notification, interface, etc. Some are properties of the user, such as their needs, their level of expertise, etc. For these reasons asking which package is best is a sudden failure of judgment. The first question to ask is for what purpose you need the software? Is it for education, teaching, student-projects or research?

The main question is: What are the important aspects to look for in a package? The answer depends on specific applications. However some general criteria are: Input facilities, Processing that allows some programming, Optimization capability, Output facilities, Environment including training and support services, Input-output statistical data analysis capability, and certainly the Cost factor.

You must know which features are appropriate for your situation, although, this is not based on a "Yes" or "No" judgment.

A.3.2.4 System Dynamics and Discrete Event Simulation

The modeling techniques used by system dynamics and discrete event simulations are often different at two levels: The modeler way of representing systems might be different, the underlying simulators' algorithms are also different. Each technique is well tuned to the purpose it is intended. However, one may use a discrete event approach to do system dynamics and vice versa.

Traditionally, the most important distinction is the purpose of the modeling. The discrete event approach is to find, e.g. how many resources the decision maker needs such as how many trucks, and how to arrange the resources to avoid bottlenecks, i.e. excessive of waiting lines, waiting times, or inventories. While the system dynamics approach is to prescribe for the decision making to, e.g. timely respond to any changes, and how to change the physical structure, e.g. physical shipping delay time, so that inventories, sales, production, etc.

System dynamics is the rigorous study of problems in system behavior using the principles of feedback, dynamics and simulation. In more words system dynamics is characterized by:

- Searching for useful solutions to real problems, especially in social systems (businesses, schools, governments, ...) and the environment.
- Using computer simulation models to understand and improve such systems.
- Basing the simulation models on mental models, qualitative knowledge and numerical information.
- Using methods and insights from feedback control engineering and other scientific disciplines to assess and improve the quality of models.
- Seeking improved ways to translate scientific results into achieved implemented improvement.

Systems dynamics approach looks at systems at a very high level so is more suited to strategic analysis. Discrete event approach may look at subsystems for a detailed analysis and is more suited, e.g. to process re-engineering problems.

Systems dynamics is indicative, i.e. helps us understand the direction and magnitude of effects (i.e. where in the system do we need to make the changes), whereas discrete event approach is predictive (i.e. how many resources do we need to achieve a certain goal of throughout).

Systems dynamics analysis is continuous in time and it uses mostly deterministic analysis, whereas discrete event process deals with analysis in a specific time horizon and uses stochastic analysis.

Some interesting and useful areas of system dynamics modeling approach are:

- Short-term and long term forecasting of agricultural produce with special reference to field crops and perennial fruits such as grapes, which have significant processing sectors of different proportions of total output where both demand and supply side perspectives are being considered.
- Long term relationship between the financial statements of balance sheet, income statement and cash flow statement balanced against scenarios of the stock market's need to seek a stable/ growing share price combined with a satisfactory dividend and related return on shareholder funds policy.
- Managerial applications include the development and evaluation of short-term and long-term strategic plans, budget analysis and assessment, business audits and benchmarking.
- A modeler must consider both as complementary tools to each other. Systems dynamic to look at the high level problem and identify areas which need more detailed analysis. Then, use discrete event modeling tools to analyze (and predict) the specific areas of interest.

A.3.2.5 Parallel and Distributed Simulation

The increasing size of the systems and designs requires more efficient simulation strategies to accelerate the simulation process. Parallel and distributed simulation approaches seem to be a promising approach in this direction. Current topics under extensive research are:

- Synchronization, scheduling, memory management, randomized and reactive/adaptive algorithms, partitioning and load balancing.
- Synchronization in multi-user distributed simulation, virtual reality environments, HLA, and interoperability.
- System modeling for parallel simulation, specification, re-use of models/code, and parallelizing existing simulations.

- Language and implementation issues, models of parallel simulation, execution environments, and libraries.
- Theoretical and empirical studies, prediction and analysis, cost models, benchmarks, and comparative studies.
- Computer architectures, VLSI, telecommunication networks, manufacturing, dynamic systems, and biological/social systems.
- Web based distributed simulation such as multimedia and real time applications, fault tolerance, implementation issues, use of Java, and CORBA.

A.3.3 Simulation-Based Optimization Techniques

Discrete event simulation is the primary analysis tool for designing complex systems. Simulation, however, must be linked with a optimization techniques to be effectively used for systems design. We present several optimization techniques involving both continuous and discrete controllable input parameters subject to a variety of constraints. The aim is to determine the techniques most promising for a given simulation model.

Many man-made systems can be modeled as Discrete Event Systems (DES); examples are computer systems, communication networks, flexible manufacturing systems, production assembly lines, and traffic transportation systems. DES evolve with the occurrence of discrete events, such as the arrival of a job or the completion of a task, in contrast with continuously variable dynamic processes such as aerospace vehicles, which are primarily governed by differential equations. Owing to the complex dynamics resulting from stochastic interactions of such discrete events over time, the performance analysis and optimization of DES can be difficult tasks. At the same time, since such systems are becoming more widespread as a result of modern technological advances, it is important to have tools for analyzing and optimizing the parameters of these systems.

Analyzing complex DES often requires computer simulation. In these systems, the objective function may not be expressible as an explicit function of the input parameters; rather, it involves some performance measures of the system whose values can be found only by running the simulation model or by observing the actual system. On the other hand, due to the increasingly large size and inherent complexity of most man-made systems, purely analytical means are often insufficient for optimization. In these cases, one must resort to simulation, with its chief advantage being its generality, and its primary disadvantage being its cost in terms of time and money. Even though, in principle, some systems are analytically tractable, the analytical effort required to evaluate the solution may be so formidable that computer simulation becomes attractive. While the price for computing resources continue to dramatically decrease, one nevertheless can still obtain only a statistical estimate as opposed to an exact solution. For practical purposes, this is quite sufficient.

These man-made DES are costly, and therefore it is important to operate them as efficiently as possible. The high cost makes it necessary to find more efficient means of conducting simulation and optimizing its output. We consider optimizing an objective function with respect to a set of continuous and/or discrete controllable parameters subject to some constraints.

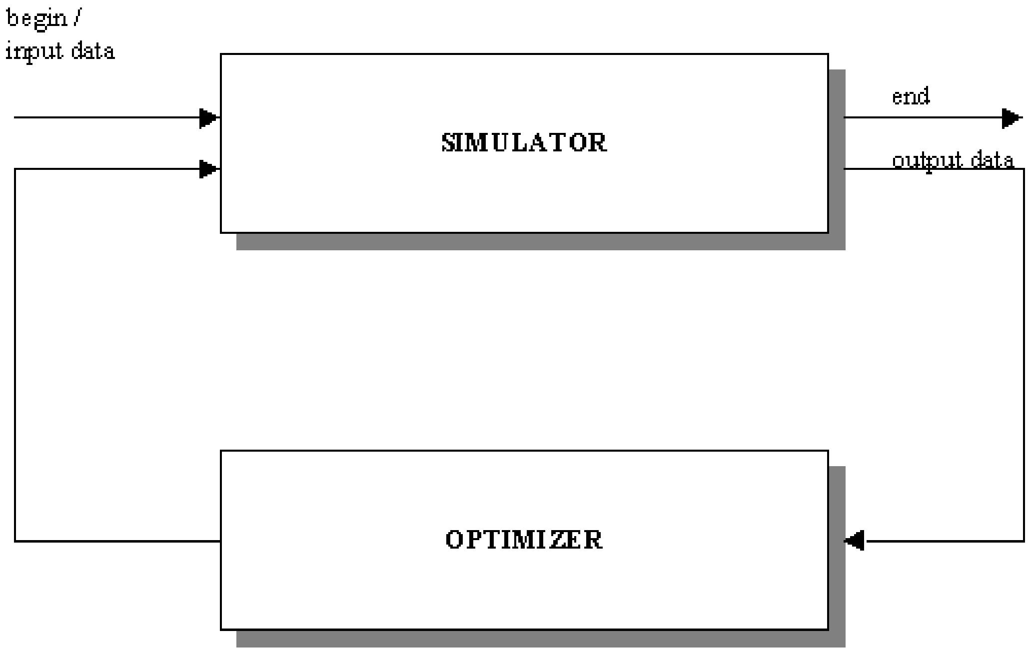


Figure A.9: Integration Scheme of Simulation and Optimization.

In almost all simulation models, an expected value can express the system's performance. Consider a system with continuous parameter v V, where V is the feasible region. Let

$$J(v) = E_{Y\,|\,v}\,[Z\,(Y)]$$

be the steady state expected performance measure, where Y is a random vector with known probability density function (pdf), f(y; v) depends on v, and Z is the performance measure.

In discrete event systems, Monte Carlo simulation is usually needed to estimate J(v) for a given value $v = v_0$. By the law of large numbers

$$J(v_0) = 1/n\ \Sigma\ Z\,(y_i)$$

converges to the true value, where y_i, i = 1, 2, ..., n are independent, identically distributed, random vector realizations of Y from f (y; v_0), and n is the number of independent replications. The aim is to optimize J(v) with respect to v.

We shall group the optimization techniques for simulation into seven broad categories; namely, Deterministic Search, Pattern Search, Probabilistic Search, Evolutionary Techniques, Stochastic Approximation, Gradient Surface, and some Mixtures of the these techniques.

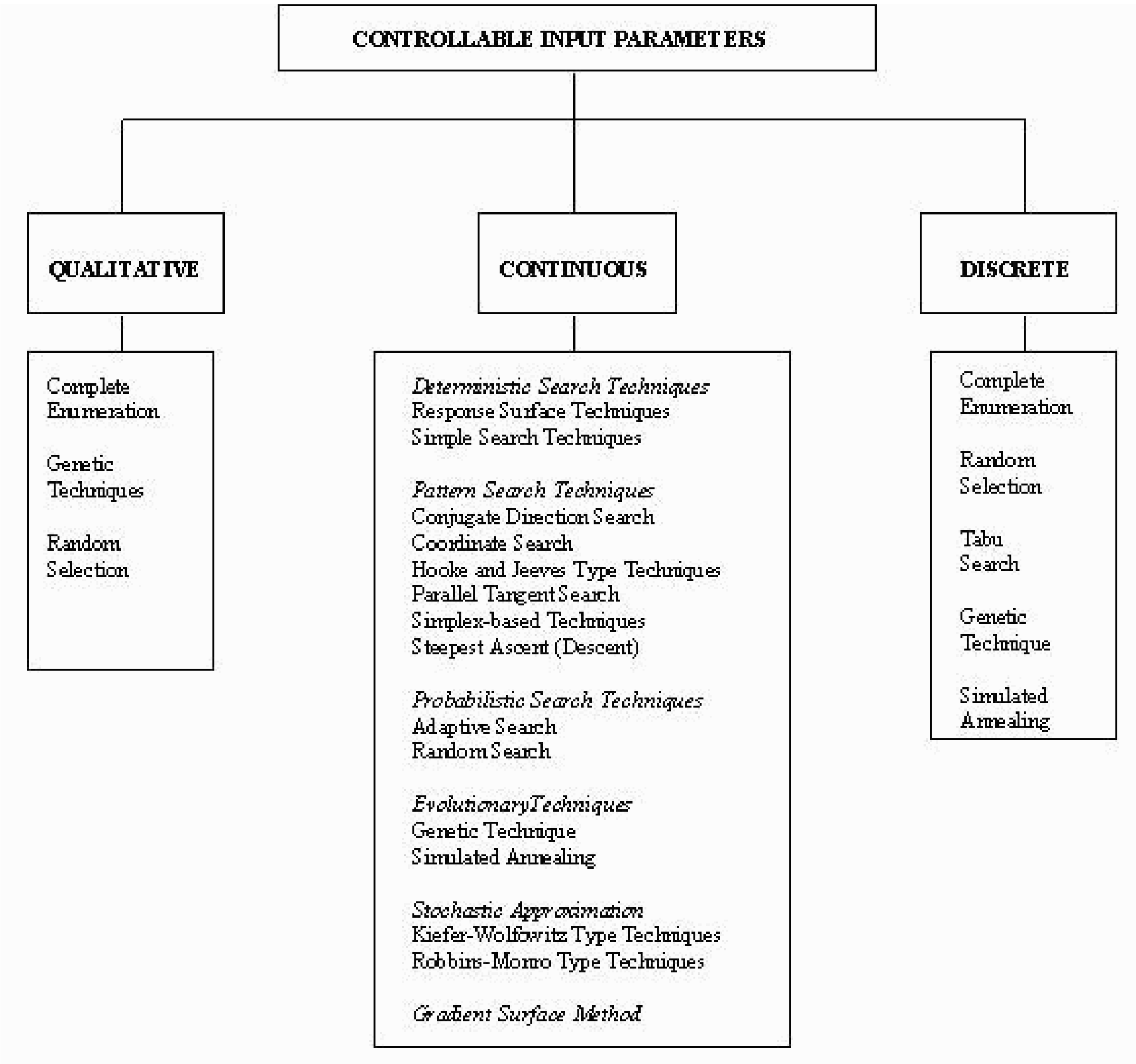


Figure A.10: A Classification of Optimization Techniques via Simulation.

A.3.3.1 Deterministic Search Techniques

A common characteristic of deterministic search techniques is that they are basically borrowed from deterministic optimization techniques. The deterministic objective function value required in the technique is now replaced with an estimate obtained from simulation. By having a reasonably accurate estimate, one hopes that the technique will perform well.

Deterministic search techniques include heuristic search, complete enumeration, and random search techniques.

A.3.3.1.1 Heuristic Search Technique

The heuristic search technique is probably most commonly used in optimizing response surfaces. It is also the least sophisticated scheme mathematically, and it can be thought of as an intuitive and experimental approach. The analyst determines the starting point and stopping rule based on previous experience with the system. After setting the input parameters (factors) to levels that appear reasonable, the analyst makes a simulation run with the factors set at those levels and computes the value of the response function. If it appears to be a maximum (minimum) to the analyst, the experiment is stopped. Otherwise the analyst changes parameter settings and makes another run. This process continues until the analyst believes that the output has been optimized. Suffice it to say that, if the analyst is not intimately familiar with the process being simulated, this procedure can turn into a blind search and can expend an inordinate amount of time and computer resources without producing results commensurate with input. The heuristic search can be ineffective and inefficient in the hand of a novice.

A.3.3.1.2 Complete Enumeration and Random Techniques

The complete enumeration technique is not applicable to continuous cases, but in discrete space v it does yield the optimal value of the response variable. All factors (v) must assume a finite number of values for this technique to be applicable. Then, a complete factorial experiment is run. The analyst can attribute some degree of confidence to the determined optimal point when using this procedure. Although the complete enumeration technique yields the optimal point, it has a serious drawback. If the number of factors or levels per factor is large, the number of simulation runs required to find the optimal point can be exceedingly large. For example, suppose that an experiment is conducted with three factors having three, four, and five levels, respectively. Also suppose that five replications are desired to provide the proper degree of confidence. Then 300 runs of the simulator are required to find the optimal point. Hence, this technique should be used only when the number of unique treatment combinations is relatively small or a run takes little time.

The random search technique resembles the complete enumeration technique except that one selects a set of inputs at random. The simulated results based on the set that yields the maximum (minimum) value of the response function is taken to be the optimal point. This procedure reduces the number of simulation runs required to yield an ‘optimal’ result; however, there is no guarantee that the point found is actually the optimal point. Of course, the more points selected, the more likely the analyst is to achieve the true optimum. Note that the requirement that each factor assumes only a finite number of values is not a requirement in this scheme. Replications can be made on the treatment combinations selected, to increase the confidence in the optimal point. Which strategy is better, replicating a few points or looking at a single observation on more points, depends on the problem.

A.3.3.1.3 Response Surface Search

Response surface search attempts to fit a polynomial to J(v). If the design space v is suitably small, the performance function J(v) may be approximated by a response surface, typically a first order, or perhaps quadratic order in v, possibly after transformation, e.g. log (v). The response surface method (RSM) requires running the simulation in a first order experimental design to determine the path of steepest descent. Simulation runs made along this path continue, until one notes no improvement in J(v). The analyst then runs a new first order experimental design around the new ‘optimal’ point reached, and finds a new path of steepest descent. The process continues, until there is a lack of fit in the fitted first order surface. Then, one runs a second order design, and takes the optimum of the fittest second order surface as the estimated optimum.

Although it is desirable for search procedures to be efficient over a wide range of response surfaces, no current procedure can effectively overcome non-unimodality (surfaces having more than one local maximum or minimum). An obvious way to find the global optimal would be to evaluate all the local optima. One technique that is used when non-unimodality is known to exist, is called the “Las Vegas” technique. This search procedure estimates the distribution of the local optima by plotting the estimated J(v) for each local search against its corresponding search number. Those local searches that produce a response greater than any previous response are then identified and a curve is fitted to the data. This curve is then used to project the “estimated incremental” response that will be achieved by one more search.

The search continues until the value of the estimated improvement in the search is less than the cost of completing one additional search.

It should be noted that a well-designed experiment requires a sufficient number of replications so that the average response can be treated as a deterministic number for search comparisons. Otherwise, since replications are expensive, it becomes necessary to effectively utilize the number of simulation runs. Although each simulation is at a different setting of the controllable variables, one can use smoothing techniques such as exponential smoothing to reduce the required number of replications.

A.3.3.2 Pattern Search Techniques

Pattern search techniques assume that any successful set of moves used in searching for an approximated optimum is worth repeating. These techniques start with small steps; then, if these are successful, the step size increases. Alternatively, when a sequence of steps fails to improve the objective function, this indicates that shorter steps are appropriate so we may not overlook any promising direction. These techniques start by initially selecting a set of incremental values for each factor. Starting at an initial base point, they check if any incremental changes in the first variable yield an improvement. The resulting improved setting becomes the new intermediate base point. One repeats the process for each of the inputs until one obtains a new setting where the intermediate base points act as the initial base point for the first variable. The technique then moves to the new setting. This procedure is repeated, until further changes cannot be made with the given incremental values. Then, the incremental values are decreased, and the procedure is repeated from the beginning. When the incremental values reach a pre-specified tolerance, the procedure terminates; the most recent factor settings are reported as the solution.

A.3.3.2.1 Conjugate Direction Search

The conjugate direction search requires no derivative estimation, yet it finds the optimum of an N-dimensional quadratic surface after, at most, N-iterations, where the number of iterations is equal to the dimension of the quadratic surface. The procedure redefines the n dimensions so that a single variable search can be used successively. Single variable procedures can be used whenever dimensions can be treated independently. The optimization along each dimension leads to the optimization of the entire surface.

Two directions are defined to be conjugate whenever the cross-product terms are all zero. The conjugate direction technique tries to find a set of n dimensions that describes the surface such that each direction is conjugate to all others.

Using the above result, the technique attempts to find two search optima and replace the n^{th} dimension of the quadratic surface by the direction specified by the two optimal points. Successively replacing the original dimension yields a new set of n dimensions in which, if the original surface is quadratic, all directions are conjugate to each other and appropriate for n single variable searches. While this search procedure appears to be very simple, we should point out that the selection of appropriate step sizes is most critical. The step size selection is more critical for this search technique because – during axis rotation – the step size does not remain invariant in all dimensions. As the rotation takes place, the best step size changes, and becomes difficult to estimate.

A.3.3.2.2 Steepest Ascent (Descent)

The steepest ascent (descent) technique uses a fundamental result from calculus (that the gradient points in the direction of the maximum increase of a function), to determine how the initial settings of the parameters should be changed to yield an optimal value of the response variable. The direction of movement is made proportional to the estimated sensitivity of the performance of each variable.

Although quadratic functions are sometimes used, one assumes that performance is linearly related to the change in the controllable variables for small changes. Assume that a good approximation is a linear form. The basis of the linear steepest ascent is that each controllable variable is changed in proportion to the magnitude of its slope. When each controllable variable is changed by a small amount, it is analogous to determining the gradient at a point. For a surface containing N controllable variables, this requires N points around the point of interest. When the problem is not an n-dimensional elliptical surface, the parallel-tangent points are extracted from bitangents and inflection points of occluding contours. Parallel tangent points are points on the occluding contour where the tangent is parallel to a given bitangent or the tangent at an inflection point.

A.3.3.2.3 Tabu Search Technique

An effective technique to overcome local optimality for discrete optimization is the Tabu Search technique. It explores the search space by moving from a solution to its best neighbor, even if this results in a deterioration of the performance measure value. This approach increases the likelihood of moving out of local optima. To avoid cycling, solutions that were recently examined are declared tabu (Taboo) for a certain number of iterations. Applying intensification procedures can accentuate the search in a promising region of the solution space. In contrast, diversification can be used to broaden the search to a less explored region. Much remains to be discovered about the range of problems for which the tabu search is best suited.

A.3.3.2.4 Hooke and Jeeves Type Techniques

The Hooke and Jeeves pattern search uses two kinds of moves; namely, an exploratory and a pattern move. The exploratory move is accomplished by doing a coordinate search in one pass through all the variables. This gives a new “base point” from which a pattern move is made. A pattern move is a jump in the pattern direction determined by subtracting the current base point from the previous base point. After the pattern move, another exploratory move is carried out at the point reached. If the estimate of J(v) is improved at the final point after the second exploratory move, it becomes the new base point. If it fails to show improvement, an exploratory move is carried out at the last base point with a smaller step in the coordinate search. The process stops when the step gets “small” enough.

A.3.3.2.5 Simplex-Based Techniques

The simplex-based technique performs simulation runs first at the vertices of the initial simplex; i.e. a polyhedron in the v-space having N+1 vertices. A subsequent simplex (moving towards the optimum) are formed by three operations performed on the current simplex: reflection, contraction, and expansion. At each stage of the search process, the point with the highest J(v) is replaced with a new point found via reflection through the centroid of the simplex. Depending on the value of J(v) at this new point, the simplex is either expanded, contracted, or unchanged. The simplex technique starts with a set of N+1 factor settings. These N+1 points are all the same distance from the current point. Moreover, the distance between any two points of these N+1 points is the same. Then, by comparing their response values, the technique eliminates the factor setting with the worst functional value and replaces it with a new factor setting, determined by the centroid of the N remaining factor settings and the eliminated factor setting. The resulting simplex either grows or shrinks, depending on the response value at the new factor settings. One repeats the procedure until no more improvement can be made by eliminating a point, and the resulting final simplex is small. While this technique will generally perform well for unconstrained problems, it may collapse to a point on a boundary of a feasible region, thereby causing the search to come to a premature halt. This technique is effective if the response surface is generally bowl-shaped even with some local optimal points.

A.3.3.3 Probabilistic Search Techniques

All probabilistic search techniques select trial points governed by a scan distribution, which is the main source of randomness. These search techniques include random search, pure adaptive techniques, simulated annealing, and genetic methods.

A.3.3.3.1 Random Search

A simple, but very popular approach is the random search, which centers a symmetric probability density function (pdf) [e.g. the normal distribution], about the current best location. The standard normal N(0, 1) is a popular choice, although the uniform distribution U[-1, 1] is also common.

A variation of the random search technique determines the maximum of the objective function by analyzing the distribution of J(v) in the bounded sub-region. In this variation, the random data are fitted to an asymptotic extreme-value distribution, and J* is estimated with a confidence statement. Unfortunately, these techniques cannot determine the location of J*, which can be as important as the J value itself. Some techniques calculate the mean value and the standard deviation of J(v) from the random data as they are collected. Assuming that J is distributed normally in the feasible region., the first trial, that yields a J-value two standard deviations within the mean value, is taken as a near-optimum solution.

A.3.3.3.2 Pure Adaptive Search

Various pure adaptive search techniques have been suggested for optimization in simulation. Essentially, these techniques move from the current solution to the next solution that is sampled uniformly from the set of all better feasible solutions.

A.3.3.4 Evolutionary Techniques

Nature is a robust optimizer. By analyzing nature's optimization mechanism we may find acceptable solution techniques to intractable problems. Two concepts that have most promise are simulated annealing and the genetic techniques.

A.3.3.4.1 Simulated Annealing

Simulated annealing (SA) borrows its basic ideas from statistical mechanics. A metal cools, and the electrons align themselves in an optimal pattern for the transfer of energy. In general, a slowly cooling system, left to itself, eventually finds the arrangement of atoms, which has the lowest energy. The is the behavior, which motivates the method of optimization by SA. In SA we construct a model of a system and slowly decrease the "temperature" of this theoretical system, until the system assumes a minimal energy structure. The problem is how to map our particular problem to such an optimizing scheme.

SA as an optimization technique was first introduced to solve problems in discrete optimization, mainly combinatorial optimization. Subsequently, this technique has been successfully applied to solve optimization problems over the space of continuous decision variables. SA is a simulation optimization technique that allows random ascent moves in order to escape the local minima, but a price is paid in terms of a large increase in the computational time required. It can be proven that the technique will find an approximated optimum. The annealing schedule might require a long time to reach a true optimum.

A.3.3.4.2 Genetic Techniques

Genetic techniques (GT) are optimizers that use the ideas of evolution to optimize a system that is too difficult for traditional optimization techniques. Organisms are known to optimize themselves to adapt to their environment.

GT differ from traditional optimization procedures in that GT work with a coding of the decision parameter set, not the parameters themselves; GT search a population of points, not a single point; GT use objective function information, not derivatives or other auxiliary knowledge; and finally, GT use probabilistic transition rules, not deterministic rules. GT are probabilistic search optimizing techniques that do not require mathematical knowledge of the response surface of the system, which they are optimizing. They borrow the paradigms of genetic evolution, specifically selection, crossover, and mutation.

Selection: The current points in the space are ranked in terms of their fitness by their respective response values. A probability is assigned to each point that is proportional to its fitness, and parents (a mating pair) are randomly selected.

Crossover: The new point, or offspring, is chosen, based on some combination of the genetics of the two parents.

Mutation: The location of offspring is also susceptible to mutation, a process, which occurs with probability p, by which a offspring is replaced randomly by a new offspring location.

A generalized GT generates p new offspring at once and kills off all of the parents. This modification is important in the simulation environment. GT are well suited for qualitative or policy decision optimization such as selecting the best queuing disciplines or network topologies. They can be used to help determine the design of the system and its operation. For applications of GT to inventory systems, job-shop, and computer time-sharing problems. GT do not have certain shortcomings of other optimization techniques, and they will usually result in better calculated optima than those found with the traditionally techniques. They can search a response surface with many local optima and find (with a high probability) the approximate global optimum. One may use GT to find an area of potential interest, and then resort to other techniques to find the optimum. Recently, several classical GT principles have been challenged.

Differential Evolution: Differential Evolution (DE) is a genetic type of algorithm for solving continuous stochastic function optimization. The basic idea is to use vector differences for perturbing the vector population. DE adds the weighted difference between two population vectors to a third vector. This way, no separate probability distribution has to be used, which makes the scheme completely self-organizing.

A.3.3.4.3 A Short Comparison

When performing search techniques in general, and simulated annealing or genetic techniques specifically, the question of how to generate the initial solution arises. Should it be based on a heuristic rule or on a randomly generated one? Theoretically, it should not matter, but in practice this may depend on the problem. In some cases, a pure random solution systematically produces better final results. On the other hand, a good initial solution may lead to lower overall run times. This can be important, for example, in cases where each iteration takes a relatively long time; therefore, one has to use some clever termination rule. Simulation time is a crucial bottleneck in an optimization process. In many cases, a simulation is run several times with different initial solutions. Such a technique is most robust, but it requires the maximum number of replications compared with all other techniques. The pattern search technique applied to small problems with no constraints or qualitative input parameters requires fewer replications than the GT. GT, however, can easily handle constraints, and have lower computational complexity. Finally, simulated annealing can be embedded within the Tabu search to construct a probabilistic technique for global optimization.

A.3.3.5 Stochastic Approximation Techniques

Two related stochastic approximation techniques have been proposed, one by Robbins and Monro and one by Kiefer and Wolfowitz. The first technique was not useful for optimization until an unbiased estimator for the gradient was found. Kiefer and Wolfowitz developed a procedure for optimization using finite differences. Both techniques are useful in the optimization of noisy functions, but they did not receive much attention in the simulation field until recently. Generalization and refinement of stochastic approximation procedures give rise to a weighted average, and stochastic quasi-gradient methods. These deal with constraints, non-differentiable functions, and some classes of non-convex functions, among other things.

A.3.3.5.1 Kiefer-Wolfowitz Type Techniques

Kiefer and Wolfowitz proposed a finite difference approximation to the derivative. One version of the Kiefer-Wolfwitz technique uses two-sided finite differences. The first fact to notice about the K-W estimate is that it requires 2N simulation runs, where N is the dimension of vector parameter v. If the decision maker is interested in gradient estimation with respect to each of the components of v, then 2N simulations must be run for each component of v . This is inefficient. The second fact is that it may have a very poor variance, and it may result in numerical calculation difficulties.

A.3.3.5.2 Robbins-Monro Type Techniques

The original Robbins-Monro (R-M) technique is not an optimization scheme, but rather a root finding procedure for functions whose exact values are not known but are observed with noise. Its application to optimization is immediate: use the procedure to find the root of the gradient of the objective function.

Interest was renewed in the R-M technique as a means of optimization, with the development of the perturbation analysis, score function (known also as likelihood ratio method), and frequency domain estimates of derivatives. Optimization for simulated systems based on the R-M technique is known as a “single-run” technique. These procedures optimize a simulation model in a single run simulation with a run length comparable to that required for a single iteration step in the other methods. This is achieved essentially be observing the sample values of the objective function and, based on these observations, updating the values of the controllable parameters while the simulation is running, that is, without restarting the simulation. This observing-updating sequence is done repeatedly, leading to an estimate of the optimum at the end of a single-run simulation. Besides having the potential of large computational savings, this technique can be a powerful tool in real-time optimization and control, where observations are taken as the system is evolving in time.

A.3.3.6 Gradient Surface Method

One may combine the gradient-based techniques with the response surface methods (RSM) for optimization purposes. One constructs a response surface with the aid of n response points and the components of their gradients.

The gradient surface method (GSM) combines the virtue of RSM with that of the single-run, gradient estimation techniques such as Perturbation Analysis, and Score Function techniques. A single simulation experiment with little extra work yields N + 1 pieces of information; i.e. one response point and N components of the gradient. This is in contrast to crude simulation, where only one piece of information, the response value, is obtained per experiment. Thus by taking advantage of the computational efficiency of single-run gradient estimators. In general, N-fold fewer experiments will be needed to fit a global surface compared to the RSM. At each step, instead of using Robbins-Monro techniques to locate the next point locally, we determine a candidate for the next point globally, based on the current global fit to the performance surface.

The GSM approach has the following advantages; The technique can quickly get to the vicinity of the optimal solution because its orientation is global [23, 39]. Thus, it produces satisfying solutions quickly;

Like RSM, it uses all accumulated information; And, in addition, it uses gradient surface fitting, rather than direct performance response-surface fitting via single-run gradient estimators. This significantly reduces the computational efforts compared with RSM. Similar to RSM, GSM is less sensitive to estimation error and local optimality; and, finally, it is an on-line technique, the technique may be implemented while the system is running.

A typical optimization scheme involves two phases: a Search Phase and an Iteration Phase. Most results in analytic computational complexity assume that good initial approximations are available, and deal with the iteration phase only. If enough time is spent in the initial search phase, we can reduce the time needed in the iteration phase. The literature contains papers giving conditions for the convergence of a process; a process has to be more than convergent in order to be computationally interesting. It is essential that we be able to limit the cost of computation. In this sense, GSM can be thought of as helping the search phase and as an aid to limit the cost of computation. One can adopt standard or simple devices for issues such as stopping rules.

For on-line optimization, one may use a new design in GSM called 'single direction' design. Since for on-line optimization it may not be advisable or feasible to disturb the system, random design usually is not suitable.

A.3.3.7 Post-Solution Analysis

Stochastic models typically depend upon various uncertain and uncontrollable input parameters that must be estimated from existing data sets. We focus on the statistical question of how input-parameter uncertainty propagates through the model into output- parameter uncertainty. The sequential stages are descriptive, prescriptive and post-prescriptive analysis.

A.3.3.8 Rare Event Simulation

Large deviations can be used to estimate the probability of rare events, such as buffer overflow, in queuing networks. It is simple enough to be applied to very general traffic models, and sophisticated enough to give insight into complex behavior.

Simulation has numerous advantages over other approaches to performance and dependability evaluation; most notably, its modelling power and flexibility. For some models, however, a potential problem is the excessive simulation effort (time) required to achieve the desired accuracy. In particular, simulation of models involving rare events, such as those used for the evaluation of communications and highly-dependable systems, is often not feasible using standard techniques. In recent years, there have been significant theoretical and practical advances towards the development of efficient simulation techniques for the evaluation of these systems.

Methodologies include: Techniques based on importance sampling, The "restart" method, and Hybrid analytic/simulation techniques among newly devised approaches.

A.3.3.9 Conclusion

With the growing incidence of computer modeling and simulation, the scope of simulation domain must be extended to include much more than traditional optimization techniques. Optimization techniques for simulation must also account specifically for the randomness inherent in estimating the performance measure and satisfying the constraints of stochastic systems. We described the most widely used optimization techniques that can be effectively integrated with a simulation model. We also described techniques for post-solution analysis with the aim of theoretical unification of the existing techniques. All techniques were presented in step-by-step format to facilitate implementation in a variety of operating systems and computers, thus improving portability.

General comparisons among different techniques in terms of bias, variance, and computational complexity are not possible. However, a few studies rely on real computer simulations to compare different techniques in terms of accuracy and number of iterations. Total computational effort for reduction in both the bias and variance of the estimate depends on the computational budget allocated for a simulation optimization. No single technique works effectively and/or efficiently in all cases.

The simplest technique is the random selection of some points in the search region for estimating the performance measure. In this technique, one usually fixes the number of simulation runs and takes the smallest (or largest) estimated performance measure as the optimum. This technique is useful in combination with other techniques to create a multi-start technique for global optimization. The most effective technique to overcome local optimality for discrete optimization is the Tabu Search technique. In general, the probabilistic search techniques, as a class, offer several advantages over other optimization techniques based on gradients. In the random search technique, the objective function can be non-smooth or even have discontinuities. The search program is simple to implement on a computer, and it often shows good convergence characteristics in noisy environments. More importantly, it can offer the global solution in a multi-modal problem, if the technique is employed in the global sense. Convergence proofs under various conditions are given in.

The Hooke-Jeeves search technique works well for unconstrained problems with less than 20 variables; pattern search techniques are more effective for constrained problems. Genetic techniques are most robust and can produce near-best solutions for larger problems. The pattern search technique is most suitable for small size problems with no constraint, and it requires fewer iterations than the genetic techniques. The most promising techniques are the stochastic approximation, simultaneous perturbation, and the gradient surface methods. Stochastic approximation techniques using perturbation analysis, score function, or simultaneous perturbation gradient estimators, optimize a simulation model in a single simulation run. They do so by observing the sample values of the objective function, and based on these observations, the stochastic approximation techniques update the values of the controllable parameters while the simulation is running and without restarting the simulation. This observing-updating sequence, done repeatedly, leads to an estimate of the optimum at the end of a single-run simulation. Besides having the potential of large savings in computational effort in the simulation environment, this technique can be a powerful tool in real-time optimization and control, where observations are taken as the system is evolving over time.

Response surface methods have a slow convergence rate, which makes them expensive. The gradient surface method combines the advantages of the response surface methods (RSM) and efficiency of the gradient estimation techniques, such as infinitesimal perturbation analysis, score function, simultaneous perturbation analysis, and frequency domain technique. In the gradient surface method (GSM) the gradient is estimated, and the performance gradient surface is estimated from observations at various points, similar to the RSM. Zero points of the successively approximating gradient surface are then taken as the estimates of the optimal solution. GSM is characterized by several attractive features: it is a single run technique and more efficient than RSM; at each iteration step, it uses the information from all of the data points rather than just the local gradient; it tries to capture the global features of the gradient surface and thereby quickly arrive in the vicinity of the optimal solution, but close to the optimum, they take many iterations to converge to stationary points. Search techniques are therefore more suitable as a second phase. The main interest is to figure out how to allocate the total available computational budget across the successive iterations.

For when the decision variable is qualitative, such as finding the best system configuration, a random or permutation test is proposed. This technique starts with the selection of an appropriate test statistic, such as the absolute difference between the mean responses under two scenarios. The test value is computed for the original data set. The data are shuffled (using a different seed); the test statistic is computed for the shuffled data; and the value is compared to the value of the test statistic for the original, un-shuffled data.

If the statistics for the shuffled data are greater than or equal to the actual statistic for the original data, then a counter c, is incremented by 1. The process is repeated for any desired m number of times. The final step is to compute (c+1)/(m+1), which is the significant level of the test. The null hypothesis is rejected if this significance level is less than or equal to the specified rejection level for the test. There are several important aspects to this non-parametric test. First, it enables the user to select the statistic. Second, assumptions such as normality or equality of variances made for the t-test, ranking-and-selection, and multiple-comparison procedures, are no longer needed. A generalization is the well-known bootstrap technique.

- Computational studies of techniques for systems with a large number of controllable parameters and constraints.
- Effective combinations of several efficient techniques to achieve the best results under constraints on computational resources.
- Development of parallel and distributed schemes and an expert system that incorporates all available techniques.

A.3.4 Simulation Products

A.3.4.1 Crystal Ball 2000 Professional Edition

Crystal Ball Pro is a suite of products that includes Crystal Ball 2000 Standard Edition (Monte Carlo simulation software for spreadsheet risk analysis), OptQuest (global optimization software that finds the best solutions for spreadsheets under conditions of uncertainty), CB Predictor (time-series forecasting software for historic data), Crystal Ball and CB Predictor Developer Kits (automate and customize your Crystal Ball simulations) and Crystal Ball Tools (improve your ability to analyze variables and save on the time spent building models).

OS: Windows 95, Windows 98, Windows NT; Microsoft Office 95 (or later)

Vendor: Decisioneering Inc. http://www.decisioneering.com

A.3.4.2 DecisionPro 3.0

DecisionPro picks up where your spreadsheet leaves off. It helps you make the best possible business decisions by applying proven management techniques such as decision tree analysis, Monte Carlo simulation, linear programming, advanced forecasting methods, and others. By integrating all of the key quantitative methods in management into a single application, DecisionPro gives you unprecedented power and flexibility. What's more, Decision Pro's innovative interface makes this power so easy to apply that you will find yourself using it even for routine problems.

OS: Windows 95, Windows 98, Windows NT

Vendor: Vanguard Software Corporation http://www.vanguardsw.com

Demo: http://www.vanguardsw.com/models/download.dsb?1

A.3.4.3 Deneb/QUEST

The Queuing Event Simulation Tool, Deneb/QUEST represents a quantum leap in the modeling and analysis of manufacturing systems. Detailed physical system properties combined with interactive 3D graphics and visual analysis deliver a new level of ease, power, and accuracy. Deneb/QUEST gives new meaning to ``interactive modeling'' and ``what-if' evaluations. The effects of any model change can be instantly seen just by pressing the RUN button. True 3D animation occurs in real-time, not as a recording. Lengthy edit/compile/run/analyze cycles are eliminated. Now, efficiently explore production scenarios, product mixes, and failure responses for machine and labor utilization, bottleneck and throughput evaluation. Analyze and justify production costs, capital investments, and monitor the value of work-in-process, inventory and ROI.

OS: Windows 95, Windows 98, Windows NT

Vendor: Deneb Robotnics http://www.deneb.com

A.4 DATA ANALYSIS AND MINING

A.4.1 Data Mining and Analysis Presentation

A.4.1.1 Data Mining Overview

Data mining, the extraction of hidden predictive information from large databases, is a powerful new technology with great potential to help companies focus on the most important information in their data warehouses. Data mining tools predict future trends and behaviors, allowing businesses to make proactive, knowledge-driven decisions. The automated, prospective analyses offered by data mining move beyond the analyses of past events provided by retrospective tools typical of decision support systems. Data mining tools can answer business questions that traditionally were too time consuming to resolve. They scour databases for hidden patterns, finding predictive information that experts may miss because it lies outside their expectations.

Most companies already collect and refine massive quantities of data. Data mining techniques can be implemented rapidly on existing software and hardware platforms to enhance the value of existing information resources, and can be integrated with new products and systems as they are brought on-line. When implemented on high performance client/server or parallel processing computers, data mining tools can analyze massive databases to deliver answers to questions such as, "Which clients are most likely to respond to my next promotional mailing, and why?"

This paper provides an introduction to the basic technologies of data mining. Examples of profitable applications illustrate its relevance to today's business environment as well as a basic description of how data warehouse architectures can evolve to deliver the value of data mining to end users.

A.4.1.2 Foundations of Data Mining

Data mining techniques are the result of a long process of research and product development. This evolution began when business data was first stored on computers, continued with improvements in data access, and more recently, generated technologies that allow users to navigate through their data in real time. Data mining takes this evolutionary process beyond retrospective data access and navigation to prospective and proactive information delivery. Data mining is ready for application in the business community because it is supported by three technologies that are now sufficiently mature: Massive data collection, Powerful multiprocessor computers, Data mining algorithms.

Data mining algorithms embody techniques that have existed for at least 10 years, but have only recently been implemented as mature, reliable, understandable tools that consistently outperform older statistical methods.

In the evolution from business data to business information, each new step has built upon the previous one. For example, dynamic data access is critical for drill-through in data navigation applications, and the ability to store large databases is critical to data mining. From the user's point of view, the four steps listed in the table were revolutionary because they allowed new business questions to be answered accurately and quickly.

Table A.3: Steps in the Evolution of Data Mining

Evolutionary Step	Business Question	Enabling Technologies	Product Providers	Characteristics
Data Collection (1960s)	"What was my total revenue in the last five years?"	Computers, tapes, disks	IBM, CDC	Retrospective, static data delivery
Data Access (1980s)	"What were unit sales in New England last March?"	Relational databases (RDBMS), Structured Query Language (SQL), ODBC	Oracle, Sybase, Informix, IBM, Microsoft	Retrospective, dynamic data delivery at record level
Data Warehousing and Decision Support (1990s)	"What were unit sales in New England last March? Drill down to Boston."	On-line analytic processing (OLAP), multidimensional databases, data warehouses	Pilot, Comshare, Arbor, Cognos, Microstrategy	Retrospective, dynamic data delivery at multiple levels
Data Mining (Emerging Today)	"What's likely to happen to Boston unit sales next month? Why?"	Advanced algorithms, multiprocessor computers, massive databases	Pilot, Lockheed, IBM, SGI, numerous startups (nascent industry)	Prospective, proactive information delivery

The core components of data mining technology have been under development for decades, in research areas such as statistics, artificial intelligence, and machine learning. Today, the maturity of these techniques, coupled with high-performance relational database engines and broad data integration efforts, make these technologies practical for current data warehouse environments.

A.4.1.3 The Scope of Data Mining

Data mining derives its name from the similarities between searching for valuable business information in a large database – for example, finding linked products in gigabytes of store scanner data – and mining a mountain for a vein of valuable ore. Both processes require either sifting through an immense amount of material, or intelligently probing it to find exactly where the value resides. Given databases of sufficient size and quality, data mining technology can generate new business opportunities by providing these capabilities:

- Automated prediction of trends and behaviors. Data mining automates the process of finding predictive information in large databases. Questions that traditionally required extensive hands-on analysis can now be answered directly from the data – quickly. A typical example of a predictive problem is targeted marketing. Data mining uses data on past promotional mailings to identify the targets most likely to maximize return on investment in future mailings. Other predictive problems include forecasting bankruptcy and other forms of default, and identifying segments of a population likely to respond similarly to given events.
- Automated discovery of previously unknown patterns. Data mining tools sweep through databases and identify previously hidden patterns in one step. An example of pattern discovery is the

analysis of retail sales data to identify seemingly unrelated products that are often purchased together. Other pattern discovery problems include detecting fraudulent credit card transactions and identifying anomalous data that could represent data entry keying errors.

- Data mining techniques can yield the benefits of automation on existing software and hardware platforms, and can be implemented on new systems as existing platforms are upgraded and new products developed. When data mining tools are implemented on high performance parallel processing systems, they can analyze massive databases in minutes. Faster processing means that users can automatically experiment with more models to understand complex data. High speed makes it practical for users to analyze huge quantities of data. Larger databases, in turn, yield improved predictions.

Databases can be larger in both depth and breadth:

- More columns. Analysts must often limit the number of variables they examine when doing hands-on analysis due to time constraints. Yet variables that are discarded because they seem unimportant may carry information about unknown patterns. High performance data mining allows users to explore the full depth of a database, without pre-selecting a subset of variables.
- More rows. Larger samples yield lower estimation errors and variance, and allow users to make inferences about small but important segments of a population.

The most commonly used techniques in data mining are:

- Artificial neural networks: Non-linear predictive models that learn through training and resemble biological neural networks in structure.
- Decision trees: Tree-shaped structures that represent sets of decisions. These decisions generate rules for the classification of a dataset. Specific decision tree methods include Classification and Regression Trees (CART) and Chi Square Automatic Interaction Detection (CHAID).
- Genetic algorithms: Optimization techniques that use processes such as genetic combination, mutation, and natural selection in a design based on the concepts of evolution.
- Nearest neighbor method: A technique that classifies each record in a dataset based on a combination of the classes of the k record(s) most similar to it in a historical dataset (where k is greater than or equal to 1). Sometimes called the k-nearest neighbor technique.
- Rule induction: The extraction of useful if-then rules from data based on statistical significance.

Many of these technologies have been in use for more than a decade in specialized analysis tools that work with relatively small volumes of data. These capabilities are now evolving to integrate directly with industry-standard data warehouse and OLAP platforms.

A.4.1.4 An Architecture for Data Mining

To best apply these advanced techniques, they must be fully integrated with a data warehouse as well as flexible interactive business analysis tools. Many data mining tools currently operate outside of the warehouse, requiring extra steps for extracting, importing, and analyzing the data. Furthermore, when new insights require operational implementation, integration with the warehouse simplifies the application of results from data mining. The resulting analytic data warehouse can be applied to improve business processes throughout the organization, in areas such as promotional campaign management, fraud detection, new product rollout, and so on. This figure illustrates an architecture for advanced analysis in a large data warehouse.

A.4.1.4.1 Integrated Data Mining Architecture

The ideal starting point is a data warehouse containing a combination of internal data tracking all customer contact coupled with external market data about competitor activity. Background information on potential customers also provides an excellent basis for prospecting. This warehouse can be implemented in a variety of relational database systems: Sybase, Oracle, Redbrick, and so on, and should be optimized for flexible and fast data access.

An OLAP (On-Line Analytical Processing) server enables a more sophisticated end-user business model to be applied when navigating the data warehouse. The multidimensional structures allow the user to analyze the data as they want to view their business – summarizing by product line, region, and other key perspectives of their business. The Data Mining Server must be integrated with the data warehouse and the OLAP server to embed ROI-focused business analysis directly into this infrastructure. An advanced, process-centric metadata template defines the data mining objectives for specific business issues like campaign management, prospecting, and promotion optimization. Integration with the data warehouse enables operational decisions to be directly implemented and tracked. As the warehouse grows with new decisions and results, the organization can continually mine the best practices and apply them to future decisions.

This design represents a fundamental shift from conventional decision support systems. Rather than simply delivering data to the end user through query and reporting software, the Advanced Analysis Server applies users' business models directly to the warehouse and returns a proactive analysis of the most relevant information. These results enhance the metadata in the OLAP Server by providing a dynamic metadata layer that represents a distilled view of the data. Reporting, visualization, and other analysis tools can then be applied to plan future actions and confirm the impact of those plans.

A.4.1.5 Data Mining Applications

A wide range of companies have deployed successful applications of data mining. While early adopters of this technology have tended to be in information-intensive industries such as financial services and direct mail marketing, the technology is applicable to any company looking to leverage a large data warehouse to better manage their customer relationships. Two critical factors for success with data mining are: a large, well-integrated data warehouse and a well-defined understanding of the business process within which data mining is to be applied (such as customer prospecting, retention, campaign management, and so on).

Some successful application areas include:

- A pharmaceutical company can analyze its recent sales force activity and their results to improve targeting of high-value physicians and determine which marketing activities will have the greatest impact in the next few months. The data needs to include competitor market activity as well as information about the local health care systems. The results can be distributed to the sales force via a wide-area network that enables the representatives to review the recommendations from the perspective of the key attributes in the decision process. The ongoing, dynamic analysis of the data warehouse allows best practices from throughout the organization to be applied in specific sales situations.
- A credit card company can leverage its vast warehouse of customer transaction data to identify customers most likely to be interested in a new credit product. Using a small test mailing, the attributes of customers with an affinity for the product can be identified. Recent projects have indicated more than a 20-fold decrease in costs for targeted mailing campaigns over conventional approaches.
- A diversified transportation company with a large direct sales force can apply data mining to identify the best prospects for its services. Using data mining to analyze its own customer

experience, this company can build a unique segmentation identifying the attributes of high-value prospects. Applying this segmentation to a general business database such as those provided by Dun & Bradstreet can yield a prioritized list of prospects by region.

- A large consumer package goods company can apply data mining to improve its sales process to retailers. Data from consumer panels, shipments, and competitor activity can be applied to understand the reasons for brand and store switching. Through this analysis, the manufacturer can select promotional strategies that best reach their target customer segments.

Each of these examples have a clear common ground. They leverage the knowledge about customers implicit in a data warehouse to reduce costs and improve the value of customer relationships. These organizations can now focus their efforts on the most important (profitable) customers and prospects, and design targeted marketing strategies to best reach them.

A.4.1.6 Conclusion

Comprehensive data warehouses that integrate operational data with customer, supplier, and market information have resulted in an explosion of information. Competition requires timely and sophisticated analysis on an integrated view of the data. However, there is a growing gap between more powerful storage and retrieval systems and the users' ability to effectively analyze and act on the information they contain. Both relational and OLAP technologies have tremendous capabilities for navigating massive data warehouses, but brute force navigation of data is not enough. A new technological leap is needed to structure and prioritize information for specific end-user problems. The data mining tools can make this leap. Quantifiable business benefits have been proven through the integration of data mining with current information systems, and new products are on the horizon that will bring this integration to an even wider audience of users.

A.4.2 Data Mining Products

The Product Features table includes many of the mainstream data mining products, focusing on those with which Exclusive Ore has had hands-on experience. While we attempt to keep the table as current as possible, we encourage you to use the hot-links to the vendor sites for the latest information on features and functionality.

Key: NN = Neural Net; Tree = Decision Tree; Naïve Bayes; k-Mns = K Means; k-NN = k-Nearest Neighbor; Stats = Traditional Statistical Techniques; Pred = Prediction; Time Series; Clust = Clustering; Assoc = Association; Win 32 = Windows 95/98/NT; UNIX; Par = Parallel Scalability (in at least one OS); API SDK = API or SDK available; SQL Ext = Uses Special SQL Extensions

Table A.4: Data Mining Products

Company	Product	NN	Tree	Naïve Bayes	k-Mns	k-NN	Stats	Pred	Time Series	Clust	Assoc	Win 32	UNIX	Par	API SDK	SQL Ext
Angoss International Ltd.	KnowledgeSEEKER		Yes					Yes				Yes	Yes	Yes		
	KnowledgeSTUDIO	Yes	Yes		Yes			Yes		Yes		Yes	Yes	Yes	Yes	Yes
Business Objects	BusinessMiner		Yes									Yes				
Cognos Incorporated	4Thought	Yes						Yes	Yes			Yes				
Informix Software Inc.	Red Brick Data Mine			Yes				Yes				Yes	Yes			Yes
International Business Machines	Intelligent Miner	Yes	Yes			Yes		Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	
Accrue Software	Decision Series	Yes	Yes	Yes				Yes		Yes	Yes		Yes	Yes	Yes	
NeuralWare	NeuralSIM	Yes						Yes				Yes				
Oracle Corp.	Darwin	Yes	Yes			Yes		Yes					Yes	Yes		
RightPoint Software	DataCruncher			Yes				Yes				Yes	Yes			
Salford Systems	CART		Yes					Yes				Yes	Yes			
SAS Institute	Enterprise Miner	Yes	Yes				Yes	Yes	Yes	Yes	Yes	Yes	Yes			
SGI	MineSet		Yes	Yes	Yes			Yes		Yes	Yes	Yes	Yes	Yes	Yes	
SPSS, Inc.	Answer Tree		Yes				Yes	Yes				Yes	Yes			
	Clementine	Yes	Yes					Yes	Yes	Yes	Yes	Yes	Yes			
	Neural Connection	Yes					Yes	Yes				Yes	Yes			
Tandem Computers Incorporated	Object-Relational Technology												Yes	Yes		Yes
Unica Technology	Pattern Recognition Workbench	Yes			Yes	Yes	Yes	Yes	Yes	Yes		Yes			Yes	
	Model 1	Yes	Yes	Yes	Yes		Yes	Yes				Yes	Yes	Yes		

A.5 ARTIFICIAL INTELLIGENCE

A.5.1 Artificial Intelligence Presentation

A.5.1.1 What is Artificial Intelligence (AI)?

Artificial intelligence can be viewed from a variety of perspectives. From the perspective of **intelligence** *artificial intelligence* is making machines "intelligent" – acting as we would expect people to act. The inability to distinguish computer responses from human responses is called the Turing test. Intelligence requires knowledge, and expert problem solving restricts domain to allow including significant relevant knowledge. From a **research** perspective *artificial intelligence* is the study of how to make computers do things which, at the moment, people do better.

AI began in the early 1960s, the first attempts were game playing (checkers), theorem proving (a few simple theorems) and general problem solving (only very simple tasks). General problem solving was much more difficult than originally anticipated. Researchers were unable to tackle problems routinely handled by human experts. The name "artificial intelligence" came from the roots of the area of study. AI researchers are active in a variety of **domains**.

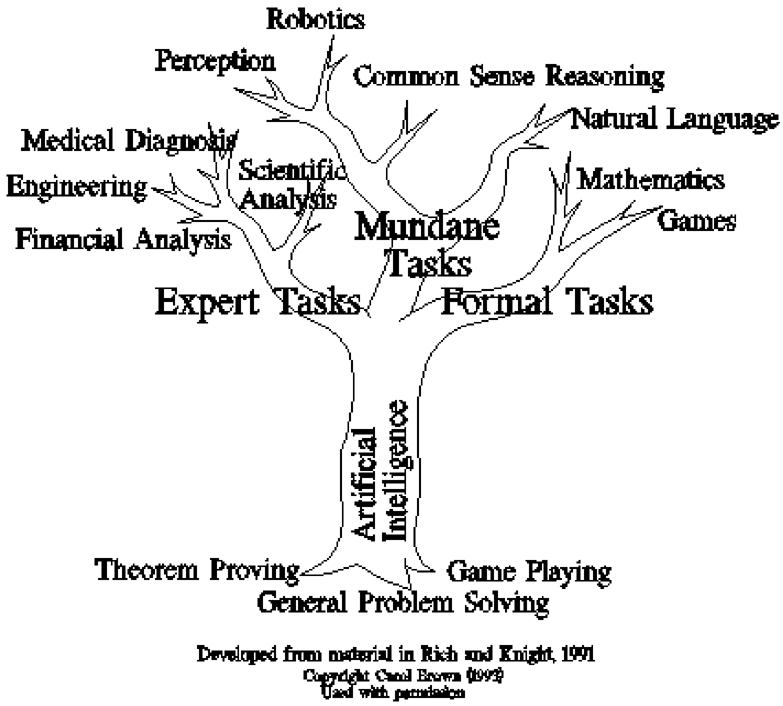


Figure A.11: Task Domains of Artificial Intelligence.

Artificial Intelligence domains:

- Formal Tasks (mathematics, games); and
- Expert tasks (financial analysis, medical diagnostics, engineering, scientific analysis, and other areas).

From a **business** perspective AI is a set of very powerful tools, and methodologies for using those tools to solve business problems.

From a **programming** perspective, AI includes the study of symbolic programming, problem solving, and search.

Typically AI programs focus on symbols rather than numeric processing.

- Problem solving – achieve goals.
- Search – seldom access a solution directly. Search may include a variety of techniques.
- AI programming languages include:
  - **LISP**, developed in the 1950s, is the early programming language strongly associated with AI. LISP is a functional programming language with procedural extensions. LISP (LISt Processor) was specifically designed for processing heterogeneous lists, typically a list of symbols. Features of LISP that made it attractive to AI researchers included run-time type checking, higher order functions (functions that have other functions as parameters), automatic memory management (garbage collection) and an interactive environment.
  - The second language strongly associated with AI is PROLOG. PROLOG was developed in the 1970s. **PROLOG** is based on first order logic. PROLOG is declarative in nature and has facilities for explicitly limiting the search space.

Object-oriented languages are a class of languages more recently used for AI programming. Important features of object-oriented languages include:

- Concepts of objects and messages;
- Objects bundle data and methods for manipulating the data;
- Sender specifies what is to be done receiver decides how to do it; and
- Inheritance (object hierarchy where objects inherit the attributes of the more general class of objects).

Examples of object-oriented languages are Smalltalk, Objective C, C++, JAVA. Object oriented extensions to LISP (CLOS – Common LISP Object System) and PROLOG (L&O – Logic & Objects) are also used.

A.5.1.2 AI Tools and Languages

AI researchers have often found that existing software tools and programming languages, and standard design and development methodologies, were too restrictive, and therefore developed new special purpose versions. However many of the ideas have been taken up more generally, so that the distinctions are no longer clear.

Some of the reasons for the special requirements are:

- Many AI problems are not well defined initially: in such cases the process of developing and testing software helps to clarify the problem.
- Often the tasks require development of very complex designs which are hard to get right. This requires powerful interactive testing facilities.
- AI problems often require different kinds of knowledge to be represented and different kinds of inference mechanisms to be used.
- Intelligent systems may need to modify themselves at run time.
- AI systems often need to construct complex networks of temporary structures, where different portions have different life-spans.

These requirements have led to the development of languages which:

- Are extendable using macros and other facilities.
- Include support for automatic garbage collection (once described by a well known computer scientist as a luxury for the idle rich).
- Use interpreters or incremental compilers.
- Have built in inference mechanisms or libraries.
- Allow modules to be modified or replaced at run time.
- Support multiple programming paradigms.
- Provide run-time type-checking instead of compile-time checking based on syntactic types.
- Include editors and other development tools as part of the run-time system.

Some of the specialised languages developed to support AI research and applications include Prolog, Scheme, Smalltalk, OPS-5 and other production system interpreters, several varieties of Lisp including Common Lisp, Pop2 and its derivatives, e.g. Pop-11, hybrid systems supporting more than one language, e.g. Loglisp, Poplog.

Using these and other languages, various tools have been developed to support knowledge acquisition and testing, theorem provers, planners, problem solvers, parsers and other forms of software for manipulating natural language, neural net toolkits, image processing tools, robot development tools, tools for designing and testing cognitively rich agents, tools for developing multi-agent systems, rule induction and learning systems, automatic program generating and testing tools, tools for doing experiments in artificial life, and tools for supporting evolutionary computation.

Some of the tools are closely related to particular theories, or intended to support particular types of techniques, e.g. constraint-manipulation toolkits, tools for building cognitive models based on SOAR, or ACT-R.

There have been some experiments in designing new forms of hardware to support AI, e.g. hardware tailored to playing chess, hardware for vision, hardware for implementing AI languages like Lisp or Prolog, hardware for neural computation, in addition to robots and robot components. In future there may be AI models or applications using entirely new forms of computers, e.g. quantum computers or DNA computers.

Many of the tools required for AI as engineering overlap with those required for AI as science, since the task of producing intelligent applied systems has much in common with the task of producing models of natural intelligent systems.

A.5.1.3 Branches of AI

There are many branches of Artificial Intelligence including:

- **Neural Networks** – These are systems that attempt to simulate intelligence by reproducing the types of physical connections that occur in animal brains.
- **Natural Language Processing** – This involves programming computers to understand natural human languages.
- **Robotics** – This field attempts to robots to act intelligently. For example to see and hear and react to other sensory stimuli.

- **Game Playing** – This involves programming computers to play games such as chess.
- **Expert Systems** – This is where computers are programmed to make decisions in real-life situations.

A.5.2 Expert Systems

A.5.2.1 Expert Systems Presentation

Definitions of expert systems vary. Some definitions are based on function. Some definitions are based on structure. Some definitions have both functional and structural components. Many early definitions assume rule-based reasoning.

A.5.2.1.1 Functional Components

What the system does (rather than how)

> "... a computer program that behaves like a human expert in some useful ways." [Winston and Prendergast, 1984, p.6]

Problem area:

- Solve problems efficiently and effectively in a narrow problem area.
- Typically, pertains to problems that can be symbolically represented.

Problem difficulty:

- Apply expert knowledge to difficult real world problems.
- Solve problems that are difficult enough to require significant human expertise for their solution.
- Address problems normally thought to require human specialists for their solution.

Performance requirement:

- The ability to perform at the level of an expert.
- Programs that mimic the advice-giving capabilities of human experts.
- Matches a competent level of human expertise in a particular field.
- Can offer intelligent advice or make an intelligent decision about a processing function.
- Allows a user to access this expertise in a way similar to that in which he might consult a human expert, with a similar result.

Explain reasoning:

- The capability of the system, on demand, to justify its own line of reasoning in a manner directly intelligible to the enquirer.
- Incorporation of explanation processes.

A.5.2.1.2 Structural Components

Use AI techniques:

- Using the programming techniques of artificial intelligence, especially those techniques developed for problem solving.

Knowledge component:

- The embodiment within a computer of a knowledge-based component, from an expert skill.
- A computer based system in which representations of expertise are stored.

The knowledge of an expert system consists of facts and heuristics. The 'facts' constitute a body of information that is widely shared, publicly available, and generally agreed upon by experts in the field.

Expert systems are sophisticated computer programs that manipulate knowledge to solve problems.

Separate knowledge and control:

- Make domain knowledge explicit and separate from the rest of the system.

Use inference procedures – heuristics – uncertainty:

- An intelligent computer program that uses knowledge and inference procedures.

The style adopted to attain these characteristics is rule-based programming.

Exhibit intelligent behavior by skillful application of heuristics.

The 'heuristics' are mostly private, little rules of good judgment (rules of plausible reasoning, rules of good guessing) that characterize expert-level decision making in the field:

- Incorporation of ... ways of handling uncertainty.

Model human expert can be thought of as a model of the expertise of the best practitioners in the field:

- Representation of domain-specific knowledge in the manner in which the expert thinks.
- Involving the use of appropriate information acquired previously from human experts.

A.5.2.1.3 How do People Reason?

They create categories (Cash is a Current Asset and a Current Asset is an Asset)

They use specific rules, a priori rules

e.g. tax law . . . so much for each deduction

Rules can be cascaded

"If A then B" . . .

"If B then C"

A—>B—>C

They Use Heuristics – "rules of thumb"

Heuristics can be captured using rules

"If the meal includes red meat

Then choose red wine"

Heuristics represent conventional wisdom

They use past experience – "cases"

Particularly evident in precedence-based reasoning

e.g. law or choice of accounting principles

Similarity of current case to previous cases provides basis for action choice

Store cases using key attributes

Cars may be characterized by: year of car; make of car; speed of car, etc.

What makes good argumentation also makes good reasoning

They use "Expectations"

"You are not yourself today"

If we differ from expectations then it is recognized

"Patterns of behavior"

A.5.2.1.4 How do Computers Reason?

- **Computer models are based on our models of human reasoning**
- **Frames:**
    - Frame attributes called "slots".
    - Each frame is a node in one or more "isa" hierarchies.

They use rules A—>B—>C

Auditing, tax…

Set of rules is called knowledge base or rule base

They use cases (Tax reasoning and tax cases where set of cases is called a case base)

They use pattern recognition/expectations (Credit card system and data base security system):

- A network of nodes and relations in some ways very similar to a traditional database and in other ways very different. Attributes called "slots", and value can be stated explicitly.
- A method for determining the value rather than the value itself .
- Each frame is a node in one or more "isa" hierarchies.
- Higher levels general concepts – lower levels specific.
- Unspecified value can be inherited from the more general node.
- Concept: prototypical representation with defaults that may be overridden.

- **Rule Based Reasoning**

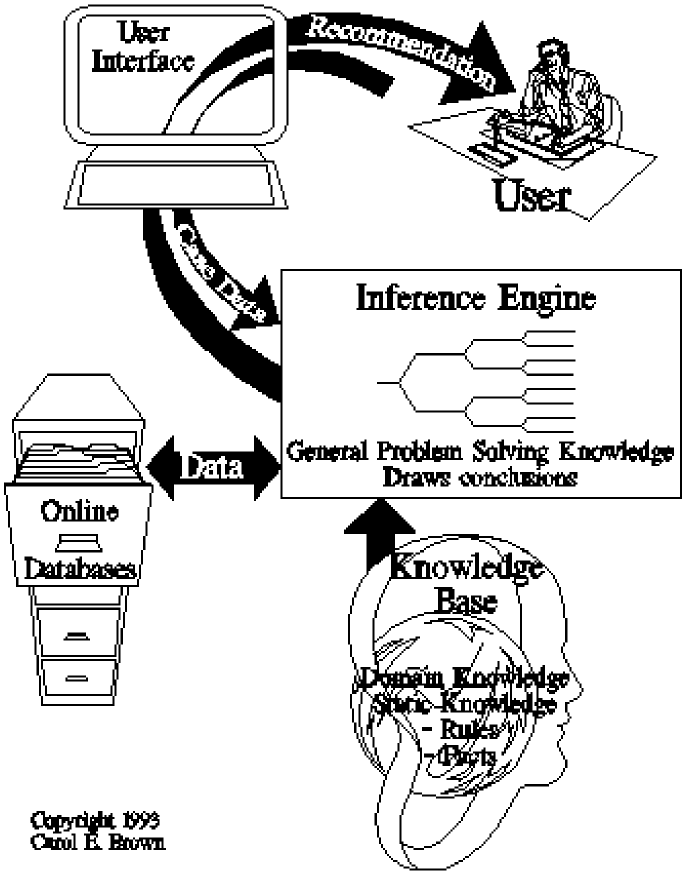


Figure A.12: Structure of a Rule-Based Expert System.

Currently, the most common form of expert system is:

- User Interface Friendly.
- Maybe "Intelligent".
- Knowledge of how to present information.
- Knowledge of user preferences...possibly accumulate with use.
- Databases.
- Contains some of the data of interest to the system.
- May be connected to on-line company or public database.
- Human user may be considered a database.

Inference Engine:

- General problem-solving knowledge or methods.
- Interpreter analyzes and processes the rules.
- Scheduler determines which rule to look at next.
- The search portion of a rule-based system.

- Takes advantage of heuristic information.
- Otherwise, the time to solve a problem could become prohibitively long.
- This problem is called the combinatorial explosion.
- Expert-system shell provides customizable inference engine.

Knowledge Base (rule base):

- Contains much of the problem solving knowledge.

Rules are of the form IF condition THEN action:

- Condition portion of the rule is usually a fact – (If some particular fact is in the database then perform this action).
- Action portion of the rule can include.
- Actions that affect the outside world (print a message on the terminal).
- Test another rule (check rule no. 58 next).
- Add a new fact to the database (If it is raining then roads are wet).

Rules can be specific, a priori rules (e.g. tax law…so much for each exemption) – represent laws and codified rules.

Rules can be heuristics (e.g. If the meal includes red meat then choose red wine). "rules of thumb" – represent conventional wisdom.

Rules can be chained together (e.g. "If A then B" "If B then C" since A—>B— >C so "If A then C") (If it is raining then roads are wet. If roads are wet then roads are slick.).

Certainty factors represent the confidence one has that a fact is true or a rule is valid.

- **Knowledge Engineering**

The discipline of building expert systems.

Knowledge acquisition:

- The process of acquiring the knowledge from human experts or other sources.
- Can involve developing knowledge to solve the problem.
- Knowledge elicitation.
- Coaxing information out of human experts.

Knowledge representation:

- Method used to encode the knowledge for use by the expert system.
- Common knowledge representation methods include rules, frames, and cases.
- Putting the knowledge into rules or cases or patterns is the knowledge representation process.

- **Case-Based Reasoning**

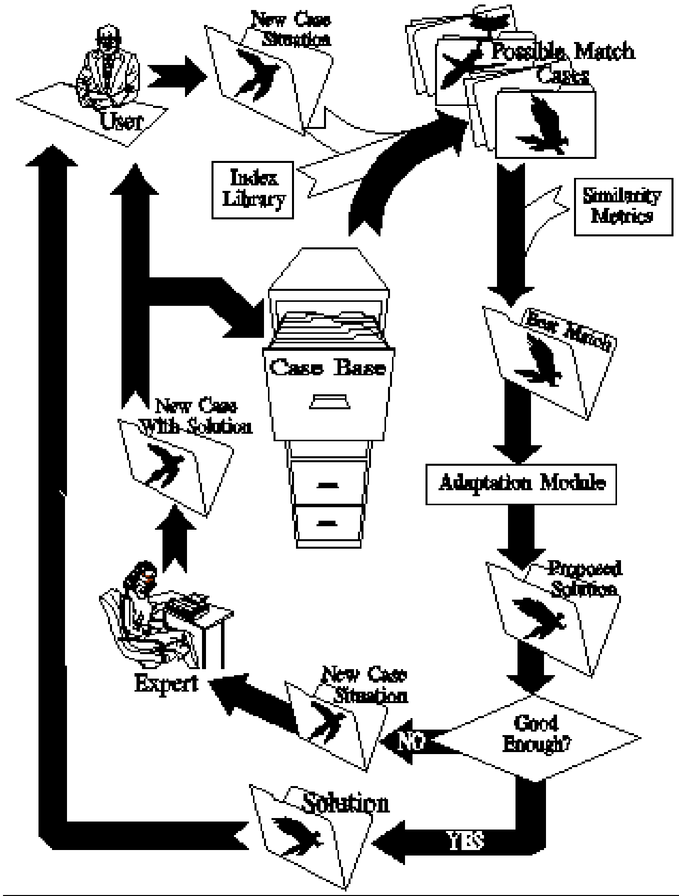


Figure A.13: The Case-Based Reasoning Process.

Uses past experiences.

Based on the premise that human beings use analogical reasoning or experiential reasoning to learn and solve complex problems.

Particularly evident in precedence-based reasoning (e.g. tax law or choice of accounting principles).

Useful when little evidence is available or information is incomplete.

Cases consist of:

- Information about the situation.
- The solution.
- The results of using that solution.
- Key attributes that can be used for quickly searching for similar patterns of attributes.

Elements in a case-based reasoning system:

- The case base – set of cases.
- The index library – used to efficiently search and quickly retrieve cases that are most appropriate or similar to the current problem.

- Similarity metrics – used to measure how similar the current problem is to the past cases selected by searching the index library.
- The adaption module – creates a solution for the current problem by either modifying the solution (structural adaptation) or creating a new solution using the same process as was used in the similar past case (derivational adaptation).

Learning:

- If no reasonably appropriate prior case is found then the current case and its human created solution can be added to the case base thus allowing the system to learn.

A.5.2.2 Advantages and Disadvantages

A.5.2.2.1 Advantages of Expert Systems

Permanence – Expert systems do not forget, but human experts may.

Reproducibility – Many copies of an expert system can be made, but training new human experts is time-consuming and expensive.

If there is a maze of rules (e.g. tax and auditing), then the expert system can "unravel" the maze.

Efficiency – can increase throughput and decrease personnel costs.

Although expert systems are expensive to build and maintain, they are inexpensive to operate.

Development and maintenance costs can be spread over many users.

The overall cost can be quite reasonable when compared to expensive and scarce human experts.

Consistency – With expert systems similar transactions handled in the same way. The system will make comparable recommendations for like situations.

Humans are influenced by:

- Recency effects (most recent information having a disproportionate impact on judgment).
- Primacy effects (early information dominates the judgment).

Documentation – An expert system can provide permanent documentation of the decision process.

Completeness – An expert system can review all the transactions, a human expert can only review a sample.

Timeliness – Fraud and/or errors can be prevented. Information is available sooner for decision making.

Breadth – The knowledge of multiple human experts can be combined to give a system more breadth that a single person is likely to achieve.

Reduce risk of doing business.

Consistency of decision making.

Documentation.

Achieve Expertise.

Entry barriers – Expert systems can help a firm create entry barriers for potential competitors.

Differentiation – In some cases, an expert system can differentiate a product or can be related to the focus of the firm.

Computer programs are best in those situations where there is a structure that is noted as previously existing or can be elicited.

A.5.2.2.2 Disadvantages of Rule-Based Expert Systems

Common sense – In addition to a great deal of technical knowledge, human experts have common sense**.** It is not yet known how to give expert systems common sense.

Creativity – Human experts can respond creatively to unusual situations, expert systems cannot.

Learning – Human experts automatically adapt to changing environments; expert systems must be explicitly updated. Case-based reasoning and neural networks are methods that can incorporate learning.

Sensory Experience – Human experts have available to them a wide range of sensory experience; expert systems are currently dependent on symbolic input.

Degradation – Expert systems are not good at recognizing when no answer exists or when the problem is outside their area of expertise.

A.5.2.3 Expert Systems Products

A.5.2.3.1 ILOG Business Rules

ILOG is the technology leader in high-performance rule engines that automate decision-making within many e-business applications. In today's constantly changing, short-turnaround Web environments, using ILOG components ensures that applications can adapt quickly, evolving ahead of the competition.

Business and e-business applications require tremendous flexibility in order to adapt to customer demands, regulatory changes and competition.

ILOG provides this flexibility with the most robust, versatile business rule management software available for C++ and Java.

Build applications that last with ILOG JRules and ILOG RulesThe Java and C++ rule engines of ILOG JRules and ILOG Rules share a common set of tools known as the Rule Kit. The Rule Kit includes:

- Rule Repository Open, extensible business rule repository for centrally storing and managing business rules, and deploying them throughout the enterprise.
- Rule Builder Integrated development environment for developing and debugging business rule applications.
- Business Rule Language Customizable and extensible business rule language, placing business rule power in the hands of business users.
- Rule Editor Powerful, adaptable Web-enabled and JavaBean components that can be embedded in applications.

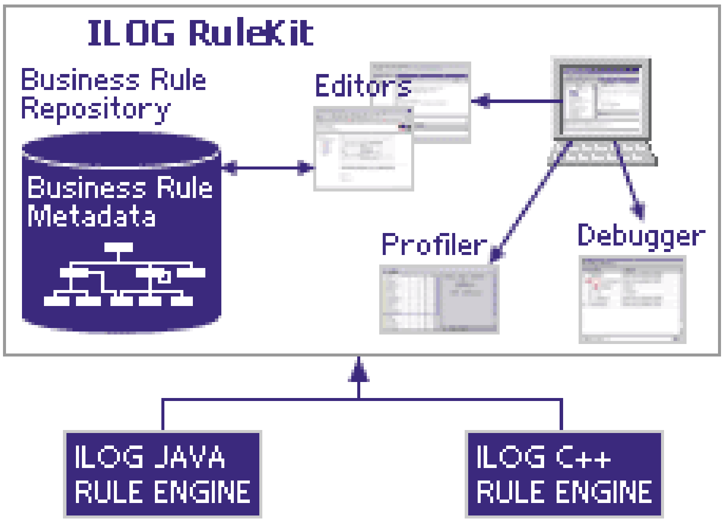


Figure A.14: ILOG Architecture.

Vendor: http://www.ilog.com

A.5.3 Neural Networks

A.5.3.1 Brief History of Neural Networks

The earliest work in neural computing goes back to the 1940s when McCulloch and Pitts introduced the first neural network computing model. In the 1950s, Rosenblatt's work resulted in a two-layer network, the perceptron, which was capable of learning certain classifications by adjusting connection weights. Although the perceptron was successful in classifying certain patterns, it had a number of limitations. The perceptron was not able to solve the classic XOR (exclusive or) problem. Such limitations led to the decline of the field of neural networks. However, the perceptron had laid foundations for later work in neural computing.

In the early 1980s, researchers showed renewed interest in neural networks. Recent work includes Boltzmann machines, Hopfield nets, competitive learning models, multilayer networks, and adaptive resonance theory models.

A.5.3.2 Artificial Neural Networks and Connectionist Models

Based on pattern recognition – used for credit assessment and fraud detection.

A set of interconnected relatively simple mathematical processing elements.

Looks for patterns in a set of examples and learns from those examples by adjusting the weights of the connections to produce output patterns.

Input to output pattern associations are used to classify a new set of examples.

Able to recognize patterns even when the data is noisy, ambiguous, distorted, or has a lot of variation.

Neural network construction and training:

- The architecture used (e.g. feed-forward).

- How the neurons are organized (e.g. an input layer with five neurons, two hidden layers with three neurons each, and an output layer with two neurons).
- The state function used (e.g. summation function).
- The transfer functions used (e.g. sigmoid squashing function).
- The training algorithm used (e.g. back-propagation).

Architecture shows how the processing elements are connected.

Commonly used architectures:

- Feed-forward.

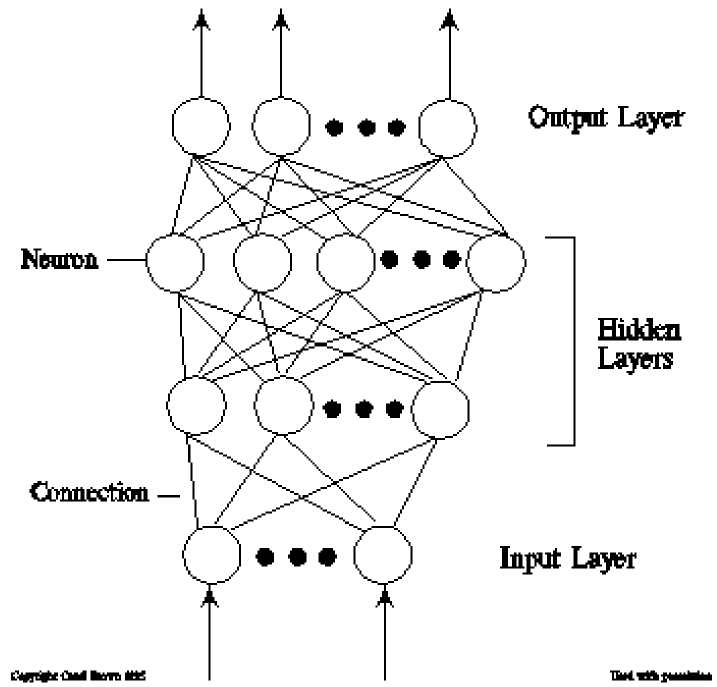


Figure A.15: Feed-Forward Neural Network Structure.

Feed-Forward Neural Network Structure is organized into a series of layers.

Layers (also called levels, fields or slabs):

- Input layer.
- One or more hidden layers.
- Output layer.

Some consider the number of layers to be part of architecture. Others consider the number of layers and nodes per layer to be attributes of the network rather than part of the architecture.

Neurons – the processing elements.

The vocabulary in this area is not completely consistent and different authors tend to use one of a small set of terms for a particular concept.

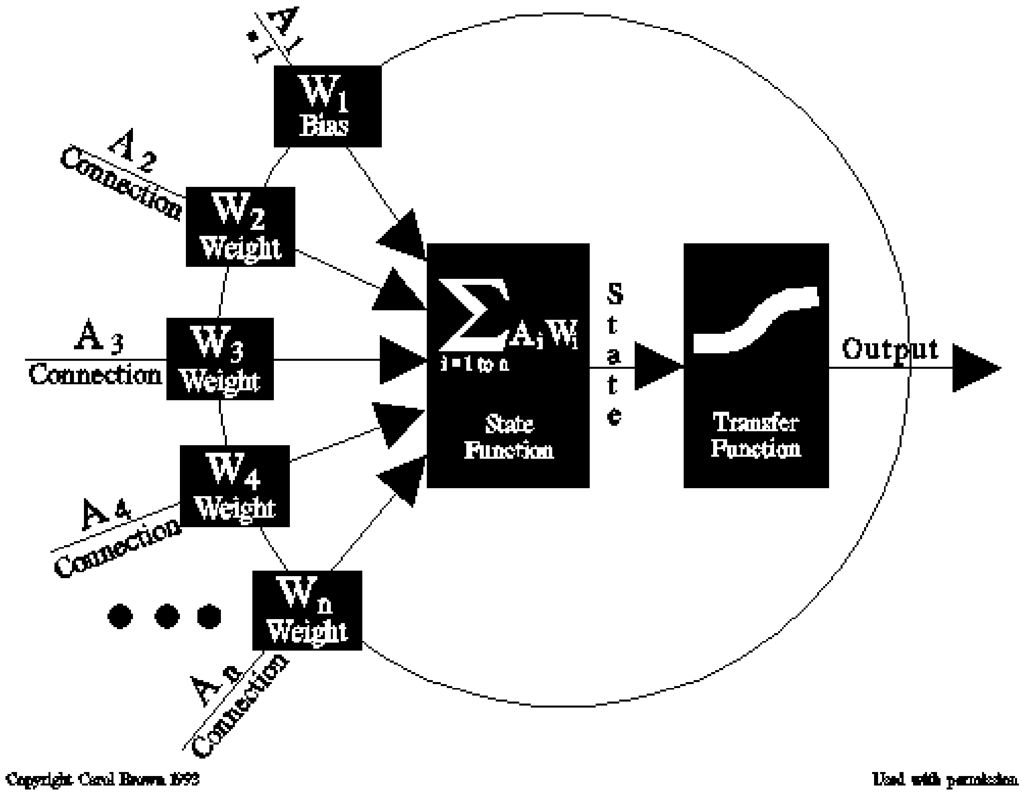


Figure A.16: Structure of a Neuron.

Structure of a Neuron consists of:

- A set of weighted input connections;
- A bias input;
- A state function;
- A non-linear transfer function; and
- An output.

Input connections have an input value that is either received from the previous neuron or in the case of the input layer from the outside.

Bias is not connected to the other neurons in the network and is assumed to have an input value of 1 for the summation function.

Weights

- A real number representing the strength or importance of an input connection to a neuron.
- Each neuron input, including the bias, has an associated weight.

State function

- The most common form is a simple summation function.
- The output of the state function becomes the input for the transfer function.

Transfer function

- A non-linear mathematical function used to convert data to a specific scale.
- Two basic types of transfer functions: continuous and discrete.
- Commonly used continuous functions used are Ramp, Sigmoid, Arc Tangent and Hyperbolic Tangent.
- Continuous functions sometimes called squashing functions.
- Commonly used discrete functions are Step and Threshold.
- Discrete transfer function sometimes called activation function.

Training

- The process of using examples to develop a neural network that associates the input pattern with the correct answer.
- A set of examples (training set) with known outputs (targets) is repeatedly fed into the network to "train" the network.
- This training process continues until the difference between the input and output patterns for the training set reaches an acceptable value.
- Several algorithms used for training networks.
- Most common is back-propagation.
- Back-propagation is done is two passes.
- First the inputs are sent forward through the network to produce an output.
- Then the difference between the actual and desired outputs produces error signals that are sent "backwards" through the network to modify the weights of the inputs.

A.5.3.3 Neural Networks Products

A.5.3.3.1 NeuroLab for Extend

Key features: NeuroLab is a neural network library for Extend (a graphical icon-based simulation program) and is a powerful and effective tool for understanding, designing and simulating artificial neural network systems. NeuroLab consists of more than 70 functional blocks and many example models. Implemented neural networks are Hopfield (regular and sparsely), perceptron, competitive (supervised and unsupervised), recurrent, context, feature map (2D to 1D and 2D to 2D), Boltzmann machine and single/ multi-layer feed-forward networks.

Each functional block in NeuroLab is easy to use and understand because of its intuitive appearance and interactive dialog box. Users can modify the network's parameters during simulation through each functional block's dialog box. NeuroLab is applicable in any field where non-linear mapping and adaptive capability are integral parts of problem solving. Major applicable fields are image processing, data classification, future prediction, adaptive control, dynamics identification, optimization and content addressable memory. NeuroLab can be used for Control of an Inverted Pendulum, Chinese Character Restoration, Exclusive OR, and Stock Prediction.

OS: Windows, Macintosh

Vendor: NeuroLab Department. http://www.mikuni.com

A.5.3.3.2 STATISTICA Neural Networks

STATISTICA Neural Networks is a comprehensive, state-of-the-art, powerful, and extremely fast neural network data analysis package, featuring:

Integrated pre- and post-processing, including data selection, nominal-value encoding, scaling, normalization and missing value substitution, with interpretation for classification, regression and time series problems. Exceptional ease of use coupled with unsurpassed analytic power; for example, a unique wizard-style Intelligent Problem Solver can walk the user step-by-step through the procedure of creating a variety of different networks and choosing the network with the best performance (a task that would otherwise require a lengthy "trial and error" process and a solid background in the underlying theory).

Powerful exploratory and analytic techniques, including Input Feature Selection algorithms (choosing the right input variables in exploratory data analysis -which is a typical application of neural networks is often a time-consuming process; STATISTICA Neural Networks can also do this for the user).

State-of-the-art, highly optimized training algorithms (including Conjugate Gradient Descent and Levenberg-Marquardt); full control over all aspects that influence the network performance such as activation and error functions, or network complexity. Support for combinations of networks and network architectures of practically unlimited sizes organized in Network Sets; selective training of network segments; merging, saving of network sets in separate files. API support for embedded solutions using Visual Basic, Delphi, C, C++ and other languages.

OS: Windows 95, Windows NT

Vendor: StatSoft. http://www.statsoft.com/stat nn.html

A.5.3.3.3 BIOSIM (Non-Commercial Software)

BIOSIM is a biologically-oriented neural network simulator. It implements four neuron models: a simple model only switching ion channels on and off, the original Hodgkin-Huxley model, the SWIM model (a modified HH model) and the Golowasch-Buchholz model (the most enhanced model). Dendrites consist of a chain of segments without bifurcation. BIOSIM includes a graphical user interface and was designed for research and teaching.

OS: information not available

Contact: Stefan Bergdoll Department of Software Engineering (ZXA/US), BASF Inc. 67056 Ludwigshafen, Germany Phone: +49-621-60-21372

Download: ftp://ftp.uni-kl.de/pub/bio/neurobio/

A.5.4 Fuzzy Logic

A.5.4.1 Fuzzy Logic Presentation

Fuzzy logic is an area of research based on the work of L.A. Zadeh. It is a departure from classical two-valued sets and logic, that uses "soft" linguistic (e.g. large, hot, tall) system variables and a continuous range of truth values in the interval [0,1], rather than strict binary (True or False) decisions and assignments. Fuzzy logic is used where a system is difficult to model exactly (but an inexact model is available), is controlled by a human operator or expert, or where ambiguity or vagueness is common. A typical fuzzy system consists of a rule base, membership functions, and an inference procedure.

A.5.4.2 Fuzzy Logic Products

A.5.4.2.1 DataEngine

DataEngine is the software for intelligent data analysis and data mining. By using neural networks, fuzzy logic and statistical methods, DataEngine provides you with the most advanced techniques for data analysis. Its flexible structure, interfaces, and powerful visualization module make it a must for everybody involved with complex data analysis tasks.

DataEngine can be extended by the user via user-defined function blocks. A DLL-based PlugIn interface enables you to integrate your own analysis tools into DataEngine in a simple and effective way. A number of examples of PlugIns are provided with this software.

OS: Windows 95, Windows 98, Windows NT

Vendor: MIT – Management Intelligenter Technologien GmbH http://www.mitgmbh.de

Demo: http://www.mitgmbh.de/mit/sp/demos/index.htm

A.5.4.2.2 Rule Maker

Key features: Rule Maker is an automatic rule generator add-on for CubiCalc or CubiCalc RTC. It uses neural networks and statistics to create variables, fuzzy sets, and rules automatically from data you supply. The product offers a choice of several methods. If you want to use fuzzy sets that you already have, Rule Maker can build a system from them. If you want build a system from scratch, it can create regularly-spaced fuzzy sets or place them according to clusters in your data. Either way, the resulting system is a full-fledged CubiCalc project.

Unlike purely neural-based automatic generation facilities, Rule Maker also handles sparse data sets. It can build a reasonable system with just a handful of data points. Rule Maker also includes new techniques invented at HyperLogic not available anywhere else.

OS: Windows 3.1 or higher, requires CubiCalc or CubiCalc RTC

Vendor: HyperLogic Corporation http://www.hyperlogic.com

A.5.4.2.3 WINROSA

Key features: WINROSA is a software that complements many tools already available for building fuzzy logic systems. Setting up a fuzzy rule base can be a very difficult task if there is no expert knowledge available regarding the process your modeling. WINROSA can assist you by automatically generating fuzzy rules from your data. It is based on the fuzzy ROSA method (Rule Oriented Statistical Analysis), which has been developed at the University of Dortmund. If you want to find relationships in your data and represent them as fuzzy rules, or if you wish to verify rule bases, then WINROSA is a must. WINROSA outstrips the performance of existing fuzzy sets completely.

OS: Windows 95, Windows NT

Vendor: MIT – Management Intelligenter Technologien GmbH http://www.mitgmbh.de

A.5.4.3 Genetic Algorithms

A.5.4.3.1 Genetic Algorithms Presentation

The Genetic Algorithm is a model of machine learning which derives its behavior from a metaphor of some of the mechanisms of evolution in nature. This is done by the creation within a machine of a population of individuals represented by chromosomes, in essence a set of character strings that are analogous to the base-4 chromosomes that we see in our own DNA. The individuals in the population then go through a process of simulated "evolution". Genetic algorithms are used for a number of different application areas. An example of this would be multidimensional optimization problems in which the character string of the chromosome can be used to encode the values for the different parameters being optimized. In practice, therefore, we can implement this genetic model of computation by having arrays of bits or characters to represent the chromosomes. Simple bit manipulation operations allow the implementation of crossover, mutation and other operations. Although a substantial amount of research has been performed on variable- length strings and other structures, the majority of work with genetic algorithms is focussed on fixed-length character strings. We should focus on both this aspect of fixed-length and the need to encode the representation of the solution being sought as a character string, since these are crucial aspects that distinguish genetic programming, which does not have a fixed length representation and there is typically no encoding of the problem. When the genetic algorithm is implemented it is usually done in a manner that involves the following cycle: Evaluate the fitness of all of the individuals in the population. Create a new population by performing operations such as crossover, fitness-proportionate reproduction and mutation on the individuals whose fitness has just been measured. Discard the old population and iterate using the new population. One iteration of this loop is referred to as a generation. There is no theoretical reason for this as an implementation model. Indeed, we do not see this punctuated behavior in populations in nature as a whole, but it is a convenient implementation model. The first generation (generation 0) of this process operates on a population of randomly generated individuals. From there on, the genetic operations, in concert with the fitness measure, operate to improve the population.

A.5.4.4 Genetic Algorithms Products

A.5.4.4.1 Evolver

Evolver is an optimization add-in for Microsoft Excel. Evolver uses innovative genetic algorithm (GA) technology to quickly solve complex optimization problems in finance, distribution, scheduling, resource allocation, manufacturing, budgeting, engineering, and more. Virtually any type of problem that can be modeled in Excel can be solved by Evolver, including previously unsolvable problems. Evolver requires no knowledge of programming or GA theory and ships with a fully illustrated manual, several examples, and free, unlimited technical support. Evolver is available in three versions: Standard, Professional, and Industrial. The Professional and Industrial versions have increased problem capacities and advanced features, including the Evolver Developers Kit.

OS: Windows 3.x, Windows 95, Windows 98, Windows NT

Microsoft Excel 5.0 or higher
Vendor: Palisade Corporation. http://www.palisade.com

Demo: http://www.palisade.com/html/trial—versions.html

A.5.4.4.2 GECO (Non-Commercial Software)

GECO (Genetic Evolution through Combination of Objects) is an extensible, object-oriented framework for prototyping GENETIC ALGORITHMs in Common Lisp. GECO makes extensive use of CLOS, the Common Lisp Object System, to implement its functionality. The abstractions provided by the classes have been chosen with the intent both of being easily understandable to anyone familiar with the paradigm of genetic algorithms, and of providing the algorithm developer with the ability to customize all aspects of its operation. It comes with extensive documentation, in the form of a PostScript file, and some simple examples are also provided to illustrate its intended use.

OS: information not available. Contact: George P.W. Williams, Jr. 1334 Columbus City Rd. Scottsboro, AL 35768, U.S.A. e-mail: george@hsvaic.hv.boeing.com

Download: ftp://ftp.aic.nrl.navy.mil/pub/galist/src/

A.6 REFERENCES

A.6.1 Artificial Intelligence

Expert Systems in Finance, Daniel E. O'Leary and Paul R. Watkins (Editors), 1992.

Expert Systems in Business and Finance: Issues and Applications, Paul R. Watkins and Lance B. Elliot (Editors), 1993.

Artificial Intelligence in Accounting and Auditing: Using Expert Systems, Volume II, Miklos A. Vasarhelyi and B.N. Srinidhi (Editors), 1993.

Expert Systems introduction, Zbigniew Swiatnicki, 1998.

Neural Networks introduction, Zbigniew Swiatnicki and Roman Wantoch-Rekowski, 1999.

A.6.2 Simulation

Techniques for Monte Carlo Optimizing, Arsham, H., Monte Carlo Methods and Applications, 4(3), 181-230, 1998.

Stochastic Optimization of Discrete Event Systems Simulation, Arsham, H., Microelectronics and Reliability, 36(10), 1357-1368, 1996.

Conditional Monte Carlo: Gradient Estimation and Optimization Applications, Fu, M. and Hu, J-Q., Kluwer Academic Publishers, 1997.

Estimating the optimum of a stochastic system using simulation, Rollans, S. and McLeish, D., Journal of Statistical Computation and Simulation, 72, 357-377, 2002.

Discrete Event Systems: Sensitivity Analysis and Stochastic Optimization by the Score Function Method, Rubinstein, R. and Shapiro, A., John Wiley & Sons, 1993.

Bossel, H., *Modeling & Simulation*, Peters, A.K., Pub., 1994. Delaney W., and Vaccari, E., *Dynamic Models and Discrete Event Simulation*, Dekker, 1989.

Fishman, G., Discrete-Event Simulation: Modeling, Programming and Analysis, Springer-Verlag, Berlin, 2001.

Fishwick, P., Simulation Model Design and Execution: Building Digital Worlds, Prentice-Hall, Englewood Cliffs, 1995.

Ghosh, S. and Lee, T., Modeling & Asynchronous Distributed Simulation: Analyzing Complex Systems, IEEE Publications, 2000.

Gimblett, R., Integrating Geographic Information Systems and Agent-Based Modeling: Techniques for Simulating Social and Ecological Processes, Oxford University Press, 2002.

Harrington, J. and Tumay, K., Simulation Modeling Methods: An Interactive Guide to Results-Based Decision, McGraw-Hill, 1998.

Haas, P., Stochastic Petri Net Models Modeling and Simulation, Springer Verlag, 2002.

Hill, D., Object-Oriented Analysis and Simulation Modeling, Addison-Wesley, 1996.

Kouikoglou, V. and Phillis, Y., *Hybrid Simulation Models of Production Networks*, Kluwer Pub., 2001.

Law, A. and Kelton, W., *Simulation Modeling and Analysis*, McGraw-Hill, 2000.

Nelson, B., Stochastic Modeling: Analysis & Simulation, McGraw-Hill, 1995.

Oakshott, L., *Business Modelling and Simulation*, Pitman Publishing, London, 1997.

Pidd, M., Computer Simulation in Management Science, Wiley, 1998.

Rubinstein, R. and Melamed, B., *Modern Simulation and Modeling*, Wiley, 1998.

Severance, F., System Modeling and Simulation: An Introduction, Wiley, 2001.

Van den Bosch, P. and Van der Klauw, A., *Modeling, Identification & Simulation of Dynamical Systems*, CRC Press, 1994.

Woods, R. and Lawrence, K., *Modeling and Simulation of Dynamic Systems*, Prentice Hall, 1997.

NATO RTG SAS-045
on
Computer Based Decision Support Tool for Helicopter Mission Planning in Disaster Relief and Military Operations

Annex B – TECHNICAL REPORT 2: OVERVIEW OF INFORMATION TECHNOLOGIES

Hakan Çanlı
1st Lieutenant Turkish Air Force
Scientific Decision Support Branch
06100 Ankara, Turkey

B.1 INTRODUCTION

This document is prepared to fulfil the requirements of the Programme of Work (POW) of NATO SAS-045 RTG on "Computer Based Decision Support Tool for Helicopter Mission Planning in Disaster Relief and Military Operations".

The POW dictates to provide an a-priori analysis of information technologies that are anticipated to support the development of NATO SAS-045 project. As it is stated in the Terms of Reference (TOR) of the afore-mentioned project, the main goal of the research is to provide the basis for developing a generic and flexible decision support tool for effective management of helicopter missions by conducting the problem analysis, investigating the concept of solutions and determining relevant technical requirements.

It is foreseen that different models, technologies and solution approaches will be utilized concurrently, and integration and interfacing will pose itself as an important technical issue within the scope of maintaining interoperability and standardization in NATO practices.

Thus, Work Item #2 of the POW states that during the Analysis Phase of the project technology surveys should be carried out on modeling, computer (software engineering) and data collection technologies; geographical information systems, digital maps, mission data compilation systems; model, data, and knowledge management repositories in NATO nations. Then, the technology mapping and capability matrix can be developed using the identified current needs and capability gaps.

In this document, it is intended to provide a broad overview that will guide any potential research dealing with the design and development of a generic decision support tool within a similar NATO context, not limited to helicopter operations.

B.2 INFORMATION SYSTEM DESIGN

The need for a faster and reliable decision support systems rises as a result of the rapid progress in both business and technology. Although hardly noticed if it is working properly, the Database Management Systems underpin all the activities of a decision support system by providing data storage and retrieval technology. This document intends to present the key points to consider in design of the information structure for decision support systems. Some DMBS commercial off-the-shelf systems are also presented.

An information system, with its very broad definition is a – computer – system that is used for the storage and retrieval of any type of information – text, numerical, graphical, video, and sound. Before designing an information system, an analysis is done to find out the requirements. For decision support systems, a mathematical model should also be considered in the analysis phase. Then, in the design phase, focus is directed towards the realization of the system to meet these requirements. Since mathematical model development [FON03] and models in NATO nations [SMI03] are two other technology surveys within analysis phase of NATO SAS-045 project, this document focuses only on the data management issues for information systems.

B.2.1 Software Architecture

An information system is composed of the data and the software, which are principally managed separately. Software architecture design is the focus of software engineering on decomposition of large systems into layers and/or partitions [MAT01]. As described by Booch, Rumbaugh, and Jacobson, "an architecture is the set of significant decisions about the organization of a software system, the selection of the structural elements and their interfaces by which the system is composed, together with their behavior as specified in the collaborations among those elements, the composition of these structural and behavioral elements into progressively larger subsystems, and the architectural style that guides this organization- these elements and their interfaces, their collaborations, and their composition" [BOO99].

An architectural design decision may be made from a broad-scoped or system perspective. Any decision that could be made from a more narrowly-scoped, local perspective is not architectural [BRE02].

	Low Impact	High Impact *(high priority, important to business)*
Systemic *(broad scope)*	not architectural (*this could be a trap*)	**focus of architectural decisions**
Local	not architectural	not generally architectural *(though might set architecture guidelines and policies as needed)*

Figure B.1: Decision Scope and Impact [BRE02].

Three levels of architecture can be distinguished as shown in Figure B.2.

Conceptual Architecture

- Architecture Diagram, CRC-R cards
- *Focus: identification of components and allocation of responsibilities to components*

Logical Architecture

- Updated Architecture Diagram (showing interfaces), Interface specifications, Component specifications and usage guides
- *Focus: design of component interactions, connection mechanisms and protocols; interface design and specification; providing contextual information for component users*

Execution Architecture

- Process View (shown on Collaboration Diagrams)
- *Focus: assignment of the runtime component instances to processes, threads and address spaces; how they communicate and coordinate; how physical resources are allocated to them*

Figure B.2: Architecture Views [BRE02].

The three levels of architecture might further be decomposed into structural and behavioral views. The structural view focuses on elements and their relations while behavioral view searches an answer to the question "how does it work?" [BRE02].

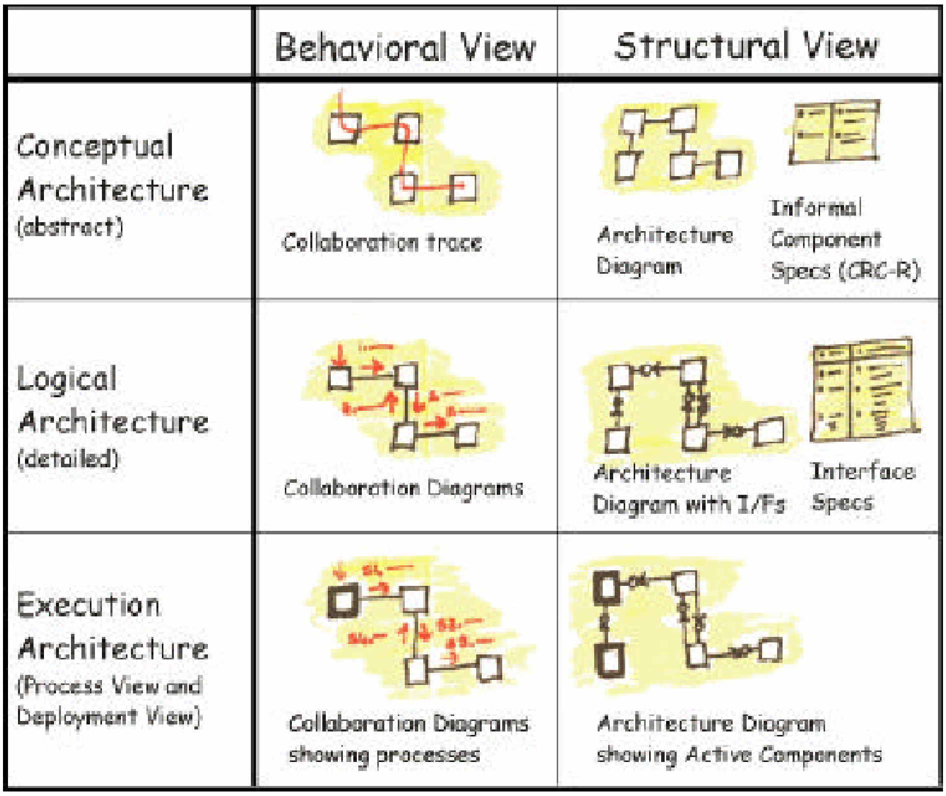


Figure B.3: Architecture Views with Structure and Behaviour [BRE02].

B.2.2 Data Management

There are three major decisions for an information system designer:

1) Data management approach;
2) Data interaction strategy; and
3) DBMS paradigm.

B.2.2.1 Data Management Approaches

B.2.2.1.1 In-Memory Data

Since memory needs to be constantly powered or hardware persistent, in-memory data management approach cannot support large systems. However, it's suitable for small data sets as in Palm organizer, Nintendo carts, etc.

B.2.2.1.2 Files

Files can be accessed sequentially (flat) or randomly (binary). As data size grows, data management and access efficiency becomes a problem. Files are usually suitable for raw sensor data, debug dumps, etc.

B.2.2.1.3 Database Management System (DBMS)

Data Base Management System is a complex set of programs that control the organization, storage, retrieval and security of data for users. Although they require administrative work, larger file sizes and more CPU resources, compared to in-memory data and files, DBMSs provide excellent performance for large data sets and flexibility in accessing data.

B.2.2.1.4 Data Management Approaches Comparison

Table B.1: Data Management Approaches Comparison [MAT01]

	In-memory data	Files	DBMSs
Data Persistence	Requires special hardware	Strong support	Strong support
Purchase cost	Special hardware cost	None	Can be costly
Lifecycle cost	Variable	Variable	Variable
Large quantities of data	Limited by Hardware	Limited by size of available disk space	Limited by size of available disk space
Performance	Very fast	"depends"	Fast
Extensibility	Limited	Limited	Excellent
Concurrent access	Object locking	File locking	Object/record locking
Crash recovery	Shadow memory	Backup files	Log files/rollbacks
Integrity	None	None	Designer can specify rules
Transaction Support	None	None	Excellent
Distribution	None	None	Sometimes
Query Language	None	Partial (ODBC/JDBC)	Powerful
Security	None	OS Protections	Simple to Extensive
Metadata	None	None	Yes
Portability *	Depends on application	Very easy	Might be time-consuming in case of complex relations
Qualification *	Very easy	Fairly easy	Requires special training

* Lines inserted into the original table by author

B.2.2.2 Data Interaction Strategy

Depending on the implementation requirements, information system designer may choose one or a group of following data interaction strategies to communicate data between data management and the application:

- Batch/Script;
- Embedded Queries (such as SQL/OQL);
- Database API;
- Stored Procedures;
- Generic OO Layer; and
- Metamodel-driven Interaction.

B.2.2.3 DBMS Paradigm

DBMSs are based on several different paradigms each of which is designed with a specific problem, industry or set of functions in mind. This section surveys the main types of database management systems.

B.2.2.3.1 Hierarchical DBMS

The Hierarchical model is the oldest of the database models derived from the Information Management Systems of the 1950s and 1960s. Data is organized in a series of records, which have a set of field values attached to it. It collects all the instances of a specific record together as a record type. These record types are analogous to the tables in the relational model, where the individual records are the rows in a table. To create links between these record types, the hierarchical model uses “Parent-Child” relationships. These are a one-to-many mapping between record types.

The records linked with parent-child relationships are organized as a single tree. From this aspect, the hierarchical model is not able to cope with linking between branches or over multiple layers. For example, we could have a tree representing the departments, sections and branches in the armed forces; however we could not specify one branch working for more than one department such as Air Force Operations Center’s relation with both Air Force Operations and General Staff Operations Center. To do this we would have to create two instances of the AFOC, which could cause concurrency inaccuracies.

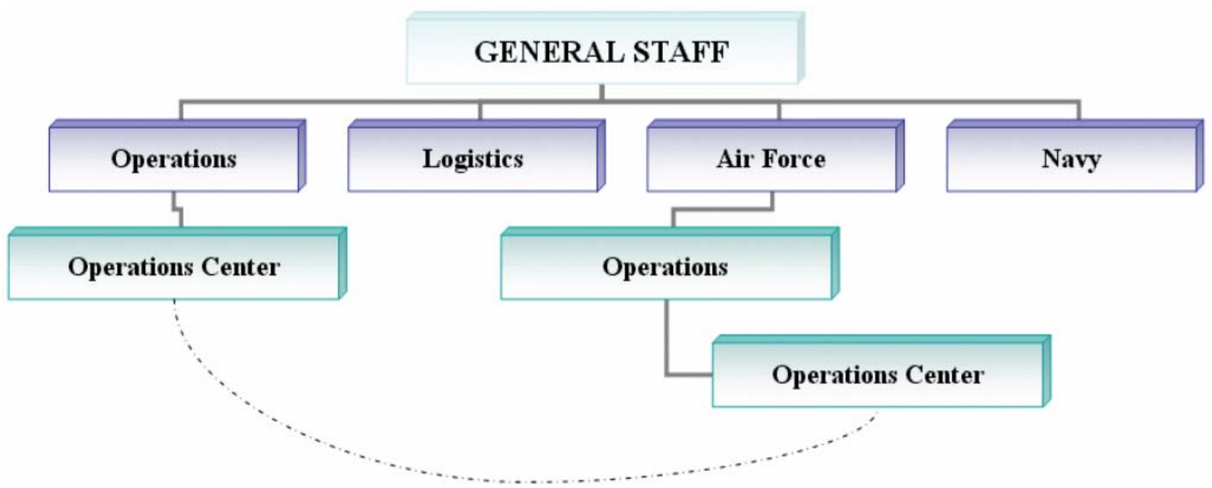


Figure B.4: Limitation to Linking Over Multiple Layers.

Although it does not allow linking over multiple layers, the hierarchical model with its single tree is much more structured than the relational model with improved throughput for transactions and simplicity of the interface for users. The hierarchical model is no longer used as the basis for current commercially produced systems; however, there are a large number of legacy installations that are likely to be phased out over time. [CER03]

B.2.2.3.2 Network DBMS

The Network model introduced in 1970s is a contemporary version of the Relational Model, both in terms of its age and its basic research done in the 1960s. Network Model represents the data in the form of a network of records and sets, which are related to each other, forming a network of links. This model is only used in legacy systems and is being phased out over time.

Records in a network database are sets of related data values equivalent to the rows in the relational model. They store the name of the record type, the attributes associated with it and the format for these attributes. Record Typesare set of records of the same type analogous to the tables in the relational model. "Associated with" relationship may occur between various record types. This relationship doesn't have a direct counterpart in the relational model, but it is similar to a query statement, which joins two tables together. This relationship makes the network model faster with certain queries at the cost of the flexibility and adaptability of the relational model.

An example of "Associated with" relationship would be the relationship between a squadron and the pilots in it. The network model uses a Bachman diagram to represent this relationship as shown below [CER03].

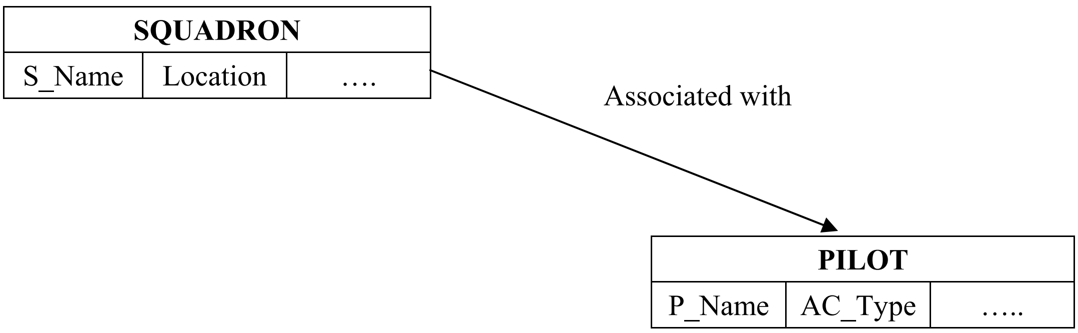


Figure B.5: "Associated with" Relationship in a Bachman Diagram.

The network model is not commonly used today to design database systems; however, there are a few instances of it being used by companies as a part of a legacy system [CER03].

B.2.2.3.3 Relational DBMS (RDBMS)

B.2.2.3.3.1 RDBMS Structure

Relational database systems (RDBMS) based on Set Theory and Predicate Logic have arisen out of the concept of data structures and relational algebra by E. F. Codd. RDBMSs organize data within "tables" and relationships can be expressed between tables and data elements. The rows of the tables are also called "tuples", and there is one tuple component for each attribute (column) in that table.

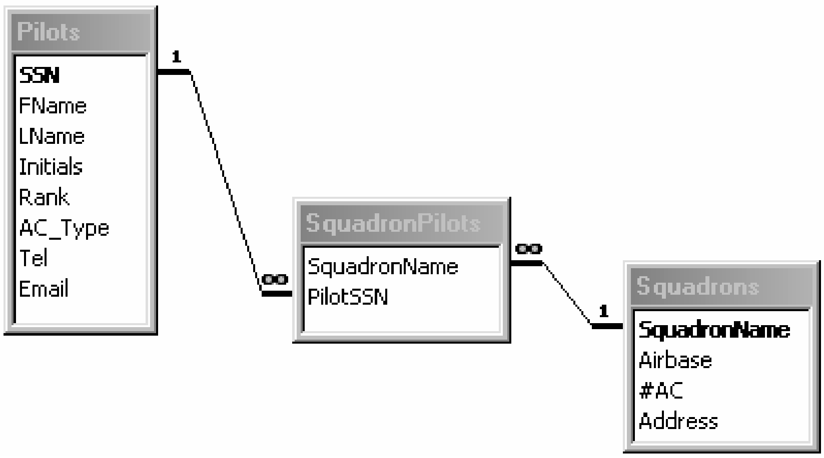


Figure B.6: Relational Design for Squadron Pilots.

In this model, the physical implementation of the database is abstracted away from the user, and the Structured Query Language (SQL) is used to extract and update data and conform as closely as possible to the theoretical relational rules of normalization. Operations, which can be carried out on the data, include "insert", "query" and "delete" commands. Since SQL is the relational database standard, the commands for most systems are almost exactly the same, with only some of the more complex operations differing slightly [CER03, BIB03]. Following SQL command inserts a new pilot into the table as shown in Figure B.7.

Insert Into Pilots Values (596874568867,'Hüseyin','Duman','Capt.','Falcon');

Insert Into SquadronPilots Values ('113. Fighter Sq.',596874568867);

Pilots : Table

SSN	FName	LName	Rank	AC_Type
287365874325	Hakan	Çanlı	1Lt	Cn235
467675643856	Ramazan	Toper	1Lt	G-IV
596874568867	Hüseyin	Duman	Capt.	Falcon

Squadrons : Table

SquadronName	Airbase
15. Transport Sq.	10th AFB
113. Fighter Sq.	9th AFB

SquadronPilots : Table

SquadronName	PilotSSN
15. Transport Sq.	287365874325
113. Fighter Sq.	596874568867
15. Transport Sq.	467675643856

Figure B.7: Table Representation for Squadron Pilots.

Table B.2: Advantages and Disadvantages of the RDBMSs [BIB03]

Advantages	Disadvantages
1) Being the most popular type of DBMS in use, technical development effort ensures that advances appear quickly and reliably. **2)** Many third party tools are tuned to work with the popular Relational DBMS via standards such as Open Database Connectivity (ODBC). **3)** It offers distributed database and distributed processing options for large consortium libraries. **4)** It includes extremely well developed management tools and security with automatic data logging and recovery. **5)** Referential integrity ensures data consistency. **6)** Transactional integrity features ensure that incomplete transactions do not occur.	**1)** Since RDBMSs have to employ many tables to conform absolutely to various normalization rules, they may be slow and resource hungry. **2)** SQL does not provide an efficient way to browse alphabetically through an index. Thus some systems cannot provide a simple title A-Z browse.

B.2.2.3.3.2 RDBMS Connectivity

Application programmer can establish connection to a database management system through several methods to access and execute operations over data. The programmer can use the proprietary programming interfaces (API) provided with the DBMS or standards such as Microsoft's Open Database Connectivity Standard and ActiveX Data Objects (ADO) or Java Database Connectivity (JDBC).

The ODBC standard provides an open, non-proprietary definition – a low-level set of calls for applications and DBMSs to exchange instructions and share data without knowing anything about each other. Typically, custom ODBC middleware drivers must be developed to transform ODBC calls into vendor-specific access requests and responses. ODBC is a low-level functional (non-object orientated) API for accessing databases. ADO provides an object orientated layer on top of ODBC, or can even operate independently of ODBC. The X/Open Group and ISO have made ODBC a standard, though there are differences from this standard and the Microsoft implementation.

The ODBC interface defines: [OPE03]:

- A library of ODBC function calls that allow an application to be connected to a DBMS, to execute SQL statements, and retrieve results.
- A standard way to connect and log on to a DBMS.
- A standardized representation for data types.

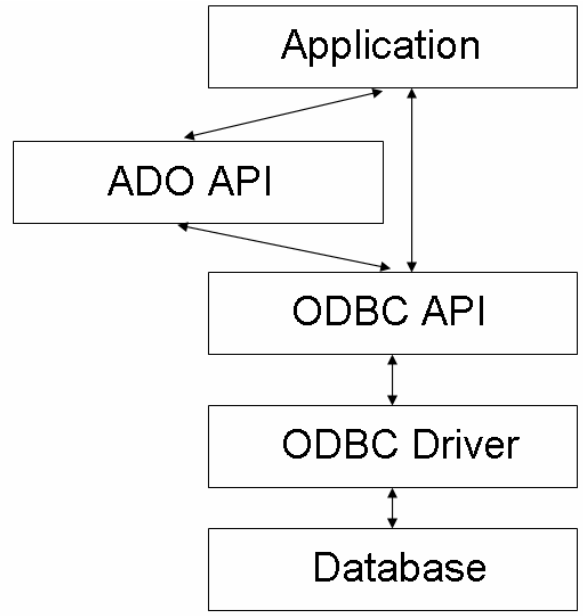


Figure B.8: ADO-ODBC Structure.

Java DataBase Connectivity on the other hand is the primary way of connecting to an SQL compatible database using the Java programming language. It's basically Java based counterpart of ODBC. JDBC gives the programmer a series of objects to represent such database concepts as connections, queries and result sets. Like ODBC, JDBC uses database drivers. Four types of connection have been defined for JDBC as shown in Figure B.9 and Figure B.10:

- Type 1: JDBC – ODBC bridge;
- Type 2: Partial Java Driver;
- Type 3: Pure Java Driver for Middleware; and
- Type 4: Pure Java, Direct-to-DB Driver.

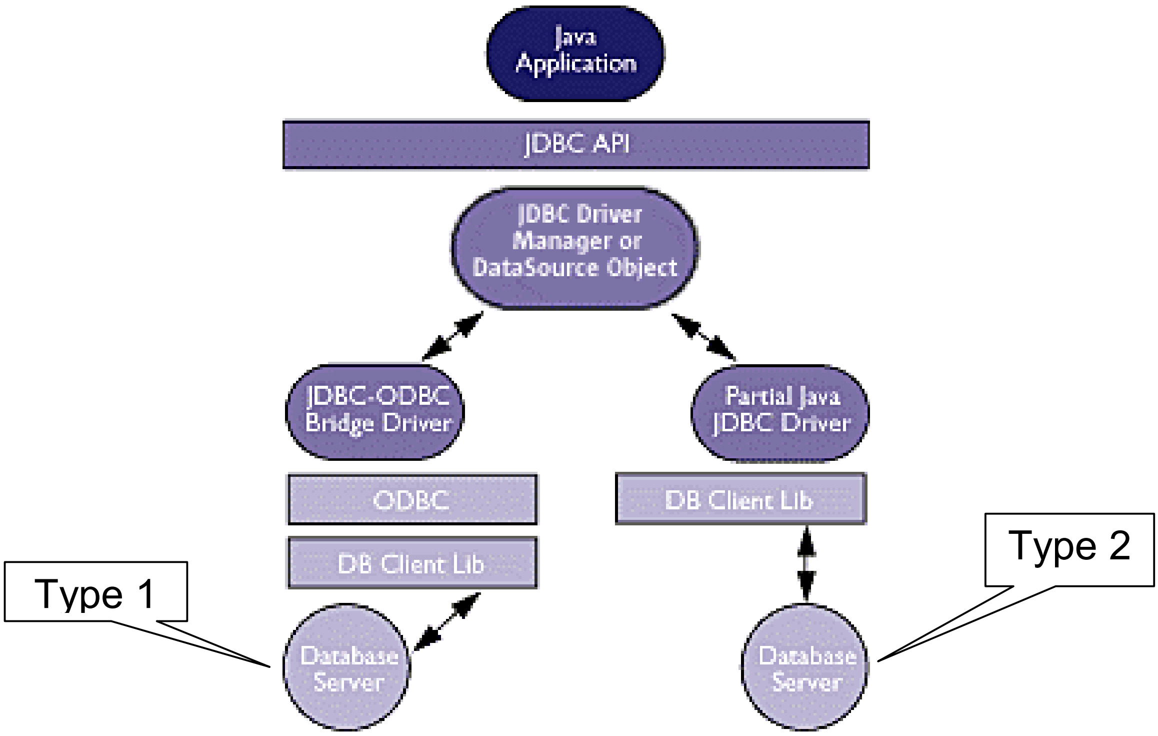


Figure B.9: JDBC Type-1/Type-2 Structure [SUN03].

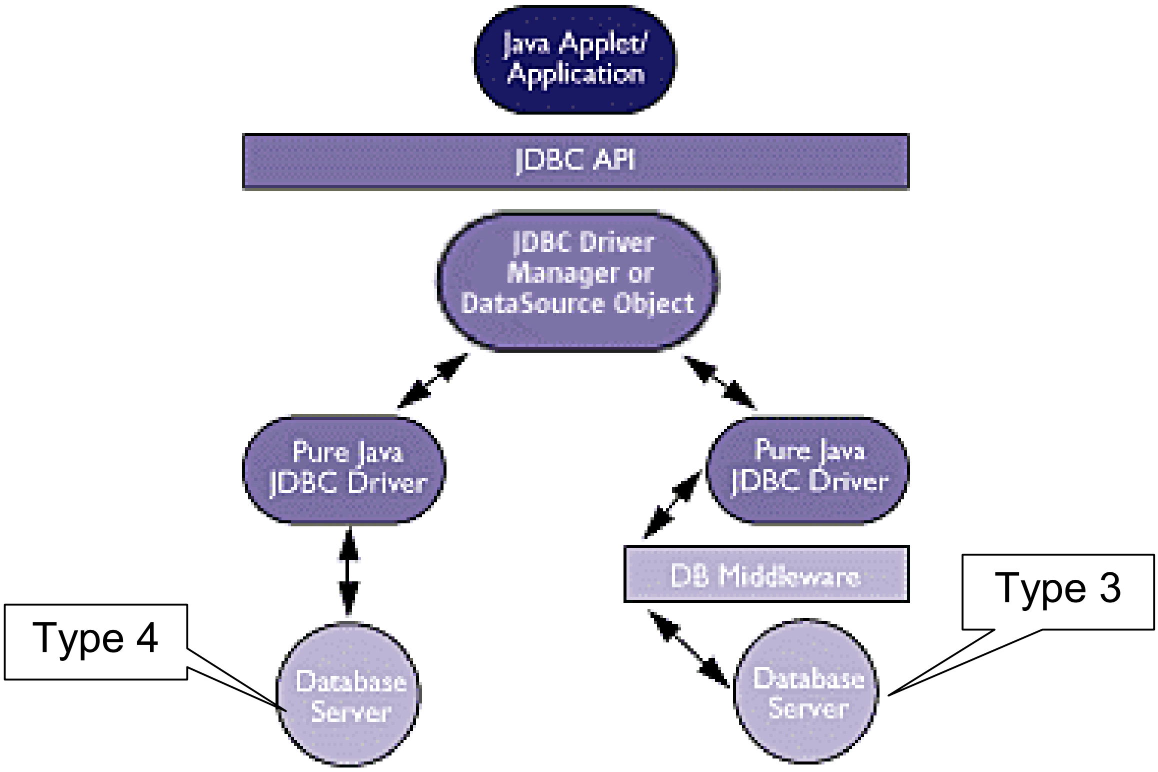


Figure B.10: JDBC Type-3/Type-4 Structure [SUN03].

B.2.2.3.4 Object-Oriented DBMS (OODBMS)

B.2.2.3.4.1 Object-Oriented Paradigm

Object-orientation is an approach to problem solving which seeks to identity the relevant objects in the problem domain. These objects are then defined and employed to solve the problem. James Rumbaugh, Michael Blaha, William Premerlani, Frederick Eddy and William Lorensen defined the term "object-oriented" as follows:

Superficially the term "object-oriented" means that we organize software as a collection of discrete objects that incorporate both data structure and behavior. This is in contrast to conventional programming in which data structures and behavior are only loosely connected [RUM91].

An "object" is the most fundamental concept in the object-oriented paradigm. It is a conceptual (logical or physical) entity composed of attributes and methods. Attributes hold the data that determine the state of the object, and methods determine the behavior of the object based on its current state. An object is normally referred to by a name and has an "identity." Attribute values of an object might change in time, perhaps as a result of performing a behavior, but it would still be the same object [STE99]. A UML (Unified Modeling Language) representation of object is shown in Figure B.11 [MUL97].

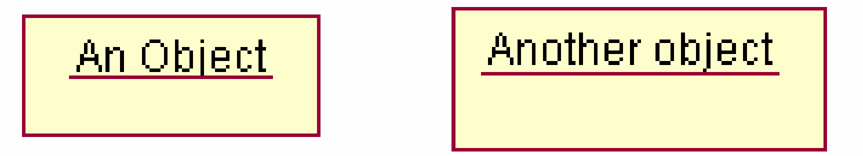


Figure B.11: UML Object Representation.

A "class" is an abstract representation for some particular type of object. Often described as a blueprint for an object, it defines objects of that type. Objects are built from the class by a process named "instantiation." As a result, any object is an instance of a class. Figure B.12 shows the UML representation of class. Different types of relationships are applicable between classes. An "association" relationship is a semantic connection shown with a line between classes or objects as in Figure B.13 [MUL97].

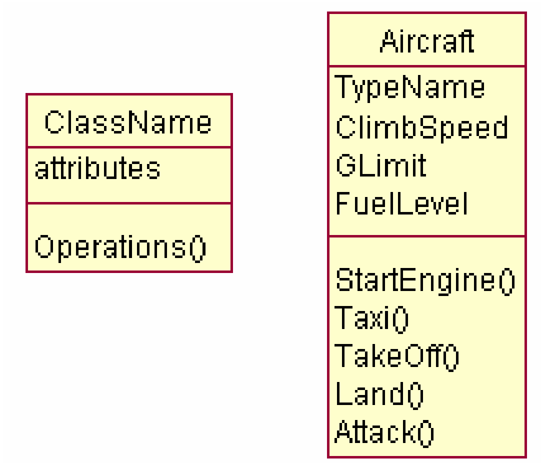


Figure B.12: UML Class Representation.

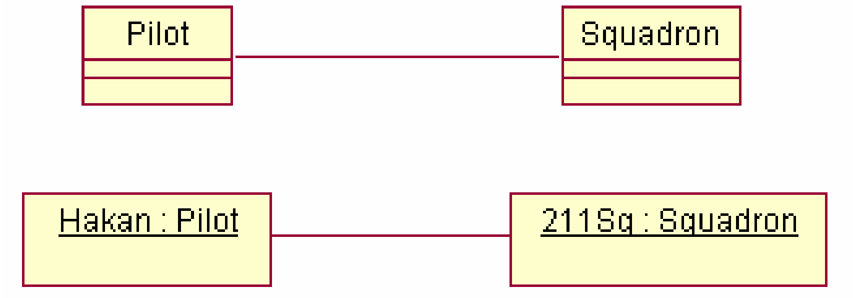


Figure B.13: UML Association Relationship.

By default, an association expresses a weak coupling between abstractions. An "aggregation" is a special type of association expressing a strong coupling. Aggregation indicates relationships like "part of," "composed of," or "master and slave." It is represented with a diamond. UML also defines even stronger coupling, "composition," meaning that when the owner object is deleted it results in the deletion of its composite objects. Composition is represented with a filled diamond [MUL97].

Inheritance is a relation where one class has all the properties and methods of its parent and extends it by including additional methods or variables. Classes are ordered within an inheritance hierarchy. A "superclass" is an abstraction of its "subclasses." The UML representation of an inheritance relation is shown in Figure B.15 [MUL97].

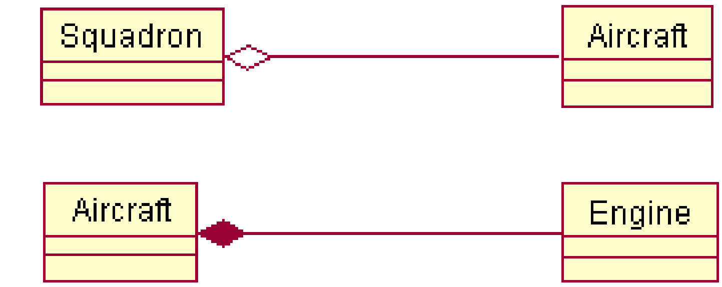


Figure B.14: UML Aggregation and Composition Relationships.

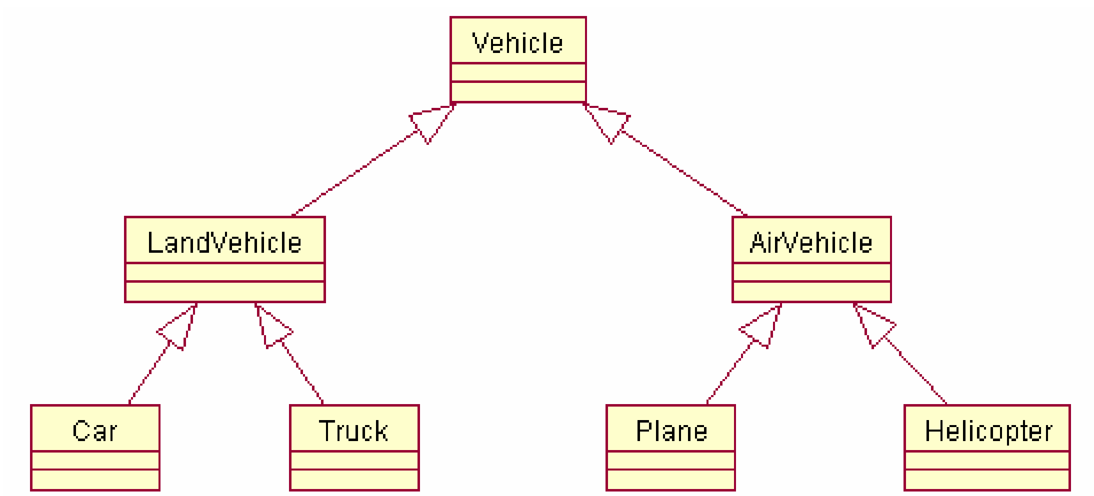


Figure B.15: UML Inheritance Relationships.

Abstraction, encapsulation, inheritance, reuse, and emphasis on the object structure instead of the procedural structure are themes well supported by the object-oriented paradigm. Rumbaugh, Blaha, Premerlani, Eddy and Lorensen define "abstraction" as focusing on the essential, inherent aspects of an entity and ignoring the accidental properties [RUM91]. Use of abstraction during analysis means concentrating on application domain concepts and not making low-level design decisions. "Encapsulation" (information hiding) is achieved by differentiating accessible and inaccessible properties of objects from outside of the object. Details of an object can be changed while its interface remains the same.

The object-oriented paradigm promises improvement in productivity by being a natural match between implementation and problem. It promotes reuse of objects and increases quality by reducing errors and coupling. It provides better maintainability by encapsulation and ease of extensibility by simply adding another object or feature to an existing object.

B.2.2.3.4.2 Object-Oriented Modeling Approaches and UML

Object-oriented modeling languages emerged in the 1970s and different approaches to object-oriented analysis and design have been proposed; in the 1990s, more than 50 different object-oriented methods were available. The confusion caused by different interpretations limited the progress of these methods. Stronger versions of these methods began to appear by late 1990s, including OOSE (Object-Oriented Software Engineering) by Ivar Jacobson, OMT (Object Modeling Technique) by Jim Rumbaugh, and Grady Booch's method. OOSE provided a use-case-oriented approach supporting requirements analysis based on interactions between users and systems. OMT was especially expressive for analysis and information systems while Booch's method was particularly expressive for system partitioning.

The unification of Booch and Rumbaugh resulted in the release of a draft version 0.8 of UML in October 1995. In fall 1995, Jacobson joined the unification process [OMG01]. Table B.3 presents the previous efforts that have influenced the unification [MUL97].

Table B.3: Origins of UML [MUL97]

Origin	Element
Booch	Categories and subsystems
Embley	Singleton classes and composite objects
Fusion	Operation descriptions, message numbering
Gamma et al.	Frameworks, patterns and notes
Harel	State charts
Jacobson	Use Cases
Meyer	Pre- and post-conditions
Odell	Dynamic classification, emphasis on events
OMT	Associations
Shlaer-Mellor	Objects' lifecycles
Wirfs-Brock	Responsibilities and collaborations

The unified methodology is designed to provide guidance to the order of team activities, to direct the task of individual developers and the team as a whole, to specify what artifacts should be developed, and to offer criteria for monitoring and measuring a project's products and activity. Jacobson, Booch and Rumbaugh list the four goals of UML as follows [MUL97]:

1) To represent complete systems using object-oriented concepts.
2) To take into account the scaling issues.
3) To establish an explicit coupling between concepts and implementation.
4) To create a modeling language usable by both human and machines.

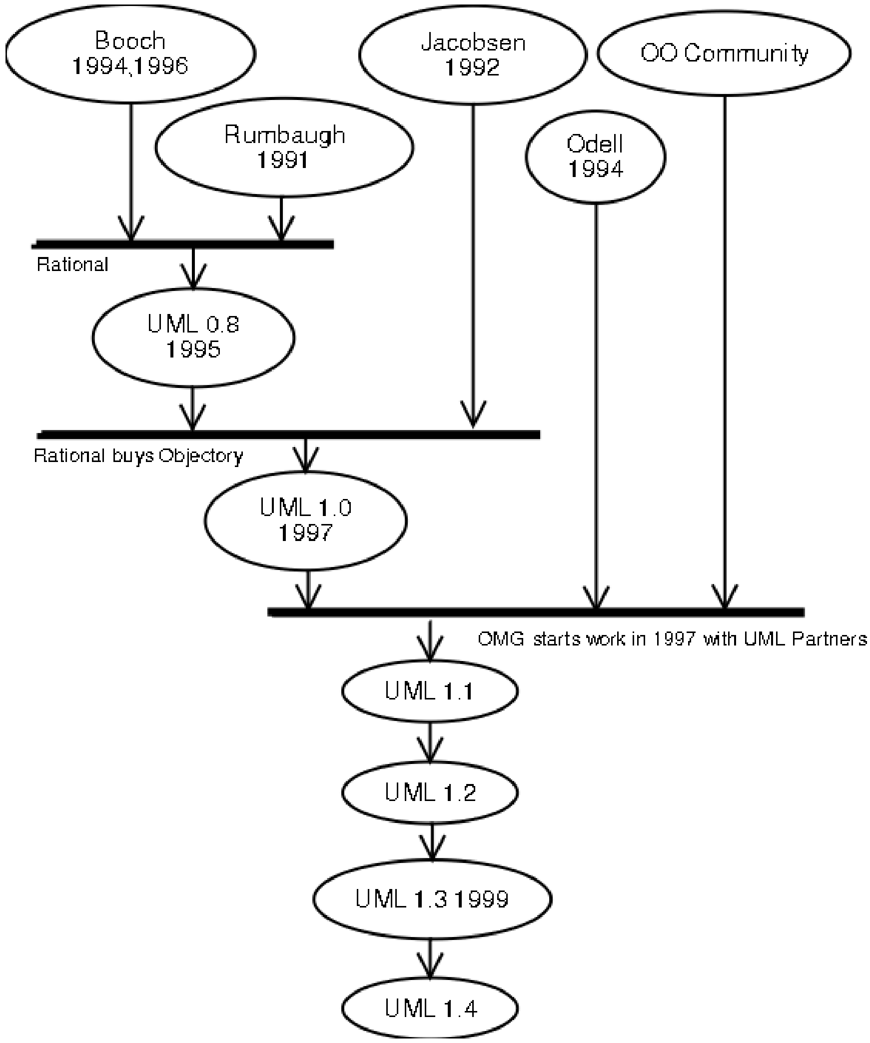


Figure B.16: UML History [VIN02].

The unified development process met the requirements of the software development community with a generic process framework that can be specialized for a variety of software systems, application areas, organizations, competence levels, and project sizes. The distinguishing aspects of UML are the ability to provide a use-case driven, architecture centric, iterative, and incremental design process [JAC99]. With these advantages UML became a widely used standard in the software industry for modeling software. Recently, OMT is about to finish the final specification of UML 2.0, which is the first major revision to the standard since its inception in 1997. The new specification is designed to support a number of model-driven development paradigms, including Model-Driven Architecture (MDA) as defined by the Object Management Group (OMG).

B.2.2.3.4.3 OODBMS

The object-oriented model is one of the most recent database models based on the concept of storing and retrieving objects, which are a collection of data items and the operations, which can be executed on them. The first commercially available object oriented DBMS became available in the mid-1980s. By the early 1990s, there were a range of OODBMSs available from a variety of vendors.

Object-Oriented Database Technology is a search for a method to solve the problems by expressing objects as relations. Some database application domains tend to inherently lend themselves to object-oriented data modeling with large amounts of data, which needs to be stored and manipulated in ways which relational systems were not designed to handle, such as Computer-aided Software Engineering (CASE), Mechanical Computer-Aided Design (MCAD), Electronics Computer-Aided Design (ECAD), Computer-Aided Manufacturing (CAM), Office Document Generation/Control Software, Graphics Packages, Scientific/Medical Applications, Knowledge Base Applications, etc. [CAT91].

Although OODBMSs allow fast navigation through links between objects, flexible locking protocols, rich type set, natural representation of objects, their theory and standards are immature. Relational model is a solid data model mathematically expressible in terms of a relational algebra and tuple calculus based on storing data in central repository for multiple applications to access and manipulate where Object-Oriented model is on the other hand based on preserving the state of an application between execution sessions and does not possess a common data model or theoretical framework.

Unlike the relational model, the OO model does not have a high level language like SQL. This gives the programmer a low level control of the system where he/she can control how data is to be stored and manipulated; however, it is much more difficult for third parties to produce add-on products.

Atkinson et al. published "The Object-Oriented Database System Manifesto" in 1989. This paper opens the debate about the definition of OODBMSs and describes the main features and characteristics that a system must have to qualify as an object-oriented database system [ATK89]. They emphasized the three points characterizing the research at its current stage: (i) the lack of a common data model, (ii) the lack of formal foundations, and (iii) strong experimental activity. To clarify the definition of object-oriented database systems, Atkinson et al. proposed characteristics that such systems should possess in three categories: mandatory, optional and open [ATK89].

B.2.2.3.4.4 OR Mapping

Object-relational mapping is the process of integrating relational models, which only store data and object-oriented model where objects have identity, state, and behavior in addition to data. General concepts and definite rules for object-relational mapping can be defined. Although O/R Mapping process has been automated and there are many commercial tools available, the automatic mapping process may cause errors during data conversion in case of a change in the object schema and it may also reduce the performance of the system. The main problem with OR mapping is "impedance mismatch" meaning that models may not match up precisely. In most cases, relational model is used for entire business – accessed by different applications, where an object model is specially designed and optimized for one application.

Following are the issues to be considered for object-relational mapping:

- Implementing identity;
- Implementing domains; and
- Defining tables and relations.

B.2.2.3.4.4.1 Implementing Identity

There are two types of identity: Existence-based or value-based. In value-based identity, some attributes of the object are combined to obtain a unique value. The designer should ensure the uniqueness of the resulting value referring to the object. An example for this type of identity may be aircraft names. For instance, F-16DFB1992126 value may mean that this aircraft is F-16, D version, with role of fighter-bomber and 126th aircraft manufactured in 1992. Although it works well with RDBMSs or small file applications, value-based identity is difficult to change since it may introduce interdependencies in the application.

Existence-based identity depends on object identifiers (OID) generated by the system guaranteeing uniqueness. The advantages of system-generated OIDs are their smaller and uniform size. However, it may be difficult to generate an OID in a distributed database system. Existence-based identity is usually preferred because of the independency it provides between object identifiers and application data so that both can be changed without affecting each other.

B.2.2.3.4.4.2 Implementing Domains

Domain is a set of values, which can be represented in a field. Language data types are the main classes of domains. The domain for a “byte” data type can be expressed as [0,255], meaning that a byte can store values between and including 0 and 255. Since RDBMS data types are often less rich than language types, methods for domain representation are required.

B.2.2.3.4.4.3 Defining Tables and Relations

In this section, examples for object relational mapping are provided for table and relation definitions including one-to-one, one-to-many, many-to-many, and inheritance relations.

Pilot_OID	Name
1	Hakan
2	Ramazan
3	Hüseyin

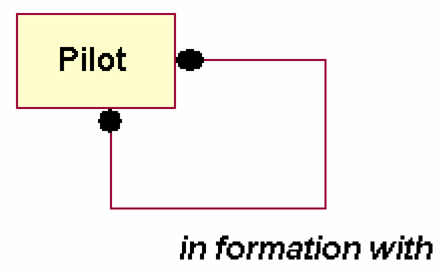
AC_OID	Pilot_OID	AC_Name
1	1	CN-235
2	2	F-16
3	3	F-4
2	3	F-16

Figure B.17: One-to-Many/One-to-One Relations.

Pilot_OID	Name
1	Hakan
2	Ramazan
3	Hüseyin

Tgt_OID	Pilot_OID	Tgt_Name
1	1	AFB_1
2	1	AFB_2
2	2	AFB_2
3	3	RADAR_1

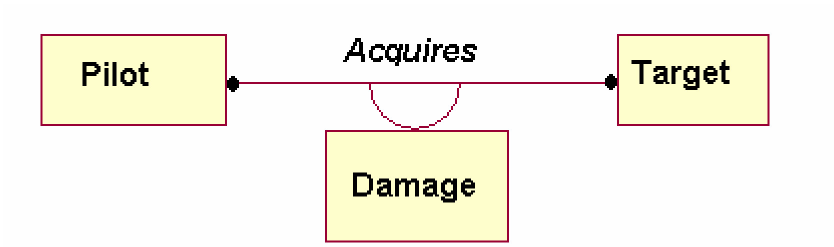
Figure B.18: Many-to-Many Relation-1.



Pilot_OID	Name
1	Hakan
2	Ramazan
3	Hüseyin

Pilot_OID	Pilot_OID
1	2
2	3

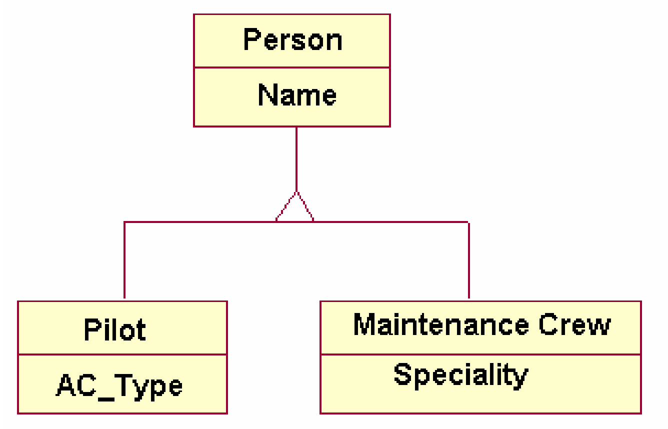
Figure B.19: Many-to-Many Relation-2.



Pilot_OID	Name
1	Hakan
2	Ramazan
3	Hüseyin

Tgt_OID	Pilot_OID	Damage
1	1	100%
2	1	50%
2	2	50%
3	3	75%

Figure B.20: Association Classes.



Person_OID	Name
1	Hakan
2	Ramazan
3	Mehmet

Person_OID	AC_Type
1	CN-235
2	F-16

Person_OID	Speciality
3	Avionics

Figure B.21: Inheritance – Separating Superclass and Subclass Tables.

Person_OID	Name	AC_Type
1	Hakan	CN-235
2	Ramazan	F-16

Person_OID	Name	Speciality
3	Mehmet	Avionics

Figure B.22: Inheritance – Pushing Attributes Down to Subclasses.

Person_OID	Name	AC_Type	Speciality
1	Hakan	CN-235	
2	Ramazan	F-16	
3	Mehmet		Avionics

Figure B.23: Inheritance – Pushing Attributes Up to Superclass.

B.2.2.3.5 Object-Relational DBMS (ORDBMS)

The Object Relational Model is a relatively recent development based on the researches analyzing the possibility of storing RDBMSs objects in the fields of a record. ORDBMSs are extended versions of RDBMSs with the ability to explicitly define new types, to store complex type values and to allow visibility into complex values. One of the reasons for ORDBMSs research was that relational model couldn't cope effectively with the new types of data that came with increasing use of object oriented programming languages. These new types include audio, video and image files and user-defined types. The second reason was heavy investment in RDBMS technology resulting in thousands of RDBMS tools and RDBMS-skilled developers, and associated risks and cost of switching to OODBMS.

B.2.2.3.6 Other DBMS (ORDBMS)

Although relational and object-oriented models meet most of the business expectations, there are other models designed to solve specific problems. These include but not limited to geo-spatial databases, real-time databases, deductive databases and multimedia databases.

Real-time database systems are specially designed to satisfy application-timing constraints by simultaneously enforcing data integrity constraints. Although a mature body of research has been done for a decade, this research has almost exclusively been devoted to extending traditional transaction processing issues such as resource scheduling policies, concurrency control, memory management, etc., to the real-time environment.

Geospatial database systems manage data spatially referenced to the Earth. These DBMSs have to deal with a great amount of data. Thus, as the user browse through a 2D or 3D map, geospatial database management system should be able to keep up with the data access that has a special pattern. For more information, refer to technology survey document on geospatial information systems [KAR03] prepared within analysis phase of NATO SAS-045 project.

B.3 MAJOR DBMS PROVIDERS

B.3.1 RDBMS

Table B.4: RDBMS Providers

Provider	Web Page
CINCOM	www.cincom.com
Computer Associates International	www.cai.com
IBM DB2	www.software.ibm.com
Informix	www.informix.com
jBASE Int.	www.jbase.com
Microsoft Corporation	www.microsoft.com
MySQL (Open Source)	www.mysql.com
Oracle	www.oracle.com
PostgreSQL (Open Source)	www.postgresql.org
Sybase	www.sybase.com
Via Systems	www.via.com

B.3.2 OODBMS

Table B.5: OODBMS Providers

Provider	Web Page
Computer Associates International	www.cai.com
eXcelon Corporation	www.exln.com
GemStone Systems	www.gemstone.com
Objectivity Incorporated	www.objectivity.com
POET Software Corporation	www.poet.com
Progress Software	www.objectstore.net
Versant Co.	www.versant.com

B.3.3 ORDBMS

Table B.6: ORDBMS Providers

Provider	Web Page
Cloudscape (JBMS)	www.cloudscape.com
IBM (DB2 v.3)	www.software.ibm.com/data/db2
Informix	www.informix.com
Microsoft	www.microsoft.com
Oracle (Oracle8)	www.oracle.com
Sybase	www.sybase.com

B.3.4 OODBMS Comparison

Table B.7 illustrates a sample comparison between vendors of OODBMSs. The criteria in this table may be referred for further vendor selection analysis.

Table B.7: Major OODBMS Vendors Comparison [MAT01]

	Objectivity	ObjectStore	POET
Database Size	Very Large: Petabyte	Large: Terabyte	Medium: Gigabyte
Architecture	Distributed	Client-Server	Embedded, Client-Server
Cache Strategy	Client-side caching	Client-side caching	Client+Server Caching, Server-side queries possible
Java Persistent Object Specification Mechanism	Inheritance or implementation of interface	Postprocessor	Preprocessor + Schema file
Java Persistence Mechanism	Named objects + reachability	Named objects (roots) + reachability	Named objects + reachability
Associations Between Objects	Special Purpose Structures	Normal Java References/Containers	Normal Java References/Containers
Transactions	Long, checkpointed, nested	Long, checkpointed, some versions nested	Long, checkpointed, nested

B.3.5 Automatic Object-Relational Mapping Tools

Table B.8: O/R Mapping Tools [MAT01]

PROVIDER	SOFTWARE
2Link Consulting	DbGen 1.0
Ardent Software, Inc.	Java Relational Binding 2.0
	C++ Relational Binding 1.0
Objectmatter, Inc.	Business Sight Framework 1.0
Persistence Software, Inc.	PowerTier 2.1
POET Software, Inc.	SQL Object Factory 5.1
Software Tree, Inc.	JDX 1.0
Sun Microsystems	JavaBlend 1.0
The Object People, Inc.	TopLink for Java 1.0
	TopLink for Smalltalk 4.02
Watershed Technologies, Inc.	Relational Object Framework 1.1

B.4 CONCLUSION

This document presented the survey on data management issues to be considered for NATO SAS-045 RTG on "Computer Based Decision Support Tool for Helicopter Mission Planning in Disaster Relief and Military Operations." The topics covered include data management approaches, data interaction strategies, DBMS paradigms and DBMS providers. This research concludes that following criteria may be the basis for designing data management architecture for NATO SAS-045:

Performance and Accessibility: Data management should address the time requirements of decision support algorithm to promote its robustness. Thus desired system to be proposed by NATO SAS-045 should respond quickly to an emergency situation.

Interoperability with other NATO models and repositories: Based on another technology survey within analysis phase of NATO SAS-045 [SMI03], most common data repositories in NATO organizations including ICC & ACC (Integrated Command and Control & Air Command and Control System) and ADAMS (Allied Deployment and Movement System), TOPFAS (Tool for Operational Planning, Force Activation and Simulation) have been implemented as relational models. In order to generate practical and flexible plans for missions supported by helicopters during a crisis situation, the decision support system should have rapid access to reliable information in a standard format. The system should also be able to distribute its results in a standard format to the related users.

Distribution and Crash Recovery: In disaster-relief operations it's probable that some parts of an information network might be damaged or inaccessible. The data management architecture to be proposed should consider data distribution and crash recovery.

B.5 REFERENCES

[MAT01] Mathias, K.S., Class Notes, Air Force Institute of Technology Dayton/OH, 2001.

[BOO99] Booch, Rumbaugh and Jacobson, The UML Modeling Language User Guide, Addison-Wesley, 1999.

[BRE02] Bredemeyer Consulting White Paper, Software Architecture: Central Concerns, Key Decisions, http://www.bredemeyer.com, 2002.

[CER03] Web page by Database Group of IT division of CERN (European Organisation for Nuclear Research), http://wwwdb.web.cern.ch

[BIB03] Information Technology for Libraries, http://www.biblio-tech.com

[RUM91] Rumbaugh, J., Blaha, M., Premerlani, W., Eddy, F. and Lorensen, W., *Object-Oriented Modeling and Design*, Prentice Hall, 1991, USA.

[STE99] Stevens, P. and Pooley, R., *Using UML*, Addison Wesley Longman, 1999, Massachusetts.

[MUL97] Muller, P.A., *Instant UML*, Wrox Press Ltd, 1997, UK.

[OMG01] Object Management Group (OMG), *UML 1.4 Specification*, 2001, Online Document, http://www.omg.org/technology/documents/formal/uml.htm

[JAC99] Jacobson, I., Booch, G. and Rumbaugh, J., "The Unified Process," IEEE Software, Vol. 16, pp. 96-102, May/June '99, USA.

[CAT91] Catell, R., Object Data Management: Object-Oriented and Extended Relational Database Systems, Addison-Wesley, 1991.

[ATK89] Atkinson, M., Bancilhon, F., DeWitt, D., Dittrich, K., Maier, D. and Zdonik, S., The Object-Oriented Database System Manifesto, 1989.

[OPE03] Open Database Connectivity Standard, http://www.openlinksw.com

[CAS99] Casmira, J., A DBMS Comparison: RDBMS vs. ORDBMS vs. OODBMS, http://www.cs.colorado.edu, 1999.

[FUS97] Fusel, M.L., Foundations of Object Relational Mapping, 1997.

[VIN02] Vinciguerra, R.L., www.vinci.org/uml/history.html, 2002.

[SUN03] Sun Microsystems web page, http://java.sun.com, 2003.

[SMI03] Smit, M.C., Summary of research on models, data and knowledge management repositories and planning process in NATO organisations, Netherlands (Technology research document prepared within analysis phase of NATO SAS-045 project.)

[FON03] Fontaine, M., Decision Support Tool Technologies, France (Technology research document prepared within analysis phase of NATO SAS-045 project.)

[KAR03] Karasakal, O., Research Report on Geographical Information Systems, Digital Maps and Automatic Digital Data Capture Decision Support Tool Technologies, Turkey (Technology research document prepared within analysis phase of NATO SAS-045 project.)

NATO RTG SAS-045
on
Computer Based Decision Support Tool for Helicopter Mission Planning in Disaster Relief and Military Operations

Annex C – TECHNICAL REPORT 3: OVERVIEW OF GEOGRAPHICAL INFORMATION SYSTEMS, DIGITAL MAPS AND MISSION DATA COMPILATION

Orhan Karasakal
Dz.K.K.lığı Bakanlıklar, 06100 Ankara, Turkey
Tel: +90 312 403 2446; Fax: +90 312 417 3065

E-mail: okarasa@dzkk.tsk.mil.tr

Marcel Smit
TNO-FEL, P.O. Box 96864, 2509 JG The Hague, The Netherlands
Tel: +31 70 374 0175; Fax: +31 70 374 0642

E-mail: M.C.Smit@fel.tno.nl

C.1 INTRODUCTION

This document is prepared to fulfill the requirements of the Programme of Work (POW) of NATO SAS-045 RTG on “Computer Based Decision Support Tool for Helicopter Mission Planning in Disaster Relief and Military Operations”.

The POW dictates to provide an a-priori analysis of Geographical Information Systems (GIS), digital maps, and mission data compilation that are anticipated to support the development of NATO SAS-045 project. As it is stated in the Terms of Reference (TOR) of the afore-mentioned project, the main goal of the research is to provide the basis for developing a generic and flexible decision support tool for effective management of helicopter missions by conducting the problem analysis, investigating the concept of solutions and determining relevant technical requirements.

It is foreseen that different models, technologies and solution approaches will be utilized concurrently, and integration and interfacing will pose itself as an important technical issue within the scope of maintaining interoperability and standardization in NATO practices.

Thus, the POW states that during the Analysis Phase of the project technology surveys should be carried out on modeling, computer and software engineering and data collection technologies; geographical information systems, digital maps, mission data compilation systems; model, data, and knowledge management repositories in NATO nations. Then, the technology mapping and capability matrix can be developed using the identified current needs and capability gaps.

In this document, it is intended to provide a broad overview that will guide any potential research dealing with the design and development of a generic decision support tool within a similar NATO context, not limited to helicopter operations.

The aim of this document is not to cover the subject matter in detail, but rather to give a general idea of techniques and products. In this report, we provide information on GIS, digital maps, mission data compilation systems in order to determine the design requirements for the helicopter mission planning decision support system.

In the next section, we provide information on GIS. Digital maps are discussed in Section 3. In Section 4, we summarize the mission data compilation systems in NATO and NATO nations. This report ends with a conclusion section.

C.2 GEOGRAPHICAL INFORMATION SYSTEMS (GIS)

Geographical Information System (GIS) is a “computer system for capturing, managing, integrating, manipulating, analysing and displaying data which is spatially referenced to the Earth.” (McDonnell and Kemp, 1995). GIS is an information system that is composed of hardware, software, data and personnel. The current generation of GIS products support data integration and manipulation, query, thematic and overlay analysis, and visualization in two and three dimensions. Geographic information technology has emerged as an important operations research tool for exploiting increasingly available amounts of valuable geographic data in a form easily comprehended by analysts and decision makers (Hanigan, 1989).

What distinguishes GIS from other forms of information systems, such as databases and spreadsheets, is that GIS deals with spatial information. GIS has the capability to relate layers of data for the same points in space (see Figure C.1), combining, analyzing and, finally, mapping out the results. Spatial information uses location, within a coordinate system, as its reference base. The most common representation of spatial information is a map on which the location of any point could be given using latitude and longitude, or other grid references.

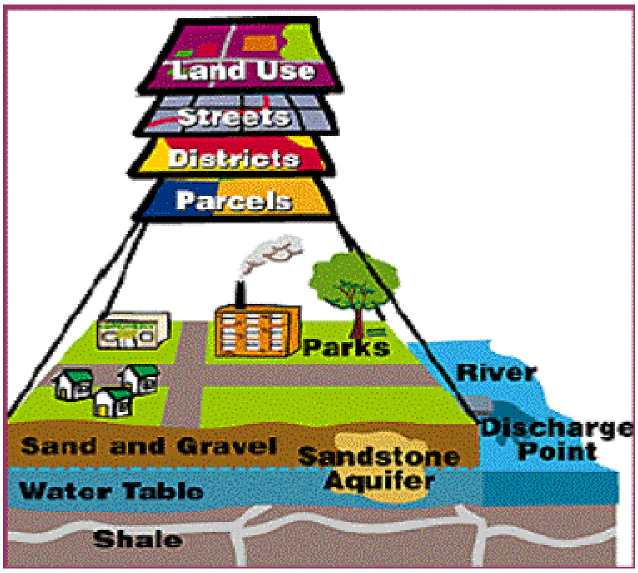


Figure C.1: Layers of Data in GIS.

Some applications of GIS are obvious, for example water supply companies use GIS as a spatial database of pipes and manholes; local governments can use GIS to manage and update property boundaries, emergency operations and environmental resources. GIS may also be used to map out the provision of services, such as health care and primary education, taking into account population distribution and access to facilities. Increasingly, GIS is being used to assist businesses in identifying their potential markets and maintaining a spatial database of their customers (see Figure C.2).

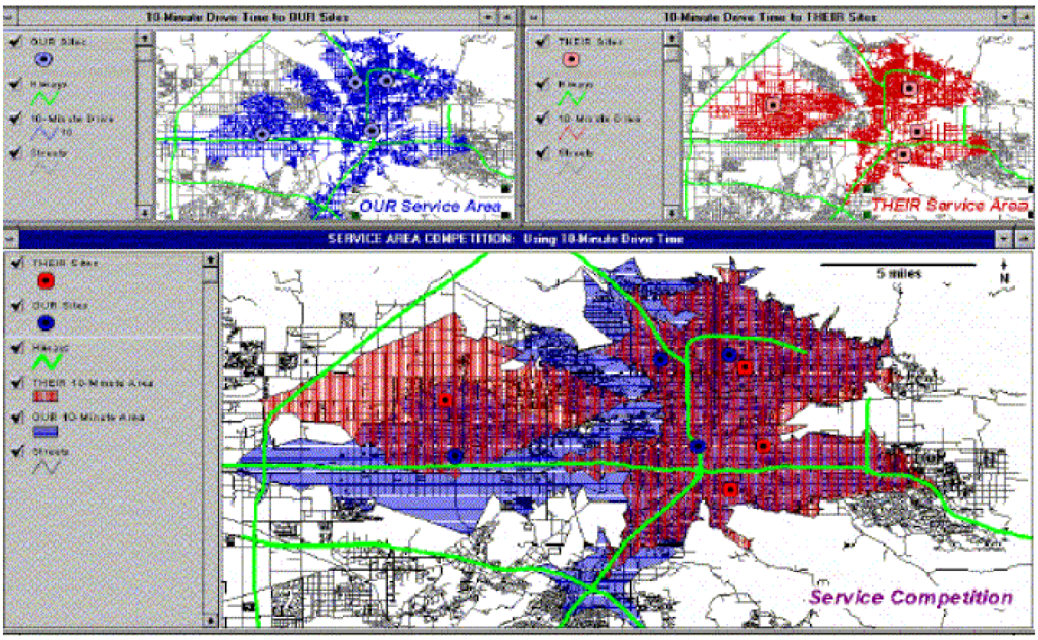


Figure C.2: Example of a Customer Relationship Management Application.

Geospatial information plays a key role in defence. Knowing what is, where is the key to mastering the battlefield, the installations, and deployed areas that support the warfighter. Military leaders must have a common tool with rapid access to relevant and accurate geospatial information for mission-critical operations. To provide this geospatial information, the decision maker's ability to use, manage, analyze, and disseminate information is a vital component for command planning and operations. Broad ranges of challenges require the deployment of flexible and scalable technology to assist the decision makers.

GIS systems can visualize, manage, analyze and present geospatial data. Intelligent maps can be created by relating organizational databases with digital maps. Thus enabling quick and efficient decision making and instant data integration between the organizations.

GIS database is created via combining data from different sources. Scanned and digitized maps, data in tables, correct location information, information from land study and remote sensing and air photos provide data for GIS.

Spatial data may be represented in GIS in one (or both) of the two following formats (see Figure C.3):

- Vector model, as geometric objects: points, lines, polygons.
- Raster model, as image files composed of grid-cells known as pixels.

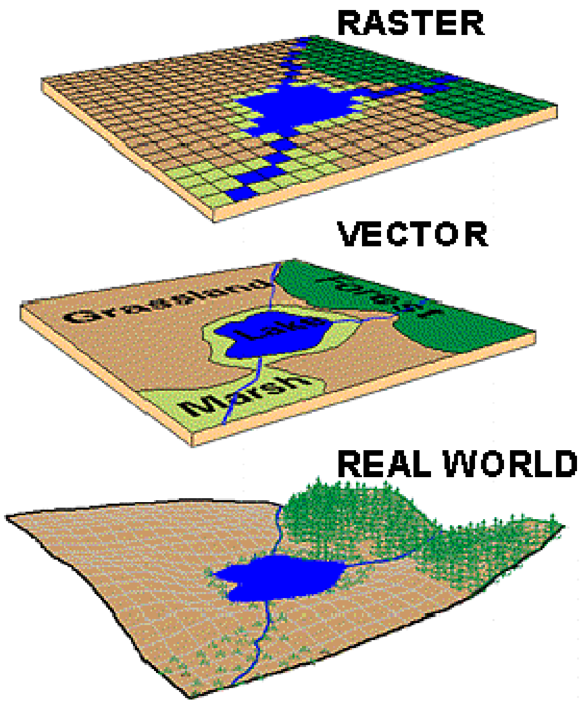


Figure C.3: Spatial Data Representation in GIS.

Geographic data have four major components. These are geographic position (location of geographic data recorded in a coordinate system), attributes (characteristics of geographic features such as "what it is"), relationships (the relationships between the geographic features), and time (geographic data may be referenced to a point in time or a period of time).

Relationships between data can be put into two categories: Spatial relationships (either topological relationships that are independent on coordinates (e.g. next to, inside) or metric relationships that are dependent on coordinates (e.g. 10 km away)) and non-spatial relationships (e.g. owner of, builder of).

Raster data can be represented using three different encoding schemes. Those are traditional raster encoding, run-length raster encoding, and quad tree raster encoding.

In vector representation, geographical features are organized as points, lines and areas called polygons. There are two different vector data models. In spaghetti model, the paper map is translated line-for-line into a list of x,y coordinates and in topological model, spatial relationships among entities are retained by explicitly recording adjacency information.

Comparison of vector and raster data models:

- Vector model is a more complex data structures and generally more compact;
- Raster model provides less explicit topological relationships;
- Some operations, such as overlay, are easier using vector data;
- Some operations, such as finding a shortest path, are easier using vector data;
- Boundaries are more accurately defined in vector model;
- Some input data, such as satellite imageries, are raster based;
- Some output devices, such as laser printers, are raster based; and
- Some output devices, such as pen plotters, are vector based.

Each data model has particular strengths and weaknesses, and the type of model used is to be determined by the nature of the work being undertaken and the data available.

The main advantage of the vector data format is that it allows precise representation of points, boundaries, and linear features. This makes it particularly useful for analysis tasks that require accurate positioning, for example in engineering or cadastral boundary databases. It is also possible in a vector-based GIS to define the spatial relationship (i.e. the connectivity and adjacency) between coverage features. This aspect of GIS is known as topology, and is important for such purposes as network analysis (for example to find an optimal path between two nodes in a complex transport network). By contrast, raster-based GIS defines the position of features in terms of x,y coordinates where topological associations are more difficult to represent. However, the main disadvantage of vector data is that the boundaries of the resulting map polygons are discrete (enclosed by well-defined boundary lines), whereas in reality the map polygons may represent continuous gradation or gradual change, as in soil maps.

Raster Data Models use a raster matrix (a grid of image cells) to represent information. The resolution (visual definition) of the raster depends on its pixel (cell) size. In other words, pixel resolution represents the size of the ground area covered by each pixel in the image. The smaller the cell size, the higher the resolution. The raster data model is, therefore, good for representing indistinct boundaries, such as thematic information on soil types, soil moisture, vegetation, ground temperatures, and so on. Furthermore, as reconnaissance satellites and aerial surveys use raster-based scanners, the information (i.e. scanned images) can be directly incorporated into GIS programs capable of working with raster data. However, the higher the grid resolution, the larger the data file is going to be. This is the main limitation of raster based GIS.

The question of which data model to use in GIS depends on the nature and objective of the GIS project. Primarily the model type will depend on the nature of the data. Issues of concern are the volume of the data generated, ease of analysis and accuracy. Generally, vector data sets are economical in terms of file size, and have a high level of positional precision, but are relatively difficult to use in mathematical computations. On the other hand, grid data sets tend to take up more file space and have a coarser resolution, but are easier to work with mathematically.

Standard hardware for GIS includes computers, scanners, digitizers, printers and line plotters. Hardware such as remote sensing units, global positioning systems (GPS) and software may be included in GIS system depending on the application.

Architecture of GIS may be client/server (distributed and simultaneous operation) or stand-alone (independent application on a PC).

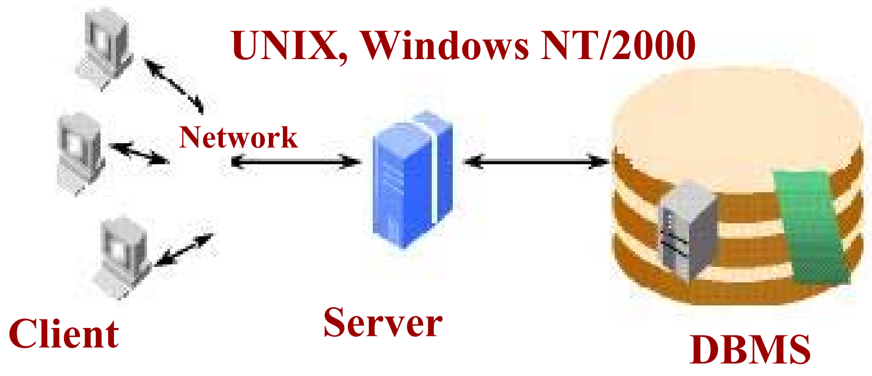


Figure C.4: Client/Server Architecture for GIS.

There are more than 300 GIS software in market. Several of those are:

- ArcInfo (www.esri.com);
- MapInfo (www.mapinfo.com);
- Idrisi (www.clarklabs.org);
- GeoMedia (www.intergraph.com/gis/geomedia);
- IlogViews (www.ilog.com); and
- Autodesk (www.autodesk.com).

GIS and data base management system (DBMS), which is a complex set of programs that control the organization, storage, retrieval and security of data for users can be integrated. Storing, querying and replication of geospatial data, and creating intelligent information by integrating data at the DBMS and geospatial data are important features of such integration. IBM Informix Spatial DataBlade, IBM DB2 Spatial Extender and Oracle Spatial Database Engine are examples of those kinds of software.

GIS applications require decisions on; required data types, database structure (relation between data), base maps and their scales and required expertise on different areas.

C.3 DIGITAL MAPS

Representation of the terrain is very important. Maps were very much used in military environments. The last decade digital maps become more used also in a military environment. Geographical data exists in many appearances. The complete world is now available on digital maps. In determining geographical data requirements the following aspects are of importance:

- Format of the file; and
- Co-ordinate reference system.

Format: The format is the way the digital data is captured. There are many formats such as usual formats of pictures like “BMP”, “TIF” and “JPG”, DLMS (consisting of DTED (terrain information) and DFAD (features)), Modsaf CTDB, INDIA/MARS, VMAP1 in combination with Vector Product Format (VPF), MilGeo-PCMAP, Geotif in combination with Shapefiles and DXF.

Co-ordinate reference system: The coordinate reference system is the system that uses coordinates to establish the position on the earth. Each system uses a projection method in order to transform a position on the globe to a 2D position. There are also many co-ordination systems that use different projections methods of the earth, e.g. UTM, UTM MGRS, WGS84, Gauss Kruger, Lambert, Geographic Coordinate System (Lat/Lon), Rijksdriehoeks (Dutch).

Digital maps have specific formats and coordinate systems in which they have been produced. Each format and co-ordinate system has some features and drawbacks depending on the application. In the absence of a universal standard, conversion between formats and coordinate systems are problematic.

In the USA the National Mapping Program (NMP) has introduced a number of standards (see also http://mapping.usgs.gov/standards/). The standards are essential for efficient sharing of products and to provide information about geospatial data. The NMP produces the following five basic types of digital geospatial data:

- **Digital Elevation Model Standards:** Standards, which define a digital file consisting of terrain elevations for ground positions at regularly spaced intervals. A Digital Elevation Model is often referred to as a DEM.
- **Digital Line Graph Standards:** Standards, which define digital vector files containing line data, such as roads or streams, digitized from quadrangle maps. A Digital Line Graph is often referred to as a DLG. The standards cover both DLG-3 and DLG-F series.
- **Digital Orthophoto Standards:** Standards, which define a digital image of an aerial photograph in which displacements caused by the camera and the terrain have been removed. A Digital Orthophoto is often referred to as a DOQ. A DOQ combines the image characteristics of a photograph with the geometric qualities of a map.
- **Digital Raster Graphic Standards:** Standards, which define a digital image, which is scanned from a USGS quadrangle map. A Digital Raster Graphic is often referred to as a DRG. A DRG is georeferenced to the surface of the Earth, inside the map neat line.
- **National Hydrography Dataset Standards:** The USGS and the Environmental Protection Agency (EPA) are jointly preparing standards for the National Hydrography Dataset (NHD) and the National Hydrography Dataset – High Resolution (NHD – HR). Both the NHD and NHD – HR are digital vector files of cataloging units which uniquely identify and interconnect the stream segments or “reaches” that comprise the country’s surface water drainage system.

Other geospatial standards and specifications can be found at the following website: http:/164.214.2.59/publications/specs/index.html.

NATO also tries to standardize digital maps used in GIS applications as much as possible in order achieve interoperability between different national systems. The list of NATO standards on collecting, using and exchanging geospatial information are as follows:

STANAG 2215-Evaluation of Land Maps, Aeronautical Charts and Digital Topographic Data (Edition 6): The aim of this STANAG is to enable producers of geographic material to standardize the system of evaluation of land maps, aeronautical charts and digital topographic data to be used by NATO armed forces.

STANAG 3809-Digital Terrain Elevation Data (DTED) Exchange Format (Edition 4): Participating nations agree to use US Performance Specification 89020B as a manual for defining the requirements within a DTED database. DTED supports various weapon and training systems. The purpose of this specification is to assure uniformity of treatment among all mapping and charting elements engaged in a coordinated production and maintenance program for this product.

STANAG 4387-ARC Standard Raster Product (ASRP)-AGeoP-5.

STANAG 7072-Vector Map (VMap) Level 0 (Edition 2): Participating nations of this STANAG agree to use US MIL-PRF-89039 Vector Map (VMap) Level 0 as a manual for military planners at the operational and, to a limited extent the tactical level of operations. This product will support Geographic Information Systems (GIS) applications. VMap0 data is suitable for use by both ground and air commanders.

STANAG 7074-Digital Geographic Information Exchange Standard (DIGEST): The aim of this agreement is to provide interoperability and compatibility among NATO systems and users for the exchange and provision of digital geographical information.

STANAG 7098-Compressed ARC Digitized Raster Graphics (CADRG): CADRG is a general-purpose product, comprising computer readable digital map and chart images. CADRG files are physically formatted within a National Imagery Transmission Format (NITF). CADRG uses WGS-84 coordinates as coordinate reference system.

STANAG 7108-ARC Digitized Raster Graphics (ADRG) (Edition 1): The aim of this agreement is to standardize ADRG produced by participating nations. Participating nations agree to produce and use ADRG products in accordance with the specifications given in this STANAG. The ARC System is a special grid system covering the entire ellipsoid of the World Geodetic System 1984. It provides a rectangular coordinate system based on 18 latitudinal zones. These allow a relative coordinate system to be used with individual raster images.

STANAG 7136 Identification of Land Maps, Aeronautical Charts, Digital Geographic Datasets and Media Containing Datasets (Edition 1): The aim of this Agreement is to establish a standard method of uniquely identifying land maps, aeronautical charts, digital geographic datasets and media containing geographic datasets, which are produced and exchanged by NATO members. Hydrographic products are excluded from this STANAG.

STANAG 7173: The Arc System (The Equal Arc-Second Raster Chart/Map System) (Edition 1): The aim of this agreement is to enable a common understanding of the Equal Arc-Second Raster Chart/Map System amongst nations exchanging digital geographical information.

C.4 MISSION DATA COMPILATION

In this section, we provide brief information on automatic data capture, remote sensing and tactical data link systems.

C.4.1 Automatic Data Capture

The traditional way of entering planning and manufacturing data into a computer system has involved manually keying it in after the data has been gathered on sheets of paper. Studies show that with this technique there is likely to be 1 error for every 300 characters entered (Lindau and Lumsden, 1999). The traditional way is error prone, non-real time, and risky in emergency situations. Whereas, automatic data capture techniques make it possible to enter a stream of data automatically in a single operation.

By expressing the data, e.g. part number, with a machine-readable code; this information can be entered into the computer system with an automatic code-reading device. The decoded data can either be transmitted directly to an attached computer, be stored locally for later transmission, or interact with a resident application program in the reading device. Data entered by such techniques has a lower error rate than manual methods, 1 error for every 3 million characters entered (Lindau and Lumsden, 1999) and data is available in real-time. Vehicle monitoring systems using GPS and SMS cellular phone technology, and meteorological sensors are two examples of automatic data capture systems.

C.4.2 Remote Sensing

Remote sensing involves the use of sensors to "capture" the spectral and spatial relations of objects and materials observable at a distance – typically from above them. (http://rst.gsfc.nasa.gov) An aerial photograph is a common example of a remotely sensed product. Determining the status of a growing crop, defining urban patterns, delineating the extent of flooding, recognizing rock types, pinpointing areas of deforestation are typical application areas.

Remote sensing data, from vegetation and climatic data to outlines of manmade structures, can form much of the information that is entered into a GIS. The technologies of remote sensing and GIS are distinct, but complimentary. Different equipment and technical skills are needed for each, but in both cases the user must have a string grasp on the information that is collected, be it forestry, geology, man-made structures, etc. Remote sensing technology focuses on sensor systems and image processing, whereas GIS requires expertise with the principles of spatial analysis, map projections and databases (Aronof, 1995).

C.4.3 Tactical Data Links

Tactical Data Links (TDLs) are the means of providing real time encrypted communications to achieve total situational awareness through use of advanced information technology. TDLs involve transmissions of bit-oriented digital information, which are exchanged via data links. Within NATO, ADatP-33 provides the procedures for operators who employ tactical data links, for planning, initializing, controlling, and terminating the exchange of real-time and near real-time tactical data between tactical data systems, and defines responsibilities for these functions.

TDL systems used in NATO and NATO nations are briefly explained below. (See http://www.fas.org/irp/program/disseminate/tadil.htm for further information on TDLs.)

C.4.3.1 Link-1

Link 1 is a duplex digital data link primarily used by NATO's Air Defence Ground Environment (NADGE). It was designed in the late 1950s to cater for point-to-point data communication. Link 1 mainly provides for exchange of air surveillance data between Control and Reporting Centres (CRCs) and Combined Air Operation Centers (CAOCs)/Sector Operation Centers (SOCs) and has a data rate of 1200/2400 bit per second (bps). It is not crypto secure and has a message set (S-series) limited to air surveillance and link management data.

C.4.3.2 Link-4A

Link 4 is a non-secure data link used for providing vector commands to fighters. It is a netted, time division link operating in the UHF band. There are 2 separate "Link 4s": Link 4A and Link 4C. Link-4A provides digital surface-to-air, air-to-surface, and air-to-air tactical communications. Link 4C is a fighter-to-fighter data link which is intended to complement Link 4A although the two links do not communicate directly with each other.

C.4.3.3 Link-11

Link-11 employs netted communication techniques and a standard message format for exchanging digital information among airborne, land-based and shipboard tactical data systems. Link-11 data communications must be capable of operation in either the high-frequency (HF) or ultra-high-frequency (UHF) bands.

C.4.3.4 Link-14

Link 14 is a broadcast HF teletype link for maritime units designed to transfer surveillance information from ships with a tactical data processing capability to non-tactical data processing ships. The Link can be either HF, VHF or UHF dependent on unit-communication fits.

C.4.3.5 Link-16

Link-16 is a relatively new tactical data link, which is being employed by several nations of NATO and Japan. Link-16 does not significantly change the basic concepts of tactical data link information exchange supported for many years by Link-11 and Link-4A. Rather, Link-16 provides certain technical and operational improvements to existing tactical data link capabilities and provides some data exchange elements, which the other data links lack. It provides significant improvements as well, such as jam resistance; improved security; increased data rate (throughput); increased amounts/granularity of information exchange; reduced data terminal size, which allows installation in fighter and attack aircraft; digitized, jam-resistant, secure voice capability; relative navigation; precise participant location and identification and increased numbers of participants.

C.4.3.6 Link-22

Link 22 is the next-generation NATO Tactical Data Link, and is also referred to as the NATO Improved Link Eleven (NILE). Link 22 is a multi-national development program that will produce a “J” series message standard in Time Domain Multiple Access architecture over extended ranges.

C.5 CONCLUSION

A generic decision support system (DSS) is expected to interact with the analyst and the decision maker for exploiting different course of actions and present the current status and the results of the actions in a form easily comprehended by analysts and decision makers. Geographic information systems and mission data compilation systems can help to fulfill this requirement.

C.6 REFERENCES

Aronof, S., 1995. “Geographic Information Systems: A Management Perspective”, Ottawa, WDL Publications. p. 294.

Hanigan, F.L., 1989. “GIS Recognized As Valuable Tool For Decision Makers”, The GIS Forum 1.4.

Lindau, R. and Lumsden, K., 1999. Int. Journal of Production Economics 59, pp. 159-167.

McDonnell R. and Kemp, K., 1995. “International GIS Dictionary”, Cambridge GeoInformation International.

NATO RTG SAS-045
on
Computer Based Decision Support Tool for Helicopter Mission Planning in Disaster Relief and Military Operations

Annex D – TECHNICAL REPORT 4: OVERVIEW OF RESEARCH ON MODELS, DATA AND KNOWLEDGE MANAGEMENT REPOSITORIES AND PLANNING PROCESS IN NATO ORGANISATIONS

Marcel Smit
TNO-FEL, P.O. Box 96864, 2509 JG The Hague, The Netherlands
Tel: +31 70 374 0175; Fax: +31 70 374 0642

Email: M.C.Smit@fel.tno.nl

D.1 INTRODUCTION

This document is prepared to fulfil the requirements of the Programme of Work (POW) of NATO SAS-045 RTG on "Computer Based Decision Support Tool for Helicopter Mission Planning in Disaster Relief and Military Operations".

The POW dictates to provide an a-priori analysis of the model, data, and knowledge management repositories (which are available in NATO nations and which can be made available for this study) and that are anticipated to support the development of NATO SAS-045 project. As it is stated in the Terms of Reference (TOR) of the aforementioned project, the main goal of the research is to provide the basis for developing a generic and flexible decision support tool for effective management of helicopter missions by conducting the problem analysis, investigating the concept of solutions and determining relevant technical requirements.

It is foreseen that different models, technologies and solution approaches will be utilised concurrently, and integration and interfacing will pose itself as an important technical issue within the scope of maintaining interoperability and standardisation in NATO practices.

Thus, the POW states that during the Analysis Phase of the project technology surveys should be carried out on modelling, computer and software engineering and data collection technologies; geographical information systems, digital maps, mission data compilation systems; model, data, and knowledge management repositories in NATO nations. Then, the technology mapping and capability matrix can be developed using the identified current needs and capability gaps.

In this document, it is intended to provide a broad overview that will guide any potential research dealing with the design and development of a generic decision support tool within a similar NATO context, not limited to helicopter operations.

In this document an overview is given of the models that are being used to support the planning process of operations and helicopter operations more specific. The planning models described in this report refer only to planning models that are being used in a military environment. SAS-045 also looks at planning of helicopter operations in disaster relief operations. However, in disaster relief operations military helicopters will be used and in this case planning of the helicopters will always be performed by military personnel using military planning models. Another point to address is that in this research no helicopter planning models have been found that are being used by civil organisations in disaster relief operations. So it is sufficient to describe the helicopter planning models that are being used in a military environment.

First the planning process inside NATO is described (Chapter 2). In Chapter 3 an overview is given of the models that exist inside NATO to support the planning process of operations. Then in Chapter 4 an overview is given of the models that exist in the nations. It is very well possible that this overview is not complete. Only the models known by one the members of SAS-045 are described here. This report will be closed with some conclusions (Chapter 5).

D.2 NATO PLANNING PROCESS

Operational planning in NATO (see also ref. [1]) can be conducted under a wide range of conditions, from routine exercises to immediate reaction to an attack on NATO territory. The main categories of planning situations are:

- The response to an emerging crisis situation that NATO might become involved in- typically a peace support operation where NATO become involved as the result of a UN resolution/mandate and request for intervention; and
- Prudent military planning for potential future operations that are not linked to any expressed threat or an actual crisis situation, but which nevertheless require advance planning to ensure NATO's ability to respond should it be called upon to do so.

The ultimate responsibility for initiating planning under any of these headings rests with NATO's political leadership – the North Atlantic Council (NAC) although military commanders are, of course, expected to conduct prudent military planning within their terms of reference. In either case, the planning procedures that NATO has adopted are the same although the urgency and time pressure will of course differ. The basic steps of the planning process are also the same for the different command levels of planning – the military strategic, the operational and the tactical/component levels. In NATO, the planning at all levels is typically conducted in parallel with close interactions throughout the command hierarchy. The basic military concepts used in the NATO Operational Planning Process (OPP) (see Figure D.1) are the same or similar to those found in national military doctrines. However, certain key aspects of the planning process will be different for the obvious reasons that NATO is an alliance of 19 sovereign nations and that military forces only become available to the NATO commanders through the contributions from the nations in the force generation and activation process.

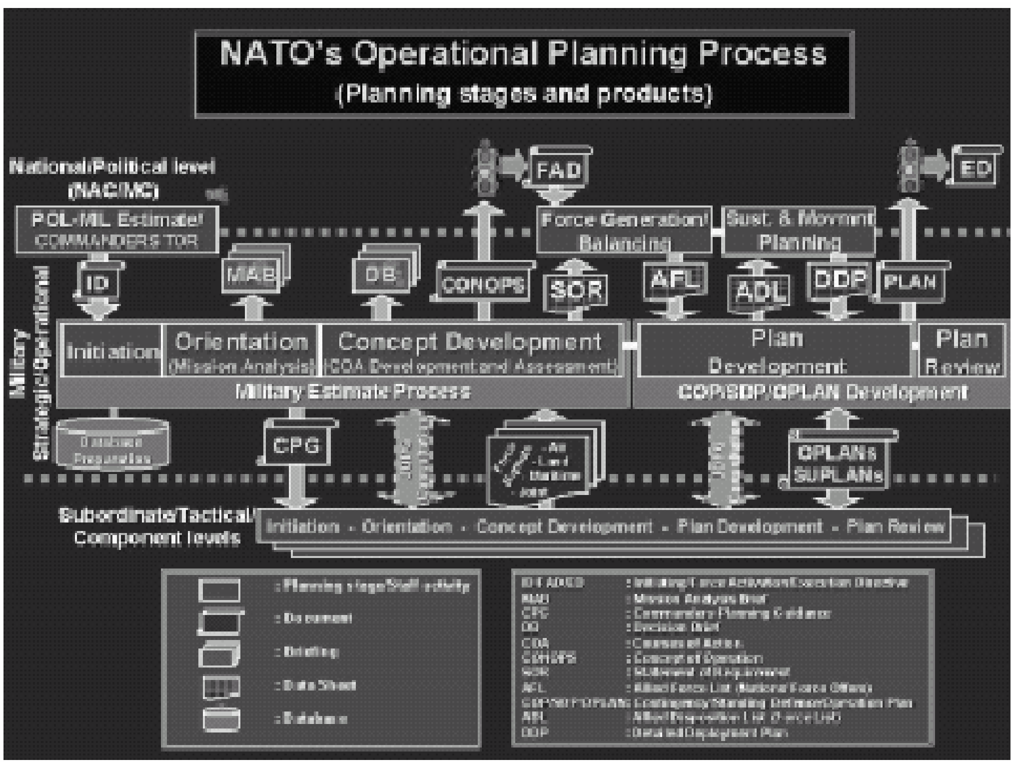


Figure D.1: Overview of NATO's Operational Planning Process.

The explicit directives from the political level are required at three critical junctures:

1) Initiation of the planning process through the Initiating Directive (ID);

2) Authorization to proceed to the force generation process with the nations through the Force Activation Directive (FAD); and
3) Authorisation to commence the execution of the operation through the Execution Directive (ED).

Planning may, of course, stop at any point in the process when the appropriate planning product for the situation at hand has been produced.

From the receipt of the Initiating Directive until approval of the Concept of Operation (CONOPS) the OPP is largely conducted within and among the NATO military headquarters with only ad hoc interaction with national military authorities. The key planning products from the strategic level military level are the CONOPS and the Statement of Requirements (SOR).

The SOR is the detailed estimate of the forces and capabilities required to conduct the operation and perform the tasks spelled out in the Initiating Directive. The SOR is expressed in terms of generic units and/or equipment.

Once the force generation process starts, the interactions between the NATO and national military headquarters expand greatly. These interactions continue throughout the force generation and the deployment planning process. At the end of the force generation process the generic requirements of the SOR have been replaced by the real and specific force contributions of the nations. For NATO-lead peace support, disaster relief or humanitarian aid operations the participation of Partnership-for-Peace nations and others will normally be invited.

The force contributing nations are ultimately responsible for the movement and transportation of their units, equipment and sustainment to the theatre destinations although the national planning is, of course, subject to close co-ordination by the appropriate NATO functional area authorities.

The key steps and tasks of the NATO OPP are further described in Figure D.2. The text also includes references to important military concepts in the planning process and should allow comparison to the concepts found in national doctrine.

OPERATIONAL PLANNING PROCESS (OPP)

Planning Stages	Steps	Tasks		Output
INITIATION	1. Receive or recognize task(s).	•Receive (or develop) Initiating Directive •Provide Strategic Military Assessment •Form the Planning Group and gather necessary information		Possible MROs
ORIENTATION	1. Conduct Mission Analysis 2. Deliver Mission Analysis Briefing 3. Develop Commander's Planning Guidance	•Review the Situation •Review Superior's Mission, Intent, Objectives & End-state •Identify/review Constraints & Restraints •Identify/determine Assumptions and Factors •Identify the Strengths and Weaknesses (OPFOR & Own) •Identify/Determine the Centre(s) of Gravity (Strategic & Operational) •Determine Decisive Points	•Identify/ determine Tasks (Assigned and Implied) •Determine Own Objectives, Desired End-state and Criteria for Success •Develop Mission Statement •Conduct Initial Force Structure Analysis •Prepare/deliver the Mission Analysis Briefing •Develop the Commander's Planning Guidance	Commander's Planning Guidance (may serve as an Initiating Directive to subordinates)
CONCEPT DEVELOPMENT	1. Staff analysis 2. Develop COAs 3. Deliver Decision Brief 4. Develop a Concept of Operations	•Analyse Situation/Make Deductions (Theatre Situation, Opposing Forces, Time and Space, Assessment of Tasks, C2, CIS) •Develop COAs •Analyse COAs (Test, War Game, Compare)	•Prepare/Deliver Decision Brief •Select Preferred COA •Develop Concept of Operations and forward for approval •Develop Statement of Requirements	Concept of Operations; SOR
PLAN DEVELOPMENT	1. Develop 2. Co-ordinate 3. Seek approval 4. Issue plan	•Refine Commander's Intent and Concept of Operations •Complete synchronisation of the plan •Confirm and identify forces through Force Generation Process •Co-ordinate and write OPLAN	•Write SUPLANS •Write Branch Plans •Seek OPLAN Approval •Issue OPLANs and SUPLAN's	Family of Plans; Allied Disposition List (ADL)
PLAN REVIEW	1. Plan Review 2. Plan Evaluation 3. Revised Decision Brief	•Conduct Progress Review for on-going operations •Conduct Periodic Review •Conduct exercises and/or war games	•Conduct Revised Decision Briefing as required •Update/Issue amendments as required •Prepare and Issue plans as required	Updates and/or develop new plans

Figure D.2: Key Steps and Tasks in NATO OPP.

Helicopter operations are part of air operations. The planning of air operations is done by CAOC (Combined Air Operations Centre), but the CAOC does not task the helicopters. CAOC is the theatre command and control centre that plans, executes and assesses joint, allied and coalition aerospace operations during a contingency or conflict. See for a more detailed description of the tasks of the CAOC the report on Air Space Management that resulted from SAS-045 (ref. [14]), e.g. in 1993 a CAOC was established to manage 50 aircraft enforcing a no-fly zone over Bosnia. To support the planning of air operations, several models are available within NATO and will be described shortly in the following sections.

D.3 NATO-MODELS

Inside NATO several models exist to support an operation. In this chapter an overview is given of the models that exist inside NATO in order to support operations. Each of the models is described and an indication is given of the possible application of the model in support of the helicopter planning model that will be developed.

D.3.1 ICC (Integrated Command and Control)

D.3.1.1 Overview

ICC is an integrated C3I environment that provides information management and decision support to NATO Combined Air Operations Center (CAOC) level air operation activities during peacetime, exercise and wartime. ICC supports the most critical Air Command and Control functions, such as Planning and Tasking, Air Task Order Generation, Current Offensive and Defensive Operations, and Recognized Air Picture

Display. In addition, ICC supports the dissemination of orders, reports and imagery between the CAOC and the Headquarters above and below the CAOC.

ICC has been developed at NC3A (NATO Consultation, Command, and Control Agency) in The Hague, The Netherlands. ICC is being used to investigate requirements of the future NATO ACCS (Air Command and Control System). ICC has evolved into NATO's interim operational capability in several ICAOC's in the NATO Northern, Central and Southern region as part of their AOPTS (Air Operations Planning and Tasking System) initiatives, and is being interfaced with national systems. Various other sites and NATO nations have decided to adopt Integrated Command and Control as their interim capability for national Air Operations Centers.

ICC (see also ref. [2]) integrates planning, tasking, intelligence, operations and airspace management modules. It is an interim solution until ACCS (Air Command and Control System) is fielded. It is expected that ACCS will be ready in 2005.

ICC consists of many modules:

- **SALTO**: The SALTO (STC Air Logic Tool) application aims at assisting the planners in both building and generating the ATO by offering a number of useful functions, like bookkeeping (usage and availability), choice enumeration and automatic constraints checking. Graphical support is offered as well in order to better present to the user functions like unit utilisation and sortie flow checks. The SALTO approach is to let the user control the planning process, rather than to provide a complete plan. SALTO assists the planner in completing this task by carrying out a number of activities that are difficult, or time-consuming of error-prone for human operators. According to the CAOC team-approach, SALTO supports co-operative planning, by automatically co-ordinating the activities of different planners who are developing the same ATO. SALTO supports OAS, OCA, AI, RECCE, AAR, EW, SAR, AEW and NATO Composite Air Operations (COMAO) Planning. The ATO can be generated according to the mission type or entirely in AIRCENT and USMTF ATO formats.
- **ASMAN**: Air Space Management procedures are basically defined in the Airspace Control Order (ACO). The ASMAN (Air Space MANagement) module (see Figure D.3) consists of automated tools for maintaining Airspace Coordination Orders. It allows the user to graphically enter airspace objects into the CAOC database and to display the ACO on the map in order to make it available for the planning and tasking activities. The ACO can be generated in AIRCENT or USMTF formats. ASMAN allows a flexible input and modifications (e.g. ASMs can be drawn on the and it allows transformations of co-ordinates).

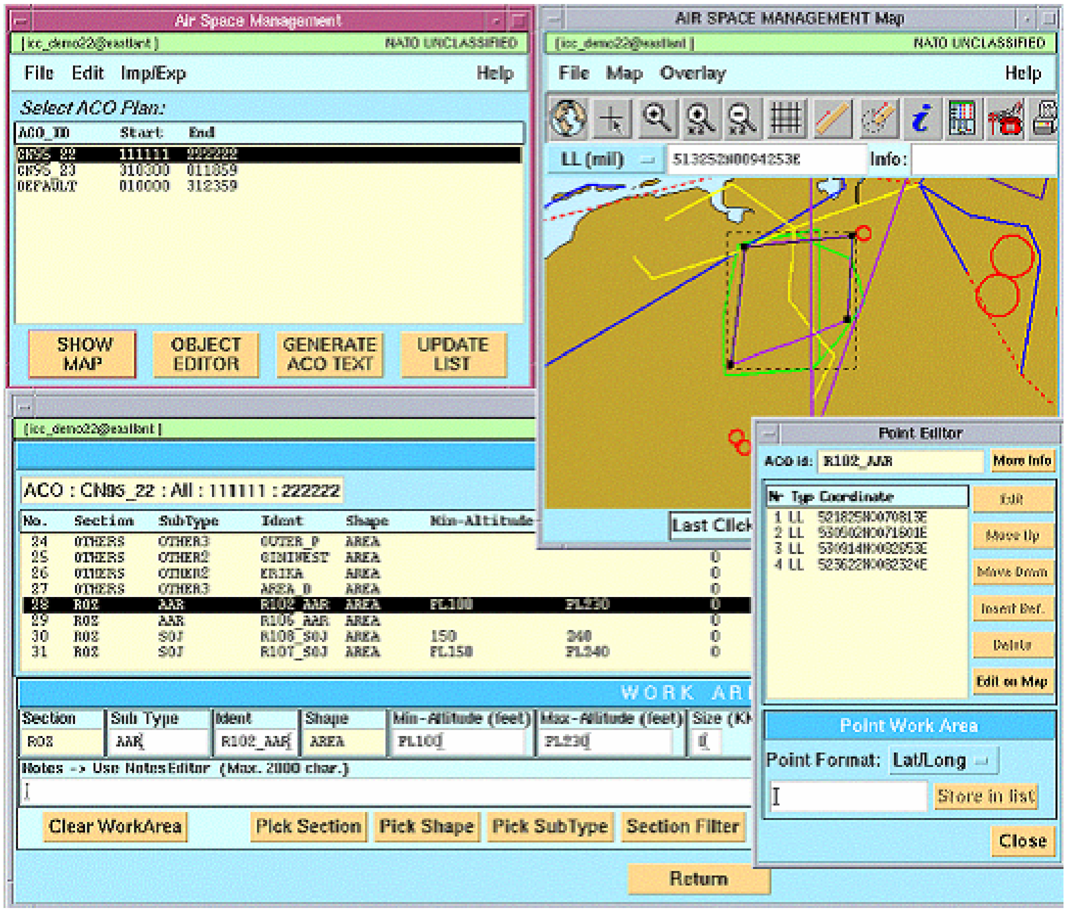


Figure D.3: Graphical Overview of ASMAN Module.

- **RESOURCE ALLOCATION**: Resource Allocation is the module assisting in the allocation of units to CAOCs. Based on the daily Air Operations Directive (AOD), Operations Order (DOO) or Master Attack Plan, it allows the planner to specify resource allocation plans, in which unit assignments and available systems are defined. These resource plans are then available for the other planning applications.
- **TGTNOM**: The TGTNOM (Target Nomination) module allows the Plans or Intel section to define a Target Nomination Plan, based on the target database and according to the specifications in the Air Operations Directive (AOD) or Daily Operations Order (DOO). This target plan is then made available for everybody in the network.
- **ADEPT**: The ADEPT (Air Defence Planning Tool) module provides the planners with automated tools to build and maintain the Air Defence posture. The graphical map provides the display of cartographic information overlaid with radar coverage, SAM coverage, the ACO and current own-force and enemy situations. ADEPT gives the planners the tools to define graphically and display AORs, BMAs, FEZs, FAORs, MEZs and other user-defined areas. The module provides a user-friendly interface for the input and modification of all elements of the OPTASKAAW/ ATOANNEXAD and automatically generates this message in the AIRCENT format for inclusion in the ACO.

- **ASVIEW**: The ASVIEW module (see Figure D.4) allows the planners to quickly and effectively check the developed ACO by providing the means of previewing the ACO execution. The module visualizes an ACO in a tabular form, graphically on a map background (2D View) and graphically in 3 dimensions (3D View). It presents a graphical simulation on a map of mission execution in the specified scenario. This tool allows the planners to quickly verify and deconflict the correctness of their plan. (automatic deconfliction of ASMs). Typical checks are the correct use or corridors, the critical timing over a target and of coordinated support missions. The module can be used for briefing the command group on future activities and can be used by Current Operations to see where their missions are supposed to be by previewing the ACO in real-time and synchronised to the exercise clock.

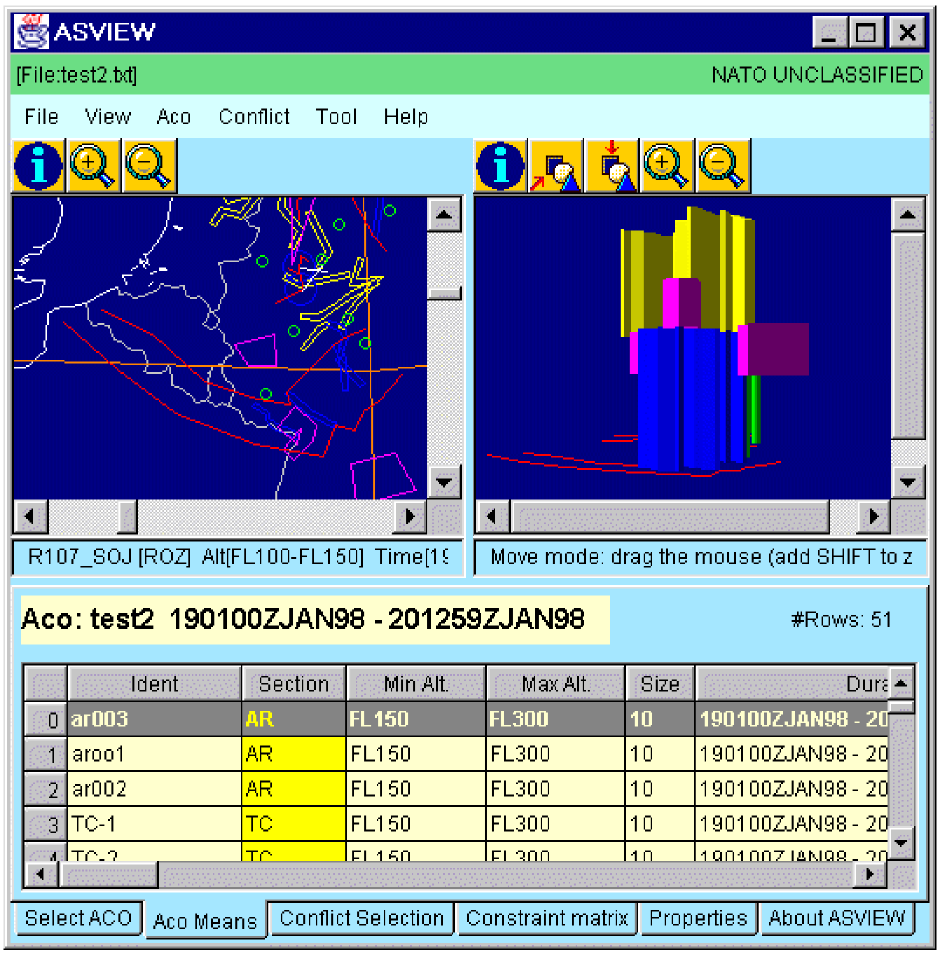


Figure D.4: Graphical Overview of the Module ASVIEW.

- **Mission Tote**: The Mission Tote module allows the operator to easily monitor the tasked missions by listing all the missions or show them on a sortie flow graph. Mission status is automatically changed as soon as the mission report is received, or as a consequence of automatic checks. Different colours are used to represent status and to give warnings about critical situations. Several summary reports can be generated automatically. This tool can also be used to monitor the status of on going air operations from a remote site on a non-interference basis. MAP: This module does not provide sufficiently detailed maps of the world, including borders, rivers, roads, etc. A new version of this module should include the use of digital maps.

- **ADAPI (COP)** Command Operational Picture (see Figure D.5): This displays recognised air pictures. (i.e. Pictures from AWACS or CRC). All sorts of overlays can also be displayed, using ARC Digitised Raster Graphic (ADRG) visualisation.

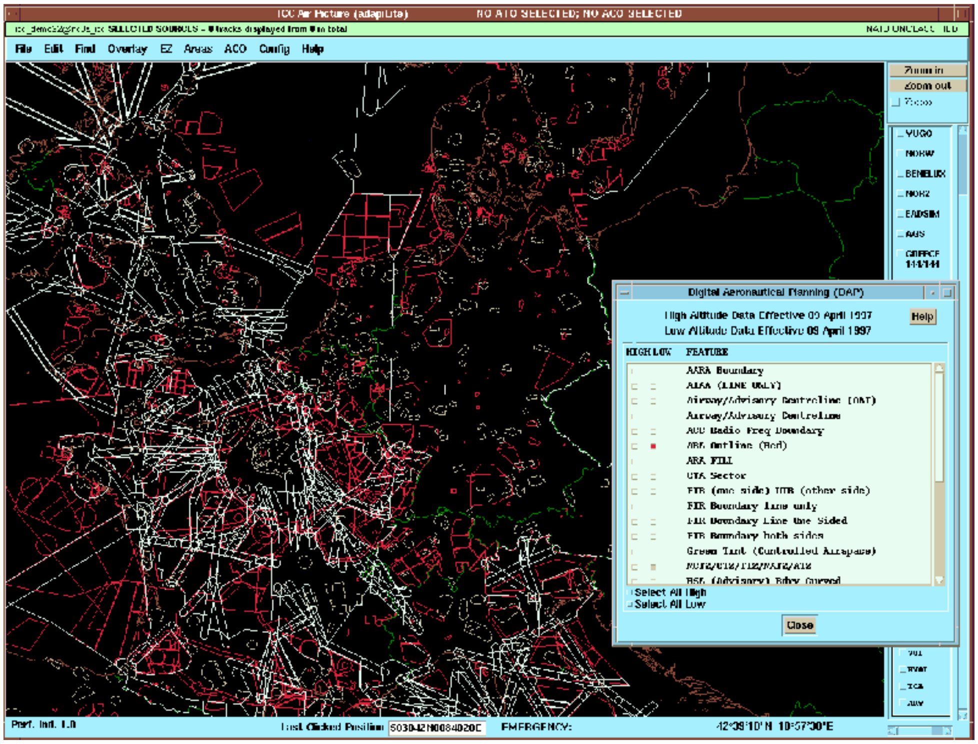


Figure D.5: Graphical Overview of Module ADAPI.

D.3.1.2 Summary

It can be concluded that ICC is mainly used for planning at higher level command and control and airspace management issues. Although ICC can be used to plan both future and current operations, it seems ICC is more focusing on current operations rather than on future operations. Nevertheless, the helicopter model that will be developed should have the capability for interfacing with ICC, as it might be necessary to use information from ICC in the helicopter model.

A two-way communication is required between the models in order to get approval for the planning. ICC is only a temporary solution. In the near future, ACCS will be replaced by ACCS.

D.3.2 ACCS (Air Command and Control System)

D.3.2.1 Overview

The ACCS programme (see also ref. [3]) is intended to combine, and automate, at the tactical level the planning, tasking and task execution of all air defensive, offensive air and air support operations in a single system operations. Although it was originally designed to be used in an air defence environment, Its scope is much broader than just air defence. The system will be composed of a balanced mix of static and deployable entities.

When operational, the ACCS will provide a unified air command and control system, enabling NATO's European nations (including new Alliance members) to seamlessly manage all types of air operations over their territory, and beyond. NATO members will be able to integrate their air traffic control, surveillance, air mission control, airspace management and force management functions.

ACCS will incorporate the most modern technologies, and will make full use of up-to-date data link communications. Through its open architecture, the system is already evolving to meet emerging operational requirements such as those associated with theatre missile defence, and it will be able to adapt to a changing operational environment such as network centric warfare.

ACCS will be fielded at the planning and tasking level in Combined Air Operations Centres (CAOC), supporting the Joint Force Air Component Commander in exercising OPCON over his subordinate forces. ACCS will be fielded at the execution level in control centres (using the generic acronym ARS) that will provide facilities for aircraft control and production and dissemination of the Joint Environment Picture. At present, NATO and the nations have requirements for 5 static and 2 deployable CAOCs, while 18 static and 2 deployable ARS are also planned.

The added value of ACCS is that it is a modern architecture for all entities, but also it can provide real-time air picture enhancements. Figure D.6 gives an overview of the ACCS architecture (see also ref. [4]).

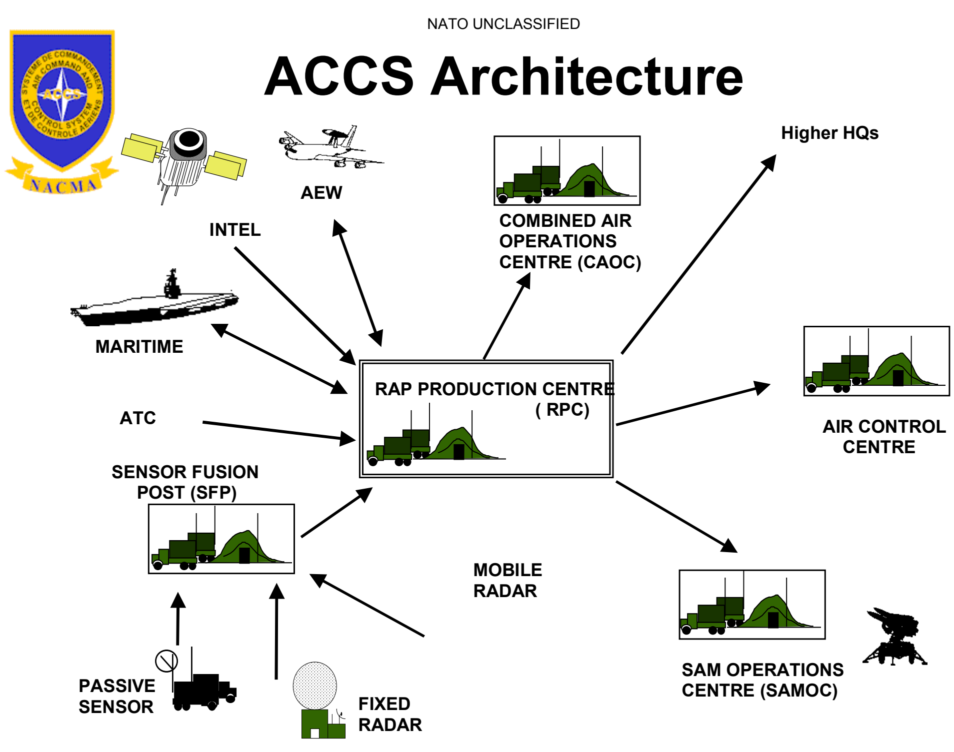


Figure D.6: Overview of ACCS Architecture.

Additional requirements in the areas of Theatre Missile Defence and Ground Surveillance are already being addressed. ACCS follows an incremental approach, and the software development should be ready in 2005.

D.3.2.2 Summary

ACCS is not a planning model, but it is an operational command and control system. Here also, an interface should exist between ACCS and the helicopter model. ACCS does not provide two ways of communications.

D.3.3 ADAMS (Allied Deployment and Movement System)

D.3.3.1 Overview

ADAMS has been developed in support of multinational force movement planning (strategic deployment). This system is in wide use throughout NATO and NATO nations for analysis, generation and co-ordination of movement plans. ADAMS provides the users with the tools to plan and manage deployment operations. The software also includes conversion modules for interfacing between ADAMS and national mobility management systems.

ADAMS is a program developed by NC3A. (see also refs. [5] and [6]). The purpose of this model is to provide the theatre commander with strategic deployment information on the dates, times, locations and equipment of arriving forces. It was designed for planning and monitoring strategic deployments within NATO's area of responsibility.

It is a system for analysis, planning and executing deployments at strategic level, it is primarily about a system of exchange of information between combined forces on the intentions of movement. The responsibility for the movements remains national. The system makes it possible however to mutually exchange in a common format the plans of movement and to detect the inconsistencies (like, the simultaneous use exceeding the capacities of a port) on the level of the SHAPE to make their resolution possible. The developments in hand are directed towards the assistance with the analysis and the co-ordination of the plans. In particular, the system is not currently regarded as a system of "command and control". ADAMS is not a movement control program. It cannot manage the movement of forces past their final destination. ADAMS provides visibility of movements projected or reported over time rather than real-time.

D.3.3.2 Modules

ADAMS is divided into seven specific components shown in Figure D.7.

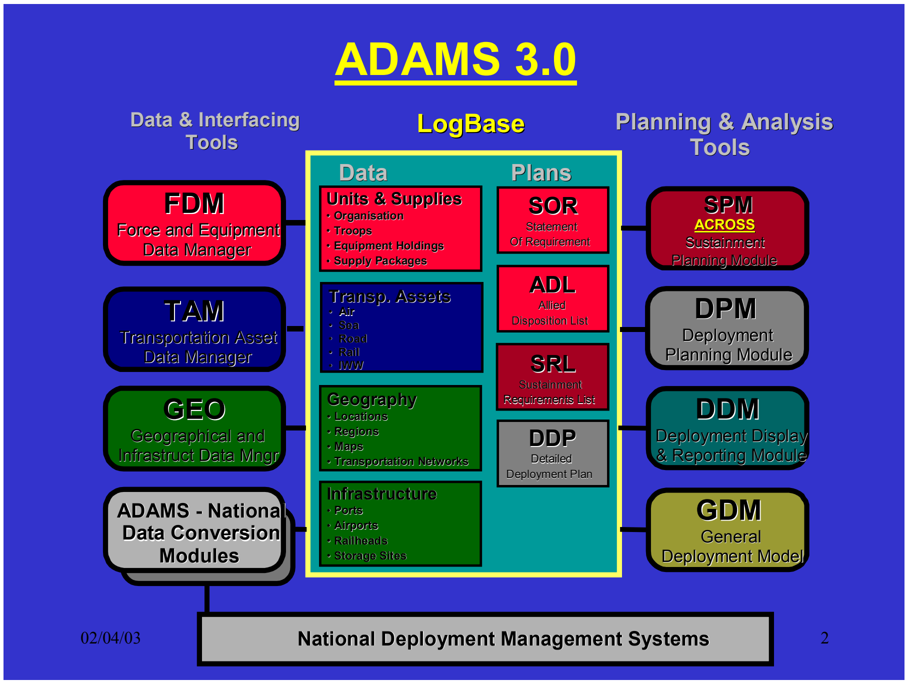


Figure D.7: Modules in ADAMS.

- **FDM**: the force data management module for the selection of the forces.
- **TAM**: the transportation asset data manager for the planning/management of the means of transport which allows, for example, to analyse the additional means of transport necessary for a given planning.
- **GEO**: the geographical and infrastructure data manager.
- **DPM** (the Deployment Planning Module): This module, the forces are organised based on the method of deployment and matched to their mode of transport. The module facilitates planning of movements by all modes from all available facilities. Decisions are made on transportability and height and width restrictions, which influence the type of assets that will be sent and the routes they will follow into the theatre.
- **SPM** (The Sustainment Planning Module): this module determines the sustainment requirement for forces. This module also determines the transportation requirements for supplies based on the consumption factors stored in the database.
- **DDM** (The Deployment Display Module): this module is used to perform major analysis and deconfliction of the proposed movements in accordance with the detailed deployment plans. ADAMS examines the DDP and uses the system to deconflict national movements.
- **GDM** (The General Deployment Module: this module is a simulation model that uses many "what-if" scenario's to develop various options to the basic plan. This model evaluates the "what-if" of possible changes to deployment assets, infrastructure, and timelines based on the existing plan. After these scenarios are run, the results can be analysed to determine the effects of changes to the basic plan. This is a valuable tool for planners to use because of the constant changes that normally occur before a deployment.

D.3.3.3 Data

Data is organised in a relational database with a separate Database Management Module. Three databases are accessible by the user:

- Forces (units and supplies);
- Means of transport (transport assets); and
- Facilities (infrastructure: ports, airports, railheads, storage sites).

Data concerning force descriptions are nationally seized by the national HQ. Only the characteristics of obstruction of the major hardware (dimensions, weight) remain to be seized.

The majority of the sealift data are ensured by the LLOYD COMPANY who provided his descriptive files to the NC3A. Thus 2 600 civil ships appear in the base of the NC3A. It remains to describe the trains, the civil aircraft and military as well as the commercial barges. Facilities should not be the subject of a systematic data entry. It is advised to seize them progressively whenever there is a need.

D.3.3.4 Operations of Planning and Movement

The following operations constitute the core of ADAMS.

A graphic help makes it possible to the user to carry out the first planning. The user has the possibility of bursting the units (while transporting the personnel by aircraft and the hardware either by airlift according to the type of aircraft available or per sea) and assigns them to different means, then amalgamates them near their destination.

After having chosen the means of transport and taking account of the operational availability of the unit, fine planning can be carried out. As an example, consider the units which do not have a resource allocation of transport and which must use the maritime way without knowing the name of the RORO. Then the planner selects them and determines the resource allocation of transport for them.

Taking into account the speed of the means of transport which appear in the data base, the arrival time of the unit can easily be deducted. Thus very simplified calculations of duration of movement are possible in the system, but the user can replace them by his own estimates as well. Figure D.8 and Figure D.9 give an overview of the deployments display view and deployment display report, respectively.

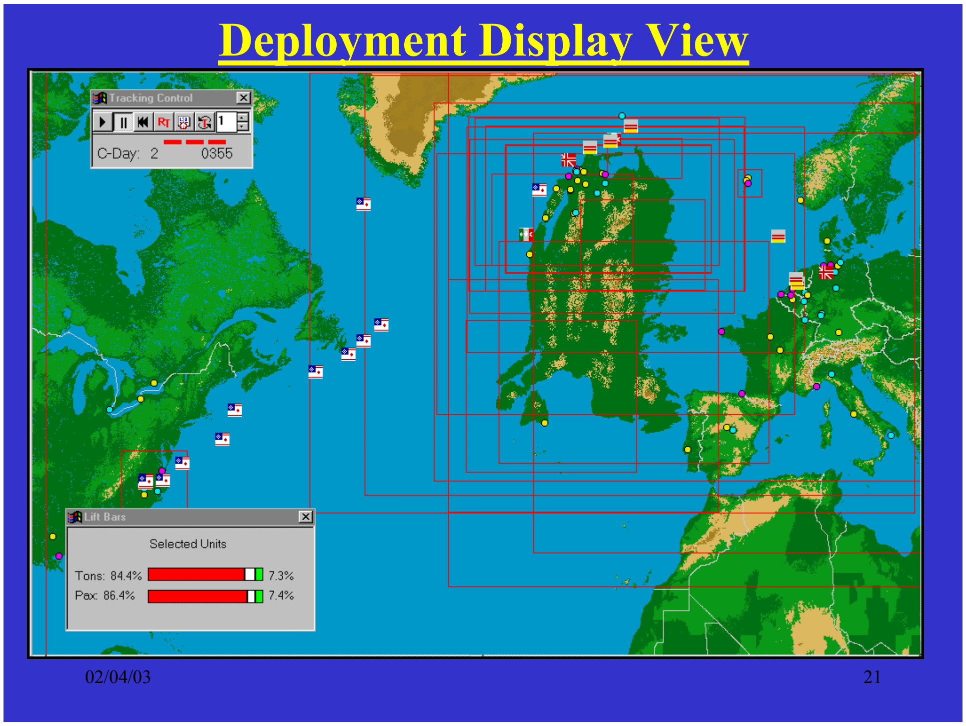


Figure D.8: Deployment Display View of ADAMS.

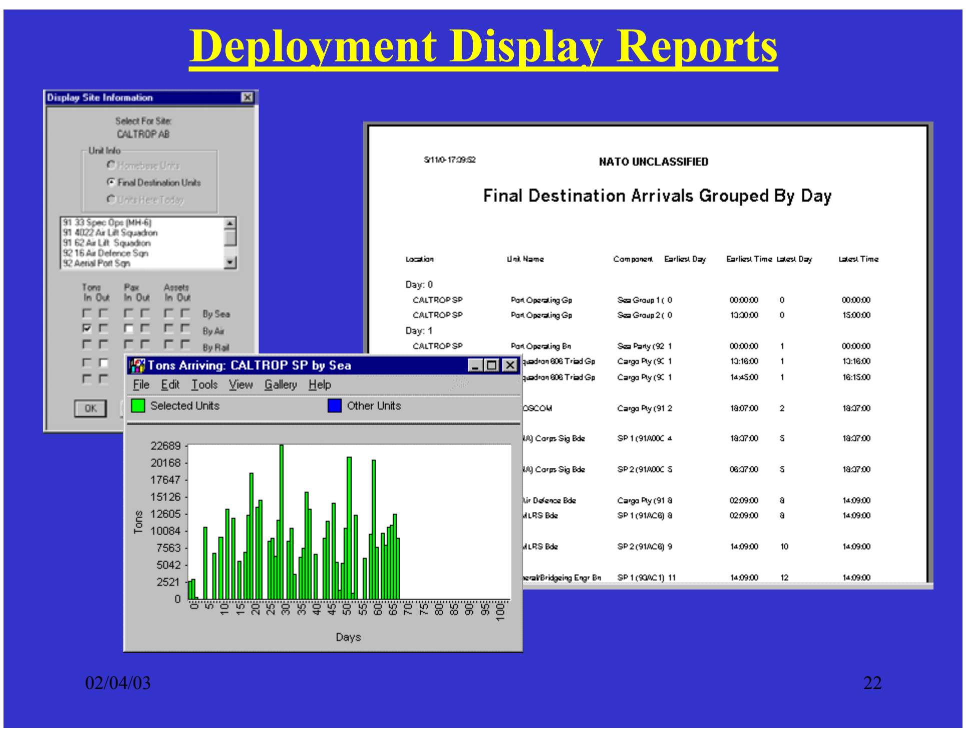


Figure D.9: Deployment Display Reports of ADAMS.

D.3.3.5 Summary

ADAMS is a planning system for planning of deployment and re-deployment at strategic level. The helicopter model to be developed will not take into account planning at strategic level. However ADAMS may plan the strategic deployment and re-deployment of helicopters. So considering the possibility that the output of ADAMS be used in the helicopter planning model, it may be useful to have an interface between ADAMS and the helicopter model.

D.3.4 LOGBASE

D.3.4.1 Overview

ADAMS uses the LogBase logistics database which has been developed over the last decade. LogBase is common to all the logistics functional area sub-systems (LogFASS), which are being developed as a part of the ACE Alliance Command Communication and Information System (ACCIS). ADAMS therefore shares this common database with both the ACROSS stockpile planning system and the Logistics Reporting Tool (LogRep). This in turn means that such features as database security, auditing and integrity controls are shared by all three systems. Other related NATO operational and logistic systems such as the Force Identification System (FIDS) are planned to share LogBase. A schematic outline of the relationship between the various LogFASS systems is shown in Figure D.10.

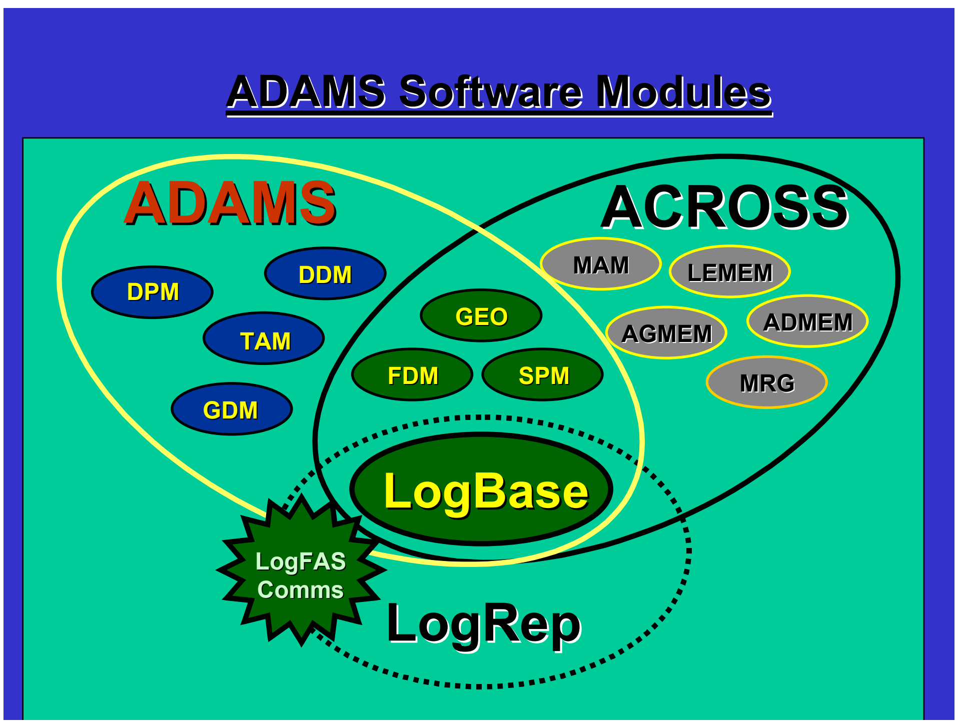


Figure D.10: Software Modules ADAMS.

Information can be exchanged within other external NATO planning systems by means of intermediate formats such as ASCII files or spreadsheets. NATO standards are used in LogBase reference data such as unit types, country codes, etc., whenever possible to facilitate interoperability.

Reportable Items Codes: LogBase includes the BI-SC Reportable Items Codes (RICs) for ACE and SACLANT. RICs have been introduced into LogBase to allow capability based queries to be carried out. The RIC system allows equipment and munitions to be classified and reported using a common set of values which transcend language, national nomenclature and coding systems. The RIC system has been developed using service expertise from both SCs and their subordinate Headquarters. The system represents a significant step forward in the exchange of information between allies and is used not only in LogRep, but also in the various modules within FIDS, ACROSS and ADAMS.

D.3.4.2 Summary

LOGBASE is a core database related to assets, forces, geography, infrastructure, medical, movements, supply and targets. From the information available on LOGBASE it seems this database provides less detailed to be of any use of the helicopter planning model.

D.3.5 TOPFAS (Tool for Operational Planning, Force Activation and Simulation)

D.3.5.1 Introduction

TOPFAS is designed to speed and improve the fidelity of the NATO operational planning process. This system will pervade all levels of operational planning from strategic/operational level through to the tactical level.

TOPFAS (see also ref. [1] and [7]) is the data and planning support system for the operational planning and force activation in accordance with the NATO Operational Planning Process (OPP). It will provide a common database and tools for the planning process as well as a common repository of the operational plans and the audit trail for the Force Generation Process. TOPFAS supports the planning process with graphical tools to the greatest possible extent; i.e. graphical lay-out of the operational design, phases and tasks, geographical mapping tools for the specification of operational planning factors and environmental conditions; and for the disposition of the forces. Troop-to-task rules, combined with military expertise, are the basis for the identification of force requirements.

D.3.5.2 Scope

The scope and aim of the TOPFAS development is to provide software and data support for all stages and activities of the OPP. The operational planning process consists of the following stages:

1) **Initiation** based on the initiating directive and including the receipt of planning inputs and preparation of the database.
2) **Orientation** with focus on the mission analysis, operational design and the identification of assigned and implied tasks and planning factors.
3) **Concept Development** with identification of the preferred course of action (COA) to be developed into the concept of operations (CONOPS) including force requirements.
4) **Plan Development** with the refinement of the CONOPS and detailed planning based on the actual forces and capabilities provided by the nations.
5) **Plan Review** for further assessment, large-scale war-gaming and exercising, including the adaptation of the force requirements to the changing operational environment.

However, the sustainment and deployment planning are handled by other special-purpose logistics management systems. The TOPFAS functionality required for this is limited to the interfacing with these. For the rest, the stages and steps of the NATO OPP are built into the TOPFAS software in the form of a Planning Wizard that guides the planner through the process and the associated TOPFAS functionality. TOPFAS is a joint planning tool. This means that the air, land, maritime and other service specific issues are all addressed within a common framework.

D.3.5.3 Users

The primary users of TOPFAS will be the NATO Strategic Commands, Combined Joint Planning Staff, Regional Commands and other NATO military headquarters with designated operational planning tasks.

D.3.5.4 Database

TOPFAS interfaces indirectly with the NATO Defence Planning Process through the planning situations and generic force definitions. In actual planning it will interface directly with Intelligence Information Systems for the preparation of the battlefield, and with the Logistics Management Systems for sustainment and deployment planning. OpsBase is the relational database for the family of TOPFAS applications. It is fully harmonised with LogBase, the relational database for the family of NATO Logistics Management Systems, and holds the relevant information about military tasks and planning factors as well as the generic and real forces and equipment that may be expected to participate in or impact on NATO-lead operations.

The planning functionality in TOPFAS requires that the database include extensive data on the military units that nations may contribute in a NATO-lead operation. Although the initial planning results are expressed in terms of requirements for generic units and capabilities, it is clear that the planning from the outset needs to take into account the limitations of real capabilities. Once the planning has progressed to the force generation and balancing, the real unit operational data requirements become obvious. By itself, the need for data management functionality within TOPFAS for the operational data on national units, their organisation and equipment makes the tool useful beyond the operational planning requirements.

D.3.5.5 Summary

The TOPFAS planning module tool is a module that helps the user go through the OPP and prepare an operational plan. It also plans at strategic level. Although TOPFAS does not concern the planning of helicopters, the model may be useful, as it guides the user through the OPP. The OPP is also one of the starting points for the helicopter model. Together with ACROSS and ADAMS, TOPFAS becomes the basis for further logistical planning of sustainment and transportation by the NATO Logistics Management Systems. Furthermore TOPFAS may provide output that may be used as input for the helicopter model.

D.3.6 PATHFINDER

D.3.6.1 Overview

The PATHFINDER programme is the flagship programme of the NATO Modelling and Simulation Group (NMSG), which is investigating advanced technology and capability for assembling federations of national models and simulations that can support the exercising and training of high level commanders and their component commands. This programme is tied to the original M&S Master Plan (MSMP) which is the base plan underpinning PATHFINDER activities. Essentially, the agreed MSMP stated that NATO should conduct a PATHFINDER development of an HLA-based federation of national simulations building on the experience base established during NATO's Distributed Multi-National Defence Simulation (DiMuNDS) project. PATHFINDER is also connected to the Defence Capability Initiative (DCI); an activity launched at the 99 NATO Washington Summit with the objective to improve relevant military capabilities with focus on improving interoperability. DCI item Effective Engagement 9 (EE9) states that NATO nations should support the development and implementation of operational simulation devices in order to enhance interoperability in training and the decision-making process. Finally, PATHFINDER is also linked to SACLANT Concept Development and Experimentation (CDE) initiative being part now of the CDE database. The CDE initiative is a process to match concepts that meet national, multi-national or NATO-wide military requirements helping to explore innovative approaches and "leap-ahead" capabilities, exploiting opportunities for transforming NATO defence forces.

As stated in the NATO Modelling and Simulation Master Plan (MSMP), «exercising the CJTF» has been identified by the Major NATO Commands (MNC)s as the top priority requirement to steer M&S activities in the Alliance. As an answer to this requirement, the PATHFINDER Programme, considered by SACLANT as the most important NMSG Programme, is to provide federations of national simulations and decision support tools integrated and linked to NATO operational C3I systems, to exercise and train the CJTF HQ Staff on the conduct of Crisis Response Operations (CRO). The federations would be planned, integrated and tested in common, but the individual national simulation developments would be executed by the participating nations. Advantages of this approach are that each simulation would meet both national and NATO requirements, the cost of development being shared between both sides.

D.3.6.2 Summary

The PATHFINDER model is not a planning model, but is a software solution that supports exercises and training of high level commanders and their component commands. At the moment it does not look really necessary that the helicopter model takes into account the techniques that are being used in the PATHFINDER programme.

D.3.7 NATO Data Models

The helicopter planning tool that will be developed will make use of a data model or database. Inside NATO, a number of data models are being accepted and being used. This section describes the NATO data models that are available and also the applicability of the NATO data model to the helicopter planning tool. The following NATO data models are considered:

- NATO Product Data Model (NPDM);
- LogBase;
- ATCCIS;
- NATO Corporate Data Model; and
- NATO Directory Data Model.

D.3.7.1 NATO Product Data Model

The NATO Product Data Model (NPDM) is a conceptual data model. It defines a common set of data definitions and data structures to support Defence System technical information management, throughout the system life cycle, in the context of NATO nations and NATO industries.

The NPDM addresses the NATO requirement for data interoperability between different Information Systems by delivering a common data semantic and thus enabling consistency of interfaces at the information level without requiring standardization of hardware or software. The NPDM uses EXPRESS, ISO 10303-11, as the modeling language to enable both human understanding and computer processing of these semantics.

NPDM has been developed by the NATO CALS Office (NCO) under the guidance of the NATO CALS Management Board (NCMB) with contributions from Association GOSET, France, Daimler Chrysler Aerospace (former DASA), Germany, Eurostep Limited, United Kingdom and Metasistemi S.p.A, Italy.

The NATO Product Data Model consists of data that is not directly necessary for the helicopter planning model. NPDM describes logistic information related to a product. It can be concluded that the helicopter planning model is not dependent on information in the NPDM or that the helicopter planning model will feed NPDM.

D.3.7.2 ATCCIS (Army Tactical Command and Control Information System)

The ATCCIS specification is a managed interface between C2 information systems. When incorporated into a system, it enables interoperability of information between any other system that also incorporates the specification. Battle space data is transferred as information. The meaning and context of the information is preserved across national and system boundaries precisely and without any ambiguity. By 2002, 18 nations and NATO agencies had incorporated these specifications into their programmes and systems.

One of the two main components of the ATCCIS specifications is a data model, The Land C2 Information Exchange Data Model (LC2IEDM). It is a product of the analysis of a wide spectrum of allied information exchange requirements by 16 nations. It models the information that allied land component commanders need to exchange (both vertically and horizontally). It serves as the common interface specification for the exchange of essential battle space information. The function, implementation and the display of the host C2 application is not the concern of ATCCIS. System developers incorporate the ATCCIS specification and include a single interface to it. Thereafter no further interfaces are required to interoperate with any other ATCCIS enabled system. The LC2IEDM is in its 5th generation (version 5). The previous version, LC2IEDM v2, is the core of the NATO Reference Model and is also a view model of NATO Corporate Data Model (STANAG 5523 / AdatP-32).

ATCCIS itself is not a specification that should be considered when developing the helicopter model. However, one of the main components of the ATCCIS specifications is the LC2IEDM data model. This data model is widely accepted inside NATO and should be considered as a basis for definitions and a guideline when developing the helicopter planning model.

D.3.7.3 NATO Corporate Data Model

The NATO Corporate Data Model has been developed by the NATO Data Administration Group (NDAG). The NDAG is a multi-national working group, responsible to the Information Systems Sub-Committee (ISSC) for the development and maintenance of NATO data management policies for recommendation to the NC3B, together with guidance on the coherent implementation of data management and administration across NATO.

The prime purpose of the NATO Corporate Data Model is to provide a source for Standard Data Elements (SDE), in direct support of NATO s Policy for Data Management. The use of SDEs enhances interoperability among NATO C3 systems, facilitates increased data sharing, reduces data handling costs and leads to better data accuracy, consistency and timeliness. It is a Joint Data Model and comprises:

- A Reference Model, which contains the semantic definitions of the SDEs, their inter-relationships and necessary information about data structures (e.g.: maximum field length, data types);
- View Models, which are project-centric data views encapsulating specific examples and combinations of the generic SDEs as well as project specific data elements that can not be found in the Reference Model; and
- Semantics and structure of meta data.

ATCCIS itself is not a specification that should be considered when developing the helicopter model. However, one of the main components of the ATCCIS specifications is the LC2IEDM data model. This data model is widely accepted inside NATO and should be considered as a basis for definitions and a guideline when developing the helicopter planning model.

D.3.7.4 NATO Directory Data Model

The NATO Directory Data Model has been developed by the ACP 133 Task Force within the ACP 133 B and has been adopted by the NATO Directory Services Working Group (DSWG). The DSWG is a multi-national group responsible to the Information Systems Sub Committee (ISSC) for the development, review and harmonization of policies, requirements and procedures concerning directories for use by NATO. The aim is to ensure interoperable electronic directory services across the tactical/strategic, SC, RC, GSRC and national boundaries within NATO.

The purpose of the NATO Directory Data Model is to maintain the NATO Directory Schema that encompasses the directory entry types, object classes, attributes, matching rules, name forms, and structure rules that are necessary for specifying the information that is stored in the Allied Directory System. Components of the NATO Directory System shall implement all of the standard object classes, attributes, and name forms defined in X.501, X.509, X.520, X.521, and X.402, as profiled in Annex D of ACP 133 B and the relevant NATO supplements. The NATO Directory Schema, called Common Content, employs the standard schema elements and also includes object classes, attributes, and name forms defined in ACP 133 B especially to meet NATO requirements. The NATO Directory Schema is specified in Annex B of ACP 133 B and is profiled in Annex D of ACP 133 B.

The NATO Directory Data Model is a Joint Data Model and comprises:

- A Reference Model, which contains the semantic definitions of the used object classes, their inter-relationships and necessary information about data structures (e.g.: maximum field length, data types); and
- View Models, which are project-centric data views, through the project specific selection of the necessary object classes from the general repository.

The NATO Directory Data Model should be considered as a basis for developing software models. The definitions mentioned in this model should be taken into account when developing the helicopter planning model.

D.3.8 Summary of NATO-Models

In Figure D.11, an overview is given of the NATO models discussed in this section and their application. As the focus of SAS-045 is on the planning of helicopter operations at operational and tactical level, only TOPFAS is of interest. However, only TOPFAS does not take into account helicopter operations. Thus, at this moment there is no NATO-model available for planning helicopter operations at operational and tactical level.

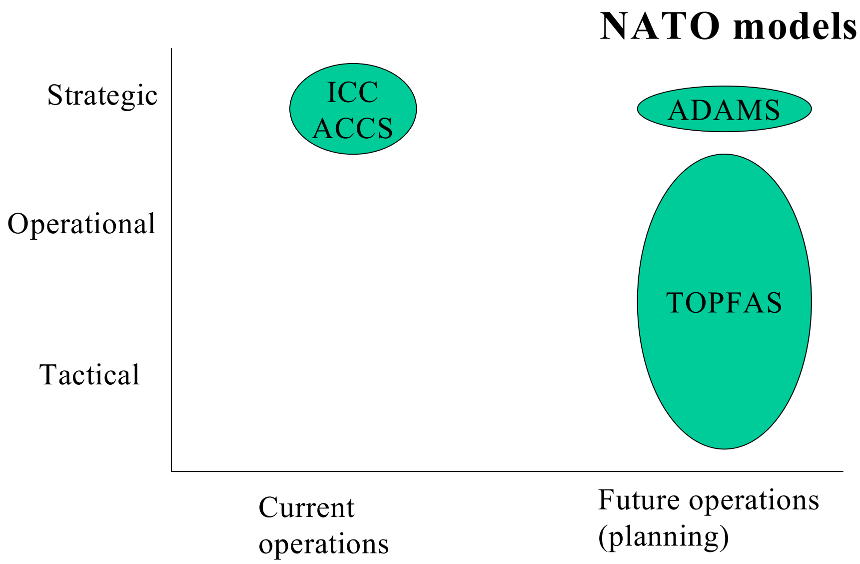


Figure D.11: Overview of NATO-Models.

D.4 PLANNING MODELS WITHIN NATO NATIONS

Among the planning models within NATO nations, there exists only one model (FELPATH) that considers solely helicopter planning. The available US models are being used mainly for planning air mobility. A short description of these models follows. Turkey possesses a planning an information system for the air force, but this system cannot be used for planning helicopter operations. Although some information (1998) has been obtained about a German model called Daedalus, this model has not been developed any further since then. At the first SAS-045 meeting, France presented information on the PROFORCE model. PROFORCE can be used for aircraft transportation and more specifically for the strategic move. France is also preparing the development of a helicopter tool to be used for airmobile studies. Furthermore, according to the Netherlands Army, the Spanish army should possess a planning model available, but further details on this model could not be found.

Other existing planning models mainly focus on aircraft and not on helicopters, e.g. Military Airlift Command (MAC) in the US has developed some models and heuristics in this area. See refs [8], [9] and [10]. CAMPS is an example of such a model.

D.4.1 CAMPS (USA)

The Consolidated Air Mobility Planning System (CAMPS) (see also ref. [11]) is Air Mobility Command's (AMC) command and control migration system that will combine the functionality of legacy systems, AMC Deployment Analysis System (ADANS) and Combined Mating and Ranging Planning System (CMARPS), into a single, integrated AMC planning and scheduling system for both airlift and air refuelling missions. Currently, AMC uses CAMPS and ADANS for planning and scheduling airlift missions, and CMARPS for planning and scheduling air-refuelling missions.

CAMPS provides the mission planner with an integrated view for planning and scheduling AMC air mobility resources to support peacetime, contingency, humanitarian, and wartime operations. CAMPS consolidates the baseline functionality of ADANS and CMARPS into a new, integrated architecture that will be Defence Information Infrastructure Common Operating Environment (DII COE) Level 5 compliant (eventually Level 7) and will conform to the U.S. Transportation Command (USTRANSCOM) and AMC enterprise environments.

CAMPS provides advanced user capabilities for operational planning and allocation management. Deliberate planning, contingency or crisis action planning, and support functions have been added in February 2002. A single Oracle database provides one-time data entry and is consistent with Department of Defence (DoD) standard data definitions and the AMC Corporate Database architecture. CAMPS uses Windows NT 4 and Windows 2000 in a client-server environment and will take advantage of DII COE software components to provide administration and support services.

CAMPS is a powerful, deployable planning tool that presents an integrated view of planning and scheduling mobility resources (airlift and tanker aircraft) while lowering life cycle costs. It provides one-time data entry and Department of Defence (DoD) standard data elements. CAMPS enhances usability and minimises training requirements for users and systems personnel, utilising several industry standard Commercial Off The Shelf (COTS) products.

As the Program/Systems Overview, CAMPS will provide a complete, integrated view of mobility requirements, resources, and commitments; and status of air mobility planning and scheduling activities at each stage from concept initiation through mission closure. It will facilitate development and presentation of alternative resource options to permit more efficient use of available assets. CAMPS replaces the legacy AMC Deployment Analysis System (ADANS) and Combined Mating and Ranging Planning System (CMARPS). It is easiest to understand the functionality of CAMPS by understanding the functionality of the legacy systems.

D.4.2 ADANS (USA)

ADANS stands for the Air Mobility Command (AMC) Deployment Analysis System (ADANS) (see also ref. [12]). ADANS contains five main software components supporting Headquarters (HQ) AMC and the TACC (including the Alternate TACC (ATACC)). It provides automated support for air mobility deliberate planning, crisis/contingency planning, peacetime and wartime scheduling, and analysis of the air mobility system.

The five major components are as follows:

- **Airlift Flow Planning:** Supports deliberate, contingency, and exercise planning from peacetime through crisis/contingency to wartime.
- **Barrelmaster:** Allows AMCs Tanker Airlift Control Center (TACC) to allocate aircraft to specific missions (migrated to CAMPS).
- **Channels:** Plans regularly scheduled airlift missions (migrated to CAMPS).
- **Special Assignment Airlift Missions (SAAMs):** Plans one-time unique missions (migrated to CAMPS).
- **Air Refueling:** Plans and schedules air refueling missions (migrated to CAMPS).

ADANS is AMC’s primary headquarters-level computer system supporting Department of Defence (DoD) airlift planning and scheduling. It is an AMC Command and Control legacy program supporting airlift and tanker planning, scheduling, and analysis during peacetime, crisis, contingency, and wartime. ADANS provides planners and schedulers automated tools necessary to plan and schedule the extensive number of air mobility mission flown by AMC during peacetime, contingency, and humanitarian missions. ADANS schedules airlift and air refuelling missions under all scenarios: routine channel missions; special assignment airlift missions; field training exercises; and contingency, crisis, and humanitarian missions.

ADANS has a narrowing customer base. After 1 Oct 2000, only the TACC and HQ AMC/DOX use the airlift flow planning component of ADANS. The other ADANS component functionality has been migrated to CAMPS. HQ USAFE also uses ADANS to support theater airlift planning and scheduling. ADANS runs on Sun Solaris workstations and utilizes a Sybase Database. ADANS has been replaced by the Consolidated Air Mobility Planning System (CAMPS) in February 2002.

In addition to the essential tools for editing, managing and analysing data, ADANS provides three optimisation models developed by CTA Operations Research experts. A linear programming based model estimates the capability of the airlift system to support a large movement. A second model supports schedulers who plan the regularly scheduled peacetime movement of aircraft from the U.S. to overseas locations. By combining linear and integer programming techniques, this model suggests how often each route should be flown to satisfy the projected demand. The largest of the models is a dynamic programming based scheduler, which generates detailed aircraft itineraries automatically based on the planner's constraints.

ADANS was made operational in early 1990 and has been used daily, 24 hours a day, since that time to schedule over 100,000 operational missions and millions of potential wartime scenario missions. During late 1990 and early 1991, ADANS was used to plan and schedule the largest airlift in history – Operations DESERT SHIELD/DESERT STORM – to support the Persian Gulf war.

D.4.3 CMARPS (USA)

The Combined Mating and Ranging Planning System (CMARPS) is the system used by AMC to schedule air refuelling for deployments from the continental U.S. to other parts of the world. The CMARPS model has been replaced by the CAMPS model in 2002.

D.4.4 TUAFIS (Turkey)

D.4.4.1 Vision and Goals

Vision of TUAFIS is to provide a comprehensive information system; including custom built Intelligence, Operations and Flight Training modules, integrated with Resource Management modules implementing "best practices", and in this manner ensure that TuAF becomes an exemplary air force setting standards for other air forces. Two goals of TUAFIS are explained as follows:

- **Functionally**
  - All core TuAF functional processes must be implemented by TIS in a manner responsive to the strategic, tactical and operational dynamics of TuAF; and
  - Where desirable, promoting the improvement of these processes by the introduction of best-practices and standard procedures.
- **Technically**
  - Survivability in peace, crisis and war conditions;

- High availability (>99.98%); and
- Ease of maintenance and development through open standards and platform independence.

D.4.4.2 Architecture

Figure D.12 illustrates the architecture of TUAFIS. It basically consists of two main modules: custom-built Battle management and ERP based Resource Management modules.

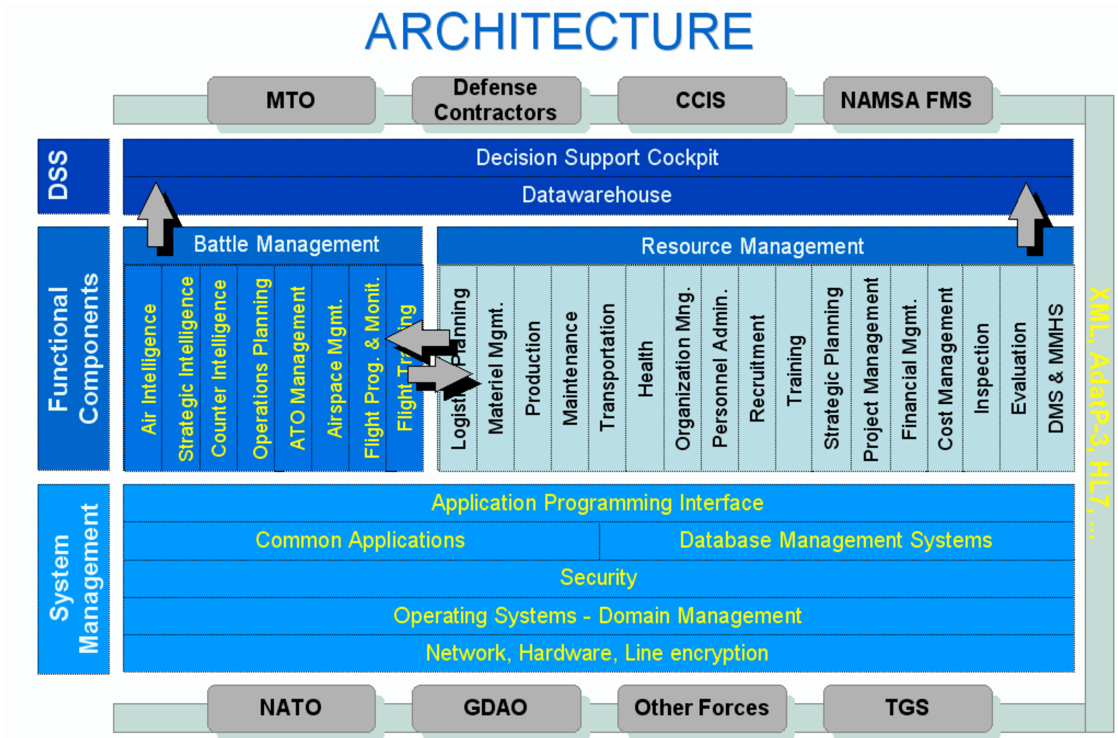


Figure D.12: Illustration of Architecture TUAFIS.

D.4.5 DAEDALUS (Germany)

Daedalus (see also ref. [13]) is a model that uses simulation to plan the deployment of aircraft and helicopters in Air Mobile operations (Luftlandeoperationen). It supports transport from the staging area to the deployment area including refuelling at a FARP. Input is the available fleet of aircraft and helicopters, personnel and material to be transported and operational restrictions.

The following elements are calculated:

- Number of aircraft and helicopters required;
- Time required for each wave;
- Time that transport resources are used; and
- Use of fuel.

The output of this model should also be an Air Movement Table.

A prototype and demonstrator of this model has been developed by IABG in 1998. After 1998 the model has not been further developed, neither has is it been used.

D.4.6 PROFORCE LEOPAR (France)

D.4.6.1 Description

PROFORCE LEOPAR can be used for aircraft transportation and more specifically it aims to simulate a force projection on one or more theatres of operation to assess the effectiveness of a fleet of aircraft or other means of transport for a given scenario.

Using data describing the force to be transported and the means of transport to be used, PROFORCE LEOPAR provides the precise loading plans of each means of transport required to realise of the transport plan.

The transport plan can start from one or several points on one or several theatres of operation. PROFORCE LEOPAR

The model is divided in three major parts:

- The reference frame;
- The loading module; and
- The planning module.

The reference frame manages all the general concepts which will be used in the various scenarios of loading and planning. In particular, it enables to store information:

- At the air bases, of which distances between bases;
- With the vectors of transport, of which speeds and seats;
- With the equipment to be transported; and
- With the pax to transport.

The loading module manages scenarios corresponding to various transport plans. To build a scenario, useful information from the reference frame for this projection is used to define the quantities of equipment and pax to be deployed. Optimisation proposes the loading of the compartments in order to minimise the number of sorties necessary to the transportation.

The planning module manages scenarios corresponding to various transportation plans. To build a planning scenario, information related to this transportation can be imported from the scenario of the corresponding loading. Optimisation is used to the plan of sorties in order to determine the earliest arrival date for both the materials and the PAX taking into account the deployment and the priorities.

D.4.6.2 Technologies Used

PROFORCE LEOPAR was developed without any optimisation engine. The established algorithms allow the tool to be used without any software package and can be used on all the operating systems. The languages used are the Java language for the User interface and C++ for the coding the algorithms. The data base management system is MySQL. The export files are in HTML format.

The ADAMS database is used to define elements to be transported, and also geographical data is taken from this database. The model is not operationally used, but only for study purposes, mainly acquisition projects. In the future France would like to link ADAMS with PROFORCE LEOPAR.

D.4.7 Helicopter Tool Project (France)

France (DGA) has decided to develop a tool to be used to study Air Mobile studies. The model will be developed in two stages:

- Stage 1 (2003 – 2004): To make a demonstration model (prototype); and
- Stage 2 (2004 – 2005): To make a helicopter model for an air mobile mission.

The development of the model will focus on the following objectives:

- Study purposes: The tool is not intended to be used as an operational tool;
- Helicopter characteristics;
- GIS data;
- Optimization of loadings; and
- Planning and scheduling.

The model will use the following technologies and techniques:

- **Operations research techniques**
  - Algorithms;
  - Heuristics;
  - Linear programming; and
  - Constraint Satisfaction problem.
- **Programming languages**
  - JAVA;
  - C++; and
  - OPL Studio (mathematical language).
- **Database system**
  - Relational databases (Access, MySQL, Interbase, …)

D.4.8 FELPATH (The Netherlands)

FELPATH (FEL Planning Aid for Transport helicopters) is an operationally used decision support system, supporting the combined planning cell of the 11 Air Maneuvre Brigade (consisting of the Tactical Helicopter

Group of the Royal Netherlands Air Force and the 11 Airmobile Brigade of the Royal Netherlands Army) in planning tactical transport helicopter operations.

FELPATH is developed by TNO Physics and Electronics Laboratory (TNO FEL) and includes routeplanning, tasking of transport helicopters and scheduling of air moves. The latest version of FELPATH is V3.2, completed mid 2003.

FELPATH consists of main two modules:

- Routeplanner; and
- Initial Fleet Allocation (IFA).

Before the main modules are entered FELPATH opens with a main screen in order to start an operation. Here general information for a certain operation can be defined, like Air Control Points and Helicopter types. Figure D.13 gives an overview of the helicopter related information that can be defined.

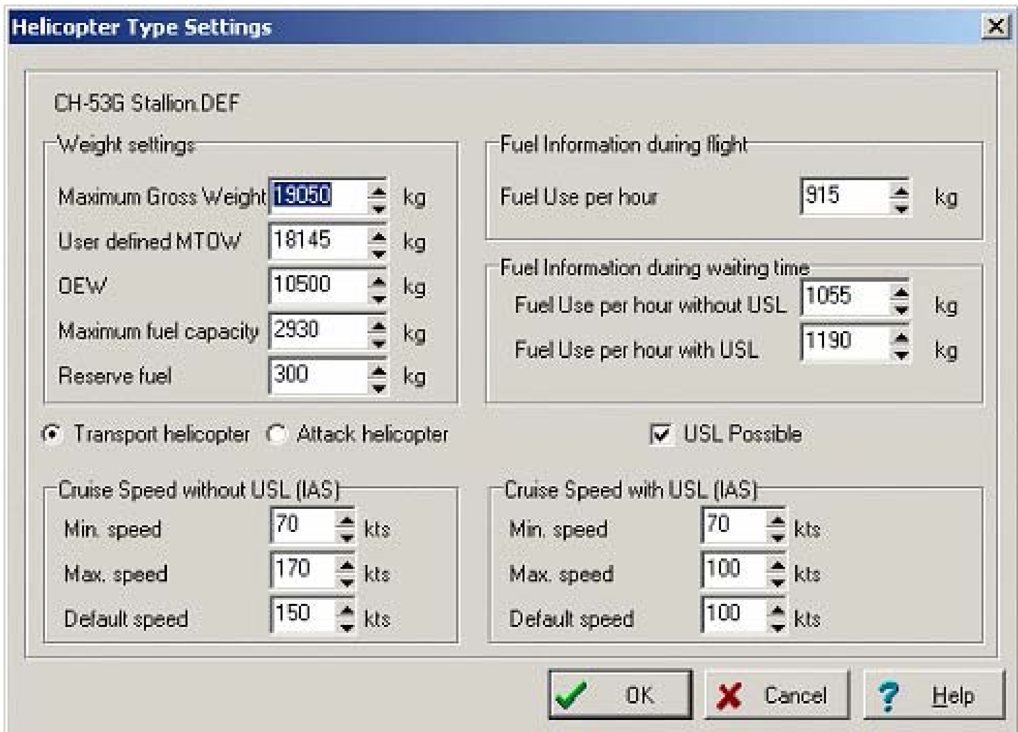


Figure D.13: Helicopter Type Settings Screen.

D.4.8.1 Routeplanner

Figure D.14 shows the main screen of the **Routeplanner** of FELPATH. Using this main screen Routes to be flown by the various types of helicopters can be defined and edited in a digital terrain. A route consists of different types of points, such as Take-off Point, Way Points, Pick-up Points, Landing Points, Fuelling Points and an End Point. Also the area of operation can be defined, zooms can be defined or selected and scanned maps can be used as an overlay.

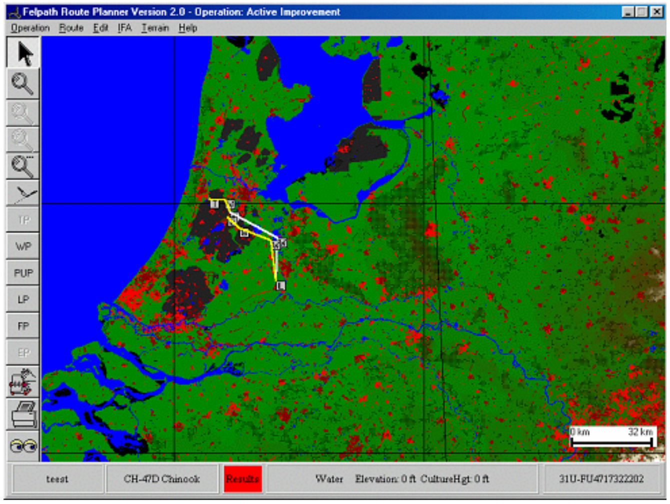


Figure D.14: Main Screen of Routeplanner.

From the route menu the route settings can be defined. This concerns the definition of the route for a certain type of helicopter. Data to be entered ate: flying altitude, atmospheric conditions, like temperature and pressure, but also wind properties as direction and velocity. Then, the route points need to be defined. The general route data may be varied for certain point(s), so for each route point the atmospheric conditions can be changed and a delay time a point may be entered. Some information can also be changed for multiple legs on the route, like flying altitude and speed (IAS or ground speed) and the wind properties. Figure D.15 gives an overview of the dialog.

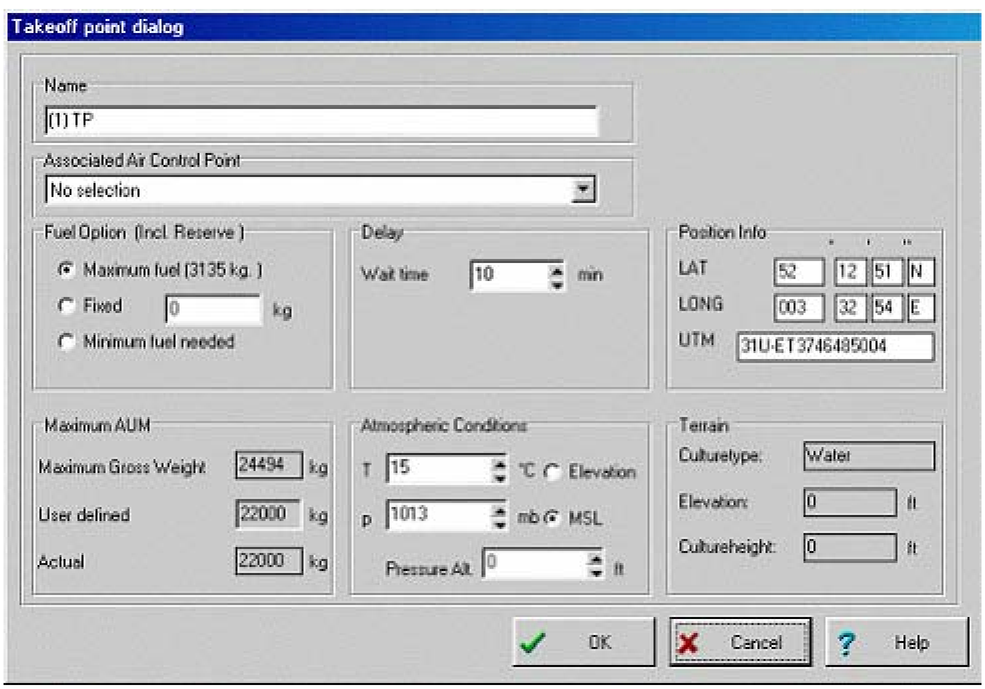


Figure D.15: Take Off Point Dialog.

Once a route has been defined a cross section of the route can be displayed. Figure D.16 gives on overview of the cross section screen.

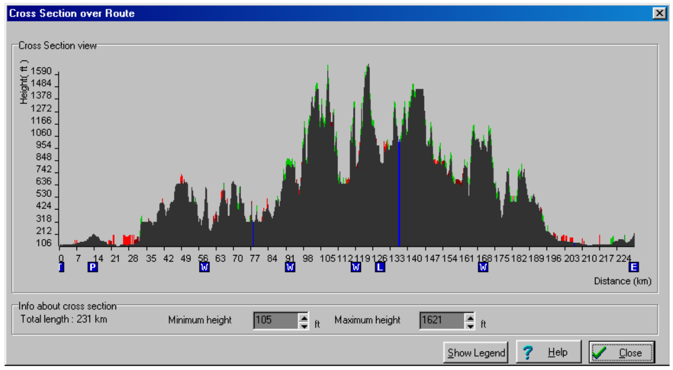
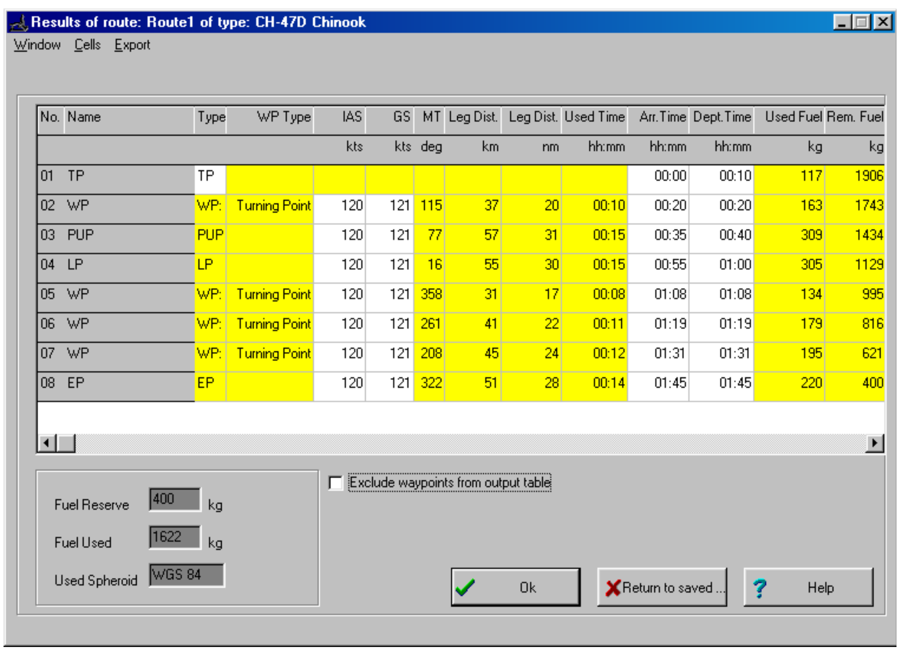


Figure D.16: Cross Section Screen.

The Routeplanner determines the total distance that has to be flown, the time needed, the fuel consumption during the route and the maximum payload that can be transported. Figure D.17 gives an overview of (a part of) the screen showing the results of the route.



No.	Name	Type	WP Type	IAS	GS	MT	Leg Dist.	Leg Dist.	Used Time	Arr. Time	Dept. Time	Used Fuel	Rem. Fuel
				kts	kts	deg	km	nm	hh:mm	hh:mm	hh:mm	kg	kg
01	TP	TP								00:00	00:10	117	1906
02	WP	WP:	Turning Point	120	121	115	37	20	00:10	00:20	00:20	163	1743
03	PUP	PUP		120	121	77	57	31	00:15	00:35	00:40	309	1434
04	LP	LP		120	121	16	55	30	00:15	00:55	01:00	305	1129
05	WP	WP:	Turning Point	120	121	358	31	17	00:08	01:08	01:08	134	995
06	WP	WP:	Turning Point	120	121	261	41	22	00:11	01:19	01:19	179	816
07	WP	WP:	Turning Point	120	121	208	45	24	00:12	01:31	01:31	195	621
08	EP	EP		120	121	322	51	28	00:14	01:45	01:45	220	400

Figure D.17: Results of Route.

The output of the Routeplanner can be exported to a number of files:

- **An Excel file**: all results are exported to an EXCEL document,
    - **A Flight log**: flight log information is exported to an EXCEL document, ready to be printed.

FLIGHTLOG			CH-47D Chinook									
		Date		Time Z/A/B		dtg L-Hour		dtg F-Hour				
		24-7-2001		13:36					Fuel		Wind	
Nr.	Name	IAS	GS	MT	KM	NM	Time	Run	Burned	Rem.	Direction	Speed
1	TP							0:00		3018		
2	WP	120	121	208	5	3	0:01	0:11	24	2995	0	0
3	WP	120	121	208	5	3	0:01	0:23	24	2854	0	0
4	PUP	120	121	101	28	15	0:07	0:40	124	2613	0	0
5	WP	120	121	229	14	7	0:04	0:49	77	2478	0	0
6	LP	120	121	270	11	6	0:03	0:52	63	2414	0	0
7	WP	120	121	53	14	8	0:04	1:01	65	2291	0	0
8	EP	120	121	88	10	6	0:03	1:03	46	2245	0	0
TOTALS					87	47	1:03		Reserve	400		

N.B. De gegevens uit de Flightlog mogen niet gebruikt worden als vervanging van de planning door de vlieger.

Speed in	kts
Direction in	deg
Time in	hh:mm
Fuel in	kg

Figure D.18: Example of a Flight Log.

- **A Co-ordinates sheet**: all information on co-ordinates is exported to an EXCEL document, ready to be printed in a kneepad format (see also Figure D.19).

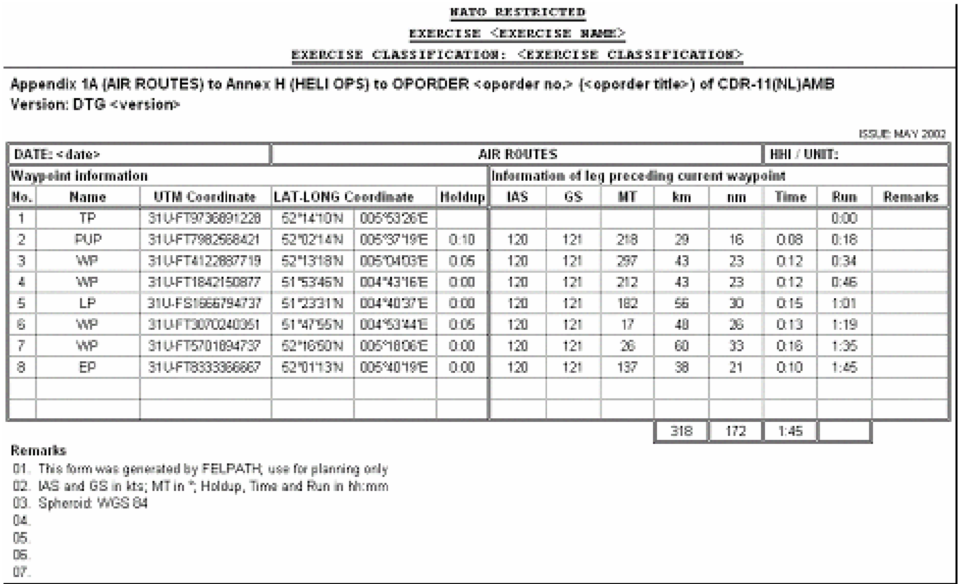
COORDINATES			CH-47D Chinook						
Spheroid		Date	Time Z/A/B		dtg L-Hour		dtg F-Hour		
WGS 84		24-7-2001	13:37				Wind		
Nr.	Name	Zone	Lat	Long	Var	Pa	Temp	Speed	Direction
1	TP	31U-FT2038493208	52°16'34'N	004°45'52'E					
2	WP	31U-FT1795688622	52°14'07'N	004°43'38'E	0	489	15	0	0
3	WP	31U-FT1552884037	52°11'41'N	004°41'25'E	0	486	15	0	0
4	PUP	31U-FT4250178642	52°08'24'N	005°04'56'E	0	491	15	0	0
5	WP	31U-FT3225169830	52°03'48'N	004°55'45'E	0	499	15	0	0
6	LP	31U-FT2110169830	52°03'57'N	004°46'00'E	0	496	15	0	0
7	WP	31U-FT3261078461	52°08'27'N	004°56'16'E	0	495	15	0	0
8	EP	31U-FT4285978821	52°08'29'N	005°05'16'E	0	498	15	0	0

N.B. De gegevens uit de Coordinates mogen niet gebruikt worden als vervanging van de planning door de vlieger.

Pressure in	ft
Speed in	kts
Direction in	deg
Temperature in	°C
Variance in	deg

Figure D.19: Example of a Coordinates Sheet.

- **An Air Routes file**: a combination of all information concerning flight-times and co-ordinates is exported to an EXCEL document (see also Figure D.20).



NATO RESTRICTED

EXERCISE <EXERCISE NAME>

EXERCISE CLASSIFICATION: <EXERCISE CLASSIFICATION>

Appendix 1A (AIR ROUTES) to Annex H (HELI OPS) to OPORDER <oporder no.> (<oporder title>) of CDR-11(NL)AMB
Version: DTG <version>

ISSUE: MAY 2002

DATE: <date>			AIR ROUTES											HHI / UNIT:
Waypoint information						Information of leg preceding current waypoint								
No.	Name	UTM Coordinate	LAT-LONG Coordinate		Holdup	IAS	GS	MT	km	nm	Time	Run	Remarks	
1	TP	31U-FT9736891228	52°14'10"N	005°53'26"E								0:00		
2	PUP	31U-FT7982568421	52°02'14"N	005°37'19"E	0:10	120	121	218	29	16	0:08	0:18		
3	WP	31U-FT4122887719	52°13'18"N	005°04'03"E	0:05	120	121	297	43	23	0:12	0:34		
4	WP	31U-FT1842150877	51°53'46"N	004°43'16"E	0:00	120	121	212	43	23	0:12	0:46		
5	LP	31U-FS1666794737	51°23'31"N	004°40'37"E	0:00	120	121	182	56	30	0:15	1:01		
6	WP	31U-FT3070240351	51°47'55"N	004°53'44"E	0:05	120	121	17	48	26	0:13	1:19		
7	WP	31U-FT5701894737	52°16'50"N	005°18'06"E	0:00	120	121	26	60	33	0:16	1:35		
8	EP	31U-FT8333366867	52°01'13"N	005°40'19"E	0:00	120	121	137	38	21	0:10	1:45		
									318	172	1:45			

Remarks
01. This form was generated by FELPATH; use for planning only
02. IAS and GS in kts; MT in °; Holdup, Time and Run in hh:mm
03. Spheroid: WGS 84
04.
05.
06.
07.

Figure D.20: Example of Air Routes Output.

- **File**: all results are exported to a semicolon-delimited text-file.

The results are also used by the Initial Fleet Allocation (IFA) module. These results can be transferred to the IFA module. Figure D.21 gives an overview of the screen that allows to transfer the results to the IFA.

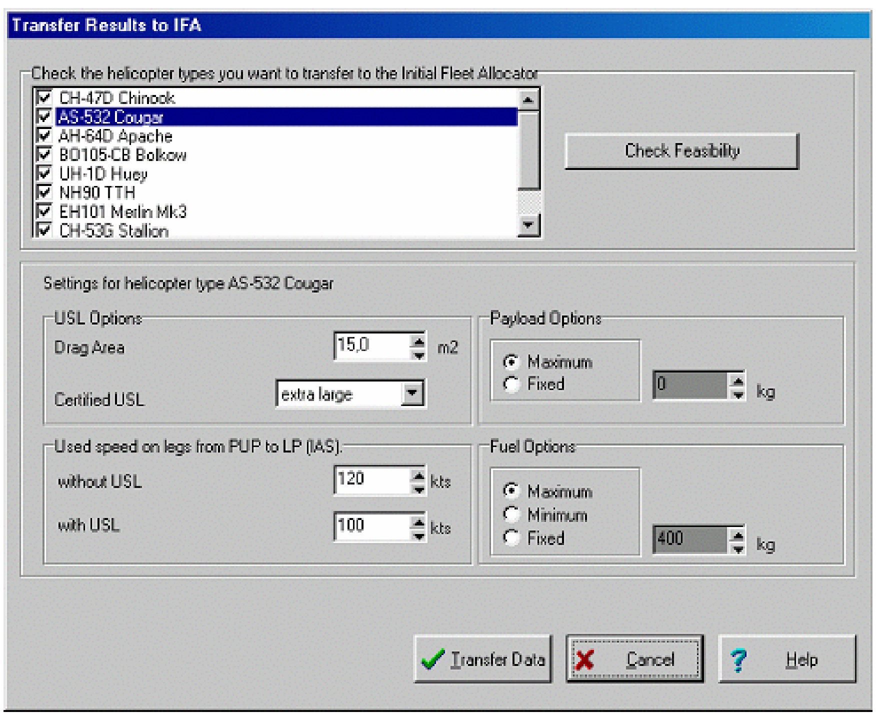


Figure D.21: Transfer Results to IFA.

As the route is planned only for one helicopter type, the route should also be checked for other helicopter types to be used in the IFA. The "check feasibility" button allows this. Clicking the button "Transfer data" will transfer all relevant data to the second module "IFA".

D.4.8.2 Initial Fleet Allocation

The module **Initial Fleet Allocation (IFA)** can be used to make a first estimation, in a quick, simple and rough way, of the number of persons, vehicles and pallets to be transported during an air mobile move. A planning is made of the number of sorties to flown and the number of flights required. Thus this module is used to allocate helicopters from the available fleet to the transport tasks that need to be performed.

The user can easily define the transport task by choosing the units to be transported and the helicopter types to be used. Using this information and the remaining maximum payload, the program produces the number of helicopter sorties required to perform the transport task and schedules them in air moves. The ultimate output of FELPATH is an Air Movement Table.

The AMT consists of information on the movement of the helicopters, like start- and end points and routing times for each flight. Figure D.22 shows the navigation screen of the IFA. It gives on overview of all the elements inside the IFA.

Figure D.22: Navigation Screen IFA.

Helicopter settings: The helicopter settings screen allows the user to enter all the helicopter types that are available to be planned in the IFA. Figure D.23 shows the helicopter settings screen. In this screen the available number of helicopters need to be filled in, Fuel data, loading and unloading times and the load configurations for this helicopter type.

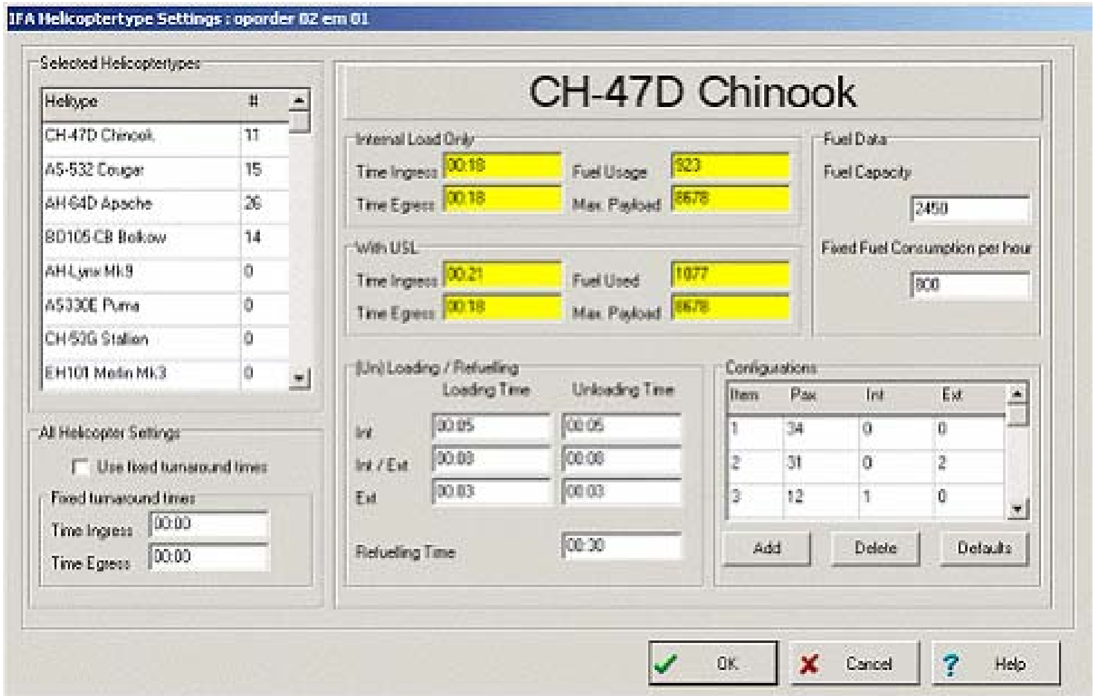


Figure D.23: Helicopter Settings Screen.

Load composition: The load configuration screen is simple, as shown in Figure D.24. Here the number of pax, the pax weight and the number of vehicles and pallets need to be filled in.

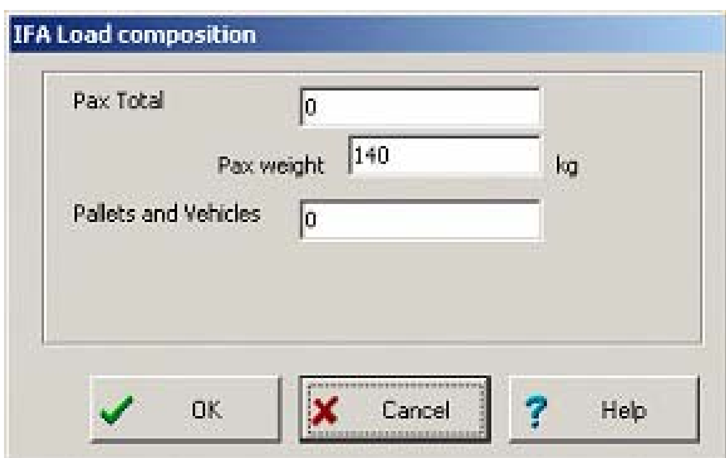


Figure D.24: Load Composition Screen.

Planning: The planning screen is the 'heart' of the IFA. Here the loads are assigned to waves. The user can either do this manually or let the model do this. Figure D.25 gives an overview of this screen after a planning has been done.

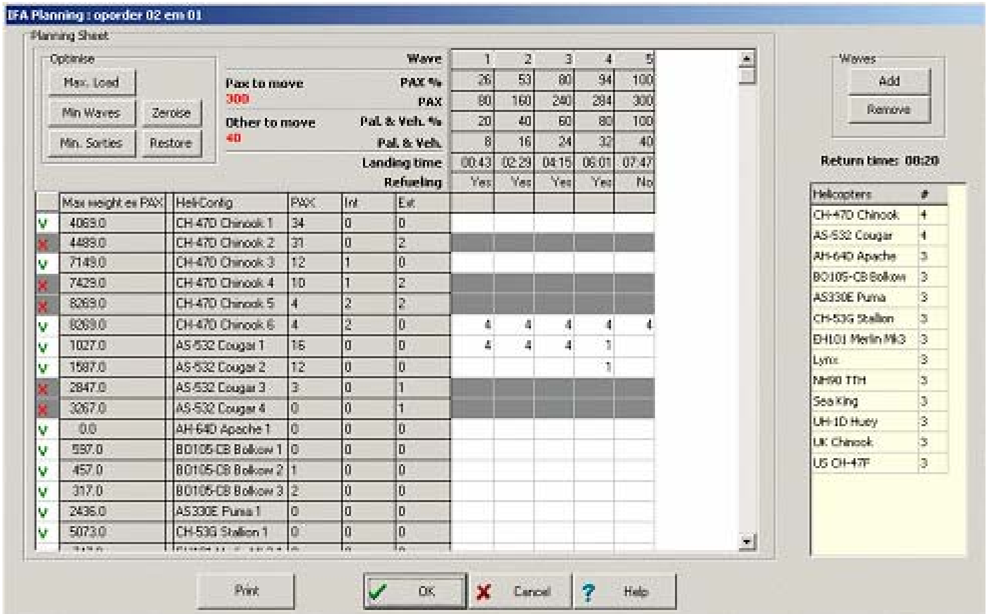


Figure D.25: Planning Screen.

Wave composition: When a planning has been made in sorties, these sorties need to be subdivided in waves. The wave composition screen allows this.

The screen shows the number of sorties for each helicopter and user can manually assign sorties to waves. Figure D.26 gives an overview of the wave composition screen.

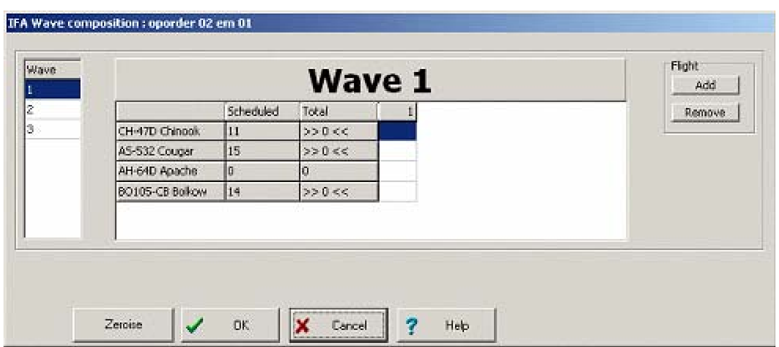


Figure D.26: Wave Composition Screen.

AMT concept: after filling in the first four screens, the planning is almost complete. Now, only some extra data need to be filled in. The AMT-concept screen allows this.

First, the landing times of the wave need to be defined, this can be done related to L-hour or in real time. Figure D.27 shows the results for a landing time in real time. Other data to be filled in, are some delay times like: Time between flights, Time for PUP to WP, Time from WP to LP. Furthermore, some reference data concerning the operation can be filled in, like exercise name, OPORDER nr., version, date, etc.

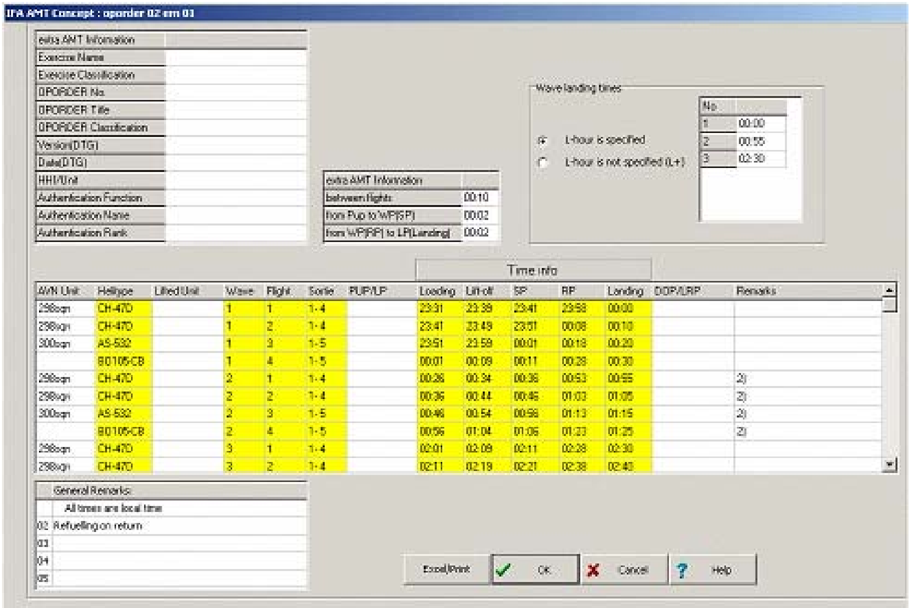
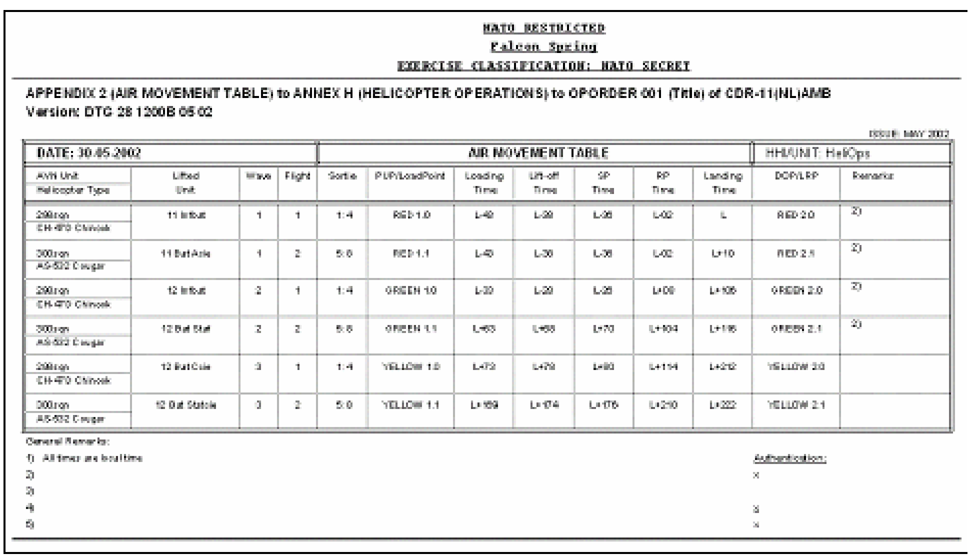


Figure D.27: AMT Concept Screen.

From this screen the final output of FELPATH can be generated. After clicking the Excel/Print button, an Air Movement Table as shown in Figure D.28 will be produced.



NATO RESTRICTED
Falcon Spring
EXERCISE CLASSIFICATION: NATO SECRET

APPENDIX 2 (AIR MOVEMENT TABLE) to ANNEX H (HELICOPTER OPERATIONS) to OPORDER 001 (Title) of CDR-11(NL)AMB
Version: DTG 28 1200B 05 02

ISSUE: MAY 2002

DATE: 30.05.2002				AIR MOVEMENT TABLE							HHI/UNIT: HeliOps	
AVN Unit / Helicopter Type	Lifted Unit	Wave	Flight	Sortie	PUP/LoadPoint	Loading Time	Lift-off Time	SP Time	RP Time	Landing Time	DOP/LRP	Remarks
298sqn / CH-47D Chinook	11 Infbat	1	1	1:4	RED 1.0	L-40	L-38	L-36	L-02	L	RED 2.0	2)
300sqn / AS-532 Cougar	11 Bat Arie	1	2	5:8	RED 1.1	L-40	L-38	L-36	L-02	L+10	RED 2.1	2)
298sqn / CH-47D Chinook	12 Infbat	2	1	1:4	GREEN 1.0	L-30	L-28	L-26	L+08	L+106	GREEN 2.0	2)
300sqn / AS-532 Cougar	12 Bat Stat	2	2	5:8	GREEN 1.1	L+63	L+68	L+70	L+104	L+106	GREEN 2.1	2)
298sqn / CH-47D Chinook	12 Bat Cole	3	1	1:4	YELLOW 1.0	L+72	L+78	L+80	L+114	L+212	YELLOW 2.0	
300sqn / AS-532 Cougar	12 Bat Statole	3	2	5:8	YELLOW 1.1	L+169	L+174	L+176	L+210	L+222	YELLOW 2.1	

General Remarks:
1) All times are local time
2)
3)
4)
5)

Authentication:

Figure D.28: Air Movement Table in Excel.

D.4.8.3 System Requirements

FELPATH can be installed on a stand-alone PC. For the Dutch Armed Forces also a LAN-version has been developed. Using FELPATH v3.2 requires the following minimum system description:

• Processor:	Pentium II 200 MHz
• RAM:	64 MB RAM
• Hard disk:	325 MB free space
• Display resolution:	1024 x 768 pixels
• Peripheral equipment:	CD-ROM player
• Operating system:	MS Windows 95/98/NT
• Required software:	MS Office 97

D.4.9 OTHELLO (The Netherlands)

OTHELLO (Operational Transport HELicopter LOadplanner) is an optimisation tool, supporting at battalion and company level of the 11 Air Mobile Brigade (11LMB) of the Royal Netherlands Army (RNLA) in planning tactical helicopter movements.

One of the results of the brigade-level planning is an Air Movement Table (AMT), which gives an overview of the number of sorties (i.e. movements of one single helicopter) per helicopter type that is required to perform the transport task. The battalions, part of 11LMB, have to work out the brigade plan in more detail. OTHELLO is a software tool that helps the battalion planners to make this detailed plan. The final output of OTHELLO is an Air Loading Table (ALT), as shown in Figure D.29.

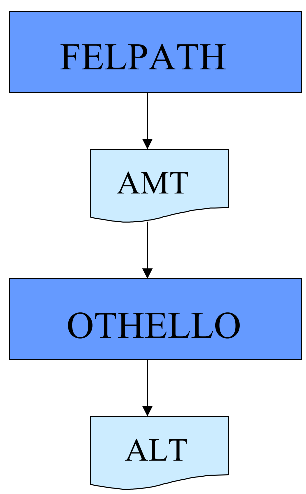


Figure D.29: Relationship between FELPATH and OTHELLO.

OTHELLO produces an ALT, taking into account the following:

- The preferred order in which the load must arrive at the drop off point (priorities of the load);
- Which loads must be transported in the same helicopter sortie (or successive sorties); and
- Which loads have to be transported separately (crossloads).

Data originating from FELPATH, needed to produce an ALT, can be imported automatically.

OTHELLO performs single helicopter planning and for this purpose it only uses certified loads for both internal and external loads. Figure D.30 gives a general overview of the first part of OTHELLO. It uses a PAX and equipment list and a list of internal and external load to produce a list of load configurations.

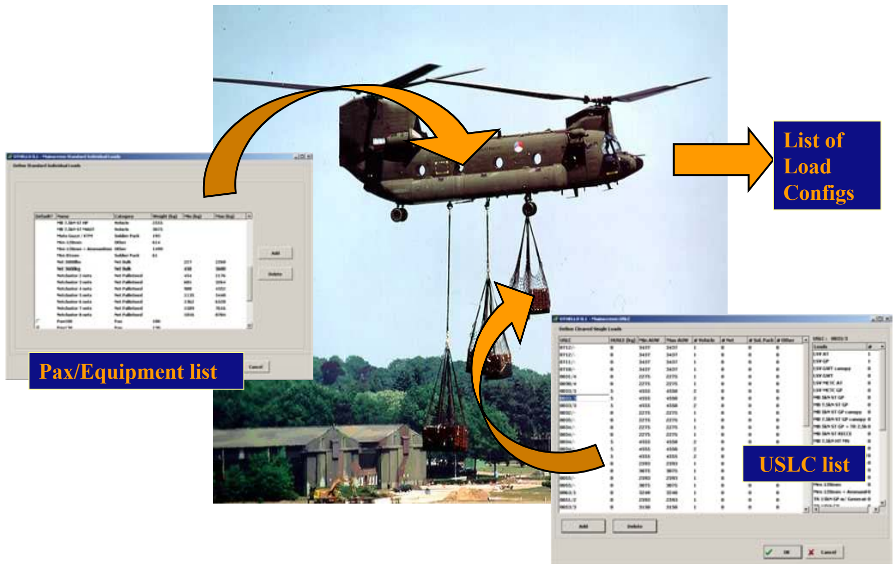


Figure D.30: Overview Determination List of Load Configurations.

Figure D.31 gives an overview of how the load configurations are determined in OTHELLO.

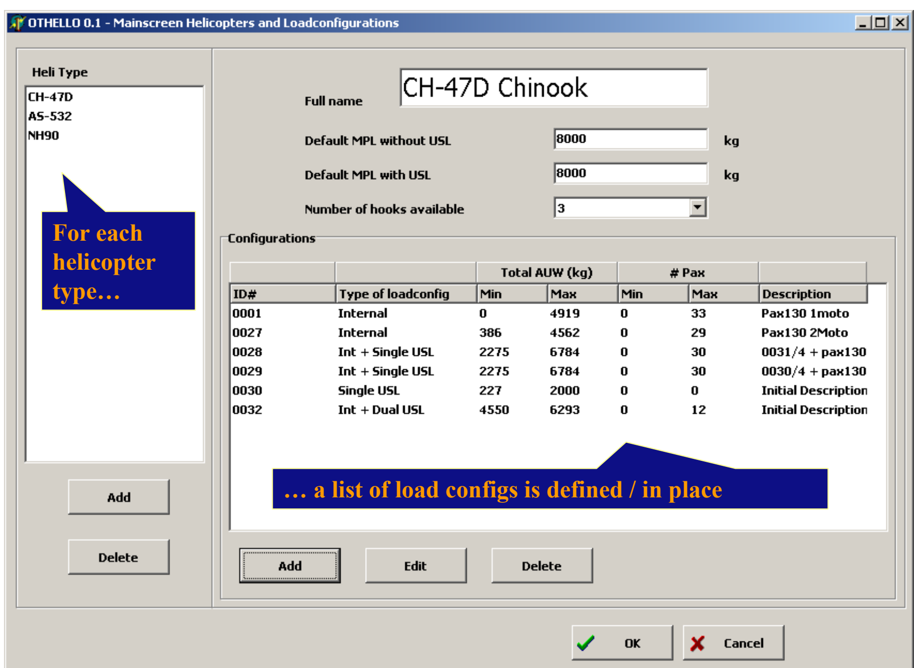


Figure D.31: Determination of Load Configurations in OTHELLO.

OTHELLO enables the planner to easily select the load to be transported by using an electronic database of all units of the 11LMB. The number of vehicles and pax can be modified per group and task forces can be added to the organisation. An overview is given in Figure D.32.

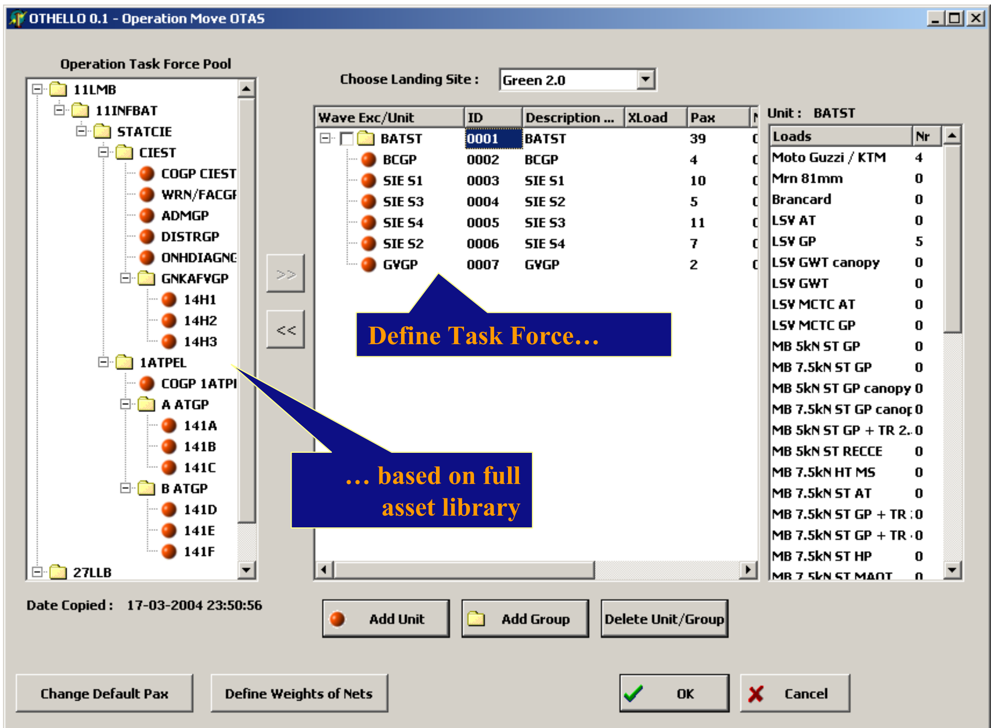


Figure D.32: Definition of Task Force.

Figure D.33 shows the next step, where the actual planning takes place, at single helicopter level.

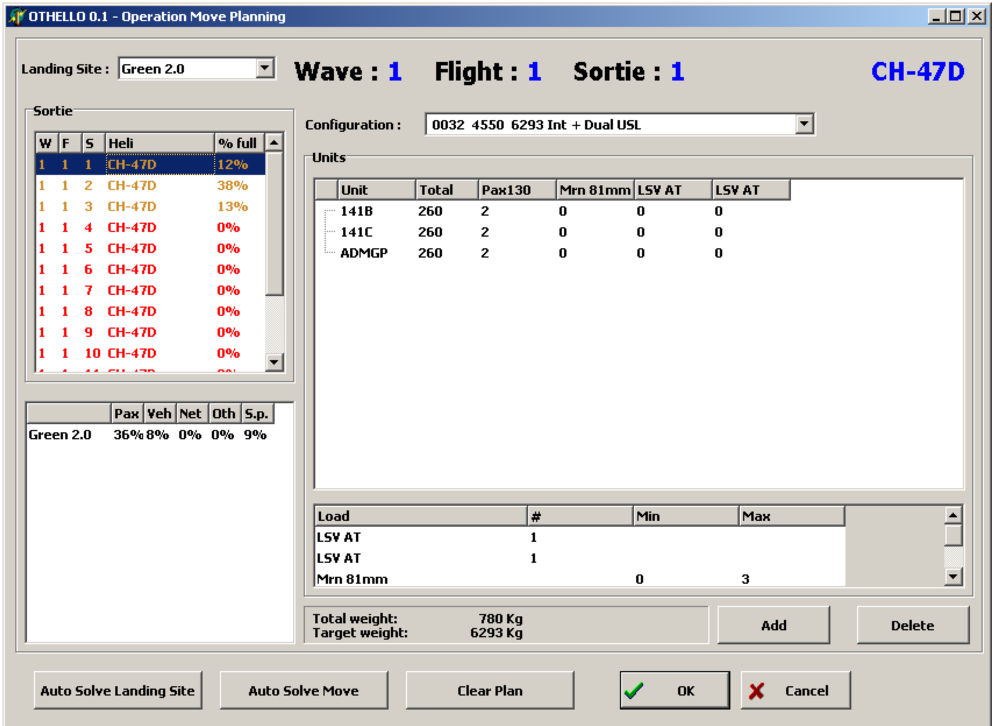


Figure D.33: The Planning in OTHELLO.

The result of the planning is an Air Loading Table (ALT) (see Figure D.34), which gives a detailed description of the exact load that will be transported per sortie, the exact locations where the load will be picked up by the helicopters and the exact arrival and loading times. TNO Physics and Electronics Laboratory (TNO-FEL) has developed a demonstrator version of OTHELLO in 2002. The development of an operational version of the software started in 2003. This leads to a first operational version of the model at the end of 2004.

UITGAVE DTG:..

Page................of................Pages

ISSUE 6

DATE:				AIRLOADING TABLE					HHI/UNIT:	
AVN- Unit/ Type	Wave	Flight	Sortie	Sub-Unit/ Usl	No of Pax/ Usls	AUW (kg)	PUP/ Load point	Arrival time	Loading time	Remarks
(A)	(B)	(B)	(B)	(C)	(D)	(E)	(F)	(G)	(H)	(J)
Voorbeeld 1:										
300/AS	-	-	1	8 pax A/- 8 pax B/-	16/-	1920	FT 12345678	H - 100	H - 55	1)
300/AS	-	-	2	8 pax C/- 8 pax S/-	16/-	1920	FT 12345678	H - 100	H - 55	1)
Voorbeeld 2:										
298/CH	1	1	1	1st pel A/ MB 7.5 kN	25/5410	6075	Red 10	1210	1255	2)
298/CH	1	1	2	2e pel A/ MB 7.5 kN	25/5410	6075	Red 10	1210	1255	2)
298/CH	1	1	3	3e pel A	30	3800	Red 10	1210	1255	
298/CH	1	1	4	Overigen A - coy	30	3600	Red 10	1210	1255	

General Remarks: All times are local times

1) Recceparty

2) 2 mtr extension strop mandatory for Usl

01 jul 000

Figure D.34: Air Loading Table.

Using OTHELLO increases the speed in planning helicopter movements on battalion level. Alternative ALTs can be generated quickly. Also, the possibility of human errors is significantly reduced. Another advantage of OTHELLO is that the optimisation tool only works with certified load configurations for the different types of transport helicopters, which ensures a feasible ALT.

D.4.10 SYNCHROMATRIX (The Netherlands)

As part of the planning process, the 11 Air Manoeuvre Brigade of the Netherlands armed forces build a time schedule for each operation. This time schedule is referred to as the synchronisation matrix, or synchromatrix. The synchromatrix provides a visual representation of all tasks that have to be performed throughout an operation, including when they start and how long they take. This way the people in the planning cell can quickly see if any of the tasks coincide and whether or not all actions are taking place in the right order.

In the past, this synchromatrix was used to be hand-drawn on a white board, which was a very tedious, time-consuming job. Also, changes to the schedule could result in a good part of the synchromatix having to be redrawn. As a result – even though it was acknowledged that the synchromatrix has a clear added value for the operation – drawing up the synchromatrix was often omitted, due to time constraints.

To solve this problem, it was suggested to investigate the feasibility of having a software tool that could 'do the job'. Currently, TNO is developing a software tool that makes it possible to create a synchromatrix electronically. This adds a number of powerful features to the original concept:

- **Convincing visual presentation**. By using the full range of colours available on a computer (instead of the four colours of the white board markers) and presenting the synchromatrix in a recognizable format known in many scheduling programs (i.e. a Gantt-chart), the different tasks and events in an operation can be easily comprehended and communicated.
- **Added intelligence**. The SYNCHROMATRIX software will allow for different tasks and events to be dependent on each other. For example, the task of refueling a helicopter will have to coincide with the event of an activated and open FRP/FARP location. Other dependencies include day/night patterns, weather, and crew availability.
- **Monitoring the operation**. The SYNCHROMATRIX software incorporates a 'time bar' that shows the current time, and also provides pop-up messages whenever an important event is about to start. This way the synchromatrix that is constructed by the PLANS cell can then be used by the CURRENT cell to monitor the operation as it takes place.

The SYNCHROMATRIX software will be the core of an integrated planning system for the 11 Air Manoeuvre Brigade (see also Figure D.35). Other models in this integrated planning system are:

- **FELPATH**. This Brigade-level planning tool for transport helicopters has been used in an operational context for several years. It combines route planning on a geographical map with transport planning. For transport planning, cargo and passengers to be transported within an operation can be distributed over the available helicopters, resulting in a number of moves (*waves*) that have to be flown. Future developments of FELPATH may include planning attack operations and enhancements to digital terrain capabilities.
- **OTHELLO**. The Brigade-level plan for transport helicopters needs to be worked out in detail at Battalion-level. This is the job of OTHELLO. The result is a detailed loading plan for each helicopter sortie in the operation. OTHELLO will be developed in the near future.
- **Squadron planning system**. The last part of the integrated planning system is a squadron-level planning system. FELPATH plans are based on rough helicopter routing estimates. The squadron planning system will verify these routes and specify the routes in more detail. The result would be detailed mission plans, which directly feed into the helicopter-specific mission planning systems. The squadron planning system would be developed by the National Aerospace Laboratory NLR.

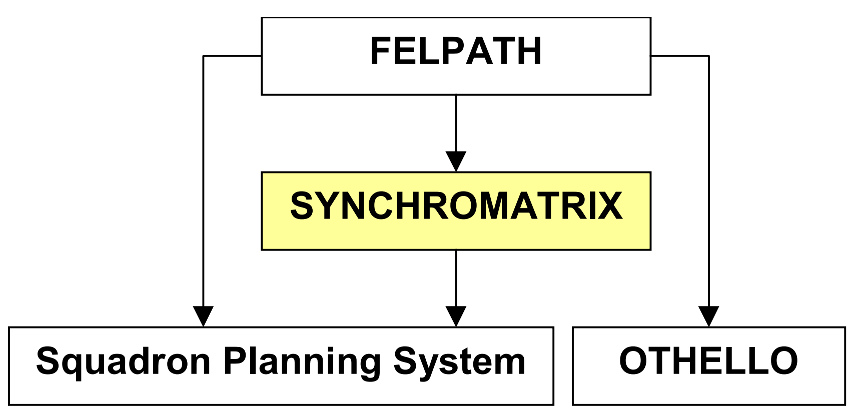


Figure D.35: Overview of Integrated Planning System RNLAF.

TNO has developed a first demonstrator version of the SYNCHROMATRIX software in 2002 (see Figure D.36), which has shown that it will be feasible to build an electronic version of the synchromatrix. A fully operational version of the software will be developed by TNO in the near future.

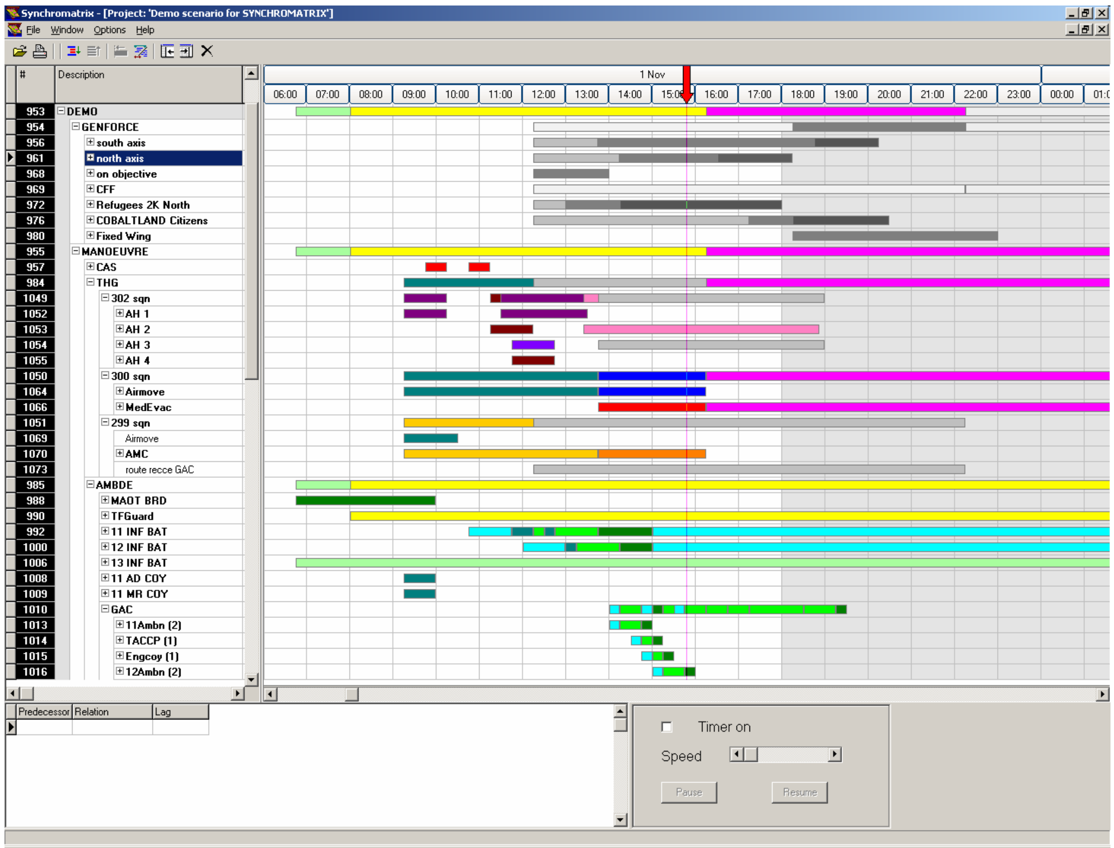


Figure D.36: Demonstration Version of Synchromatrix.

D.4.11 Summary of National Models

In Figure D.37 an overview is given of the national models discussed in this section and their application. For the current operations more national models exist than discussed here. TUAFIS from Turkey is just one example.

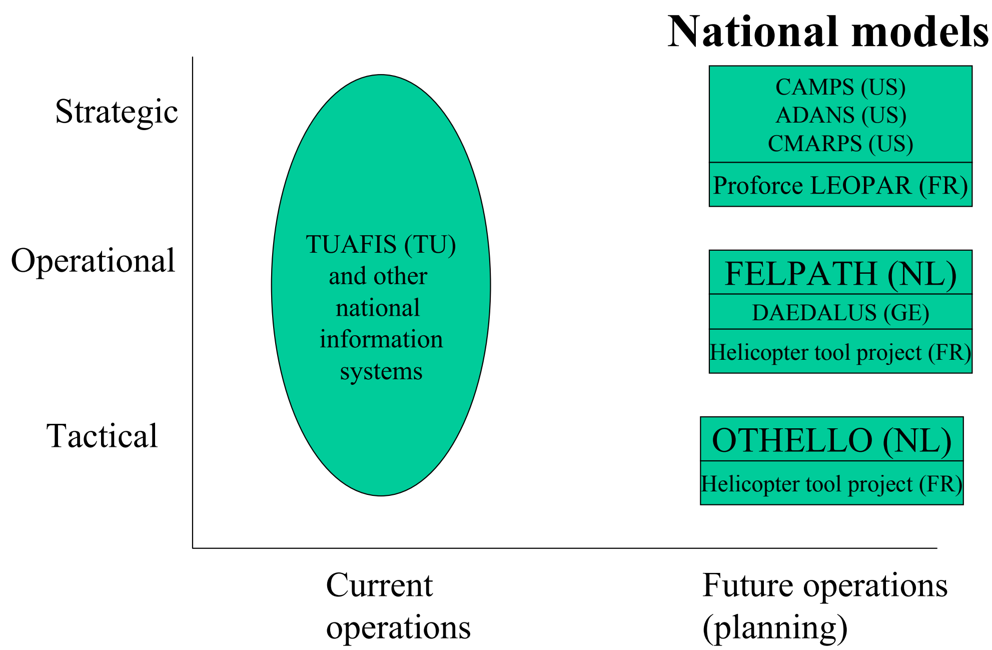


Figure D.37: Overview of National Models.

As the focus of SAS-045 is on the planning of helicopter operations at operational level, only FELPATH, DAEDALUS, the French helicopter tool and OTHELLO are of interest. However, only FELPATH is a operationally used tool. DAEDALUS has not been developed, The French helicopter tool is still to be developed and OTHELLO is under development.

D.5 CONCLUSION

NATO developed several models for planning purposes, but none of them has specifically been developed to support helicopter operations. However, it seems to be sensible that the helicopter model that will be developed within SAS-045 scope will need to use similar software platforms and database that are being used in NATO models like ADAMS.

Within the nations some planning models have been developed. Most of them are developed to support the planning of aircraft. So far only The Netherlands have developed a model called FELPATH that supports the operational planning of helicopters. FELPATH, however supports only a limited number of the functionalities that are required for the new NATO helicopter planning model. That being said, it is rational to use the lessons learned from The Netherlands, when developing the model, and probably use FELPATH as a starting point for developing the decision support tool for planning helicopter operations.

D.6 REFERENCES

[1] TOPFAS Tool for Operational Planning, Force Activation and Simulation, Hakon Thuve, presented at the 6th International Command and Control Research and Technology Symposium, United States Naval Academy (USNA) Annapolis, Maryland, June 19-21, 2001, www.dodccrp.org/6thICCRTS/Cd/Tracks/%20Papers/Track4/127_tr4.pdf

[2] Fanti, L., Airspace Management in the Initial CAOC Capability (ICC), Air C2 Branch NC3A, http://uav.navair.navy.mil/nato/brief/z)%20IV-1/index.htm

[3] Press Statement 22 July 1999, The ACCS Programme & NACMA (NATO Air Command And Control System Management Agency), http://www.nato.int/docu/pr/1999/p990722e.htm

[4] NATO ACCS Overview & Update, Keith Maxwell, NACMA, Briefing for TMD PG 24-25 February 2003.

[5] Jones, Lieutenant Colonel F.K., ADAMS: Can You Get There From Here Without It?, http://www.almc.army.mil/alog/issues/SepOct99/MS470.htm

[6] Heal, G.N., ADAMS System Overview and support to exercises and operations.

[7] TOPFAS Planning Module version 1.1 User's reference Guide.

[8] Solanski, R. and Southworth, F., An execution planning algorithm of military aircraft, Interfaces Vol. 21, No. 4, pp. 121-131.

[9] Rappoport, H., Levy, L., Golden, B. and Feshbach, D., Estimating Loads of Aircraft in Planning for the Military Airlift Command, Interfaces Vol. 21, No. 4, pp. 63-78.

[10] Rappoport, H., Levy, L., Golden, B. and Toussaint, K., A Planning Heuristic for Military Aircraft, Interfaces Vol. 22, No. 3, pp. 73-87.

[11] Global Command and Control System Consolidated Air Mobility Planning System (CAMPS) – system description, http://jitc.fhu.disa.mil/gccsiop/interfaces/camps.htm

[12] Global Command and Control System Air Mobility Command (AMC) Deployment Analysis System (ADANS) – system description, http://jitc.fhu.disa.mil/gccsiop/interfaces/adans.htm

[13] Entwicklung/Bereistelung von planungshilfen für die Einsatzunterstützung luftbeweglichrer Kräfte im Rahmen der Studie Luftbeweglichter Grossverband, Dr. Hättig, R. Hagen, IABG, January 1998.

[14] SAS-045 report on Airspace Management, to be published in 2004.

REPORT DOCUMENTATION PAGE

1. Recipient's Reference	2. Originator's References	3. Further Reference	4. Security Classification of Document
	RTO-TR-SAS-045 AC/323(SAS-045)TP/52	ISBN 978-92-837-0076-0	UNCLASSIFIED/ UNLIMITED

5. Originator	Research and Technology Organisation North Atlantic Treaty Organisation BP 25, F-92201 Neuilly-sur-Seine Cedex, France
6. Title	Computer Based Decision Support Tool for Helicopter Mission Planning in Disaster Relief and Military Operations
7. Presented at/Sponsored by	Final Report of Task Group SAS-045.

8. Author(s)/Editor(s)	9. Date
Multiple	June 2008
10. Author's/Editor's Address	**11. Pages**
Multiple	298

12. Distribution Statement	There are no restrictions on the distribution of this document. Information about the availability of this and other RTO unclassified publications is given on the back cover.

13. Keywords/Descriptors

Aerial delivery	Helicopters	Operations research
Comparison	Military logistics	Performance evaluation
Computer programs	Mission planning	Requirements
Computerized simulation	Mission profiles	Scenarios
Data bases	Models	Transport aircraft
Decision making	Operational effectiveness	

14. Abstract

The SAS-045 study group was formed to propose a framework for a generic and flexible decision support tool that can be used in effective management of helicopter missions both during humanitarian and military operations. This report provides an in-depth discussion of the research process executed by the team and the output of each of those modules. A problem analysis module provides the operational context, mission types, and the decision-making framework. A concept of solutions module presents the mathematical modeling description, the resolution method, and computational results on testing scenarios. Finally, the technical requirements module describes the information management system, database interfacing module, and the protocols and, as well, the information support tool dependencies are defined.

NORTH ATLANTIC TREATY ORGANISATION

RESEARCH AND TECHNOLOGY ORGANISATION

BP 25
F-92201 NEUILLY-SUR-SEINE CEDEX • FRANCE
Télécopie 0(1)55.61.22.99 • E-mail mailbox@rta.nato.int

DIFFUSION DES PUBLICATIONS
RTO NON CLASSIFIEES

Les publications de l'AGARD et de la RTO peuvent parfois être obtenues auprès des centres nationaux de distribution indiqués ci-dessous. Si vous souhaitez recevoir toutes les publications de la RTO, ou simplement celles qui concernent certains Panels, vous pouvez demander d'être inclus soit à titre personnel, soit au nom de votre organisation, sur la liste d'envoi.

Les publications de la RTO et de l'AGARD sont également en vente auprès des agences de vente indiquées ci-dessous.

Les demandes de documents RTO ou AGARD doivent comporter la dénomination « RTO » ou « AGARD » selon le cas, suivi du numéro de série. Des informations analogues, telles que le titre est la date de publication sont souhaitables.

Si vous souhaitez recevoir une notification électronique de la disponibilité des rapports de la RTO au fur et à mesure de leur publication, vous pouvez consulter notre site Web (www.rto.nato.int) et vous abonner à ce service.

CENTRES DE DIFFUSION NATIONAUX

ALLEMAGNE
Streitkräfteamt / Abteilung III
Fachinformationszentrum der Bundeswehr (FIZBw)
Gorch-Fock-Straße 7, D-53229 Bonn

BELGIQUE
Royal High Institute for Defence – KHID/IRSD/RHID
Management of Scientific & Technological Research for Defence, National RTO Coordinator
Royal Military Academy – Campus Renaissance
Renaissancelaan 30, 1000 Bruxelles

CANADA
DSIGRD2 – Bibliothécaire des ressources du savoir
R et D pour la défense Canada
Ministère de la Défense nationale
305, rue Rideau, 9e étage
Ottawa, Ontario K1A 0K2

DANEMARK
Danish Acquisition and Logistics Organization (DALO)
Lautrupbjerg 1-5, 2750 Ballerup

ESPAGNE
SDG TECEN / DGAM
C/ Arturo Soria 289
Madrid 28033

ETATS-UNIS
NASA Center for AeroSpace Information (CASI)
7115 Standard Drive
Hanover, MD 21076-1320

FRANCE
O.N.E.R.A. (ISP)
29, Avenue de la Division Leclerc
BP 72, 92322 Châtillon Cedex

GRECE (Correspondant)
Defence Industry & Research General Directorate, Research Directorate
Fakinos Base Camp, S.T.G. 1020
Holargos, Athens

HONGRIE
Department for Scientific Analysis
Institute of Military Technology
Ministry of Defence
P O Box 26
H-1525 Budapest

ISLANDE
Director of Aviation
c/o Flugrad
Reykjavik

ITALIE
General Secretariat of Defence and National Armaments Directorate
5th Department – Technological Research
Via XX Settembre 123
00187 Roma

LUXEMBOURG
Voir Belgique

NORVEGE
Norwegian Defence Research Establishment
Attn: Biblioteket
P.O. Box 25
NO-2007 Kjeller

PAYS-BAS
Royal Netherlands Military Academy Library
P.O. Box 90.002
4800 PA Breda

POLOGNE
Centralny Ośrodek Naukowej Informacji Wojskowej
Al. Jerozolimskie 97
00-909 Warszawa

PORTUGAL
Estado Maior da Força Aérea
SDFA – Centro de Documentação
Alfragide
P-2720 Amadora

REPUBLIQUE TCHEQUE
LOM PRAHA s. p.
o. z. VTÚLaPVO
Mladoboleslavská 944
PO Box 18
197 21 Praha 9

ROUMANIE
Romanian National Distribution Centre
Armaments Department
9-11, Drumul Taberei Street
Sector 6
061353, Bucharest

ROYAUME-UNI
Dstl Knowledge Services
Information Centre
Building 247
Dstl Porton Down
Salisbury
Wiltshire SP4 0JQ

SLOVENIE
Ministry of Defence
Central Registry for EU and NATO
Vojkova 55
1000 Ljubljana

TURQUIE
Milli Savunma Bakanlığı (MSB)
ARGE ve Teknoloji Dairesi Başkanlığı
06650 Bakanliklar
Ankara

AGENCES DE VENTE

NASA Center for AeroSpace Information (CASI)
7115 Standard Drive
Hanover, MD 21076-1320
ETATS-UNIS

The British Library Document Supply Centre
Boston Spa, Wetherby
West Yorkshire LS23 7BQ
ROYAUME-UNI

Canada Institute for Scientific and Technical Information (CISTI)
National Research Council Acquisitions
Montreal Road, Building M-55
Ottawa K1A 0S2, CANADA

Les demandes de documents RTO ou AGARD doivent comporter la dénomination « RTO » ou « AGARD » selon le cas, suivie du numéro de série (par exemple AGARD-AG-315). Des informations analogues, telles que le titre et la date de publication sont souhaitables. Des références bibliographiques complètes ainsi que des résumés des publications RTO et AGARD figurent dans les journaux suivants :

Scientific and Technical Aerospace Reports (STAR)
STAR peut être consulté en ligne au localisateur de ressources uniformes (URL) suivant: http://www.sti.nasa.gov/Pubs/star/Star.html
STAR est édité par CASI dans le cadre du programme NASA d'information scientifique et technique (STI)
STI Program Office, MS 157A
NASA Langley Research Center
Hampton, Virginia 23681-0001
ETATS-UNIS

Government Reports Announcements & Index (GRA&I)
publié par le National Technical Information Service
Springfield
Virginia 2216
ETATS-UNIS
(accessible également en mode interactif dans la base de données bibliographiques en ligne du NTIS, et sur CD-ROM)

NORTH ATLANTIC TREATY ORGANISATION

BP 25

F-92201 NEUILLY-SUR-SEINE CEDEX • FRANCE

Télécopie 0(1)55.61.22.99 • E-mail mailbox@rta.nato.int

RESEARCH AND TECHNOLOGY ORGANISATION

DISTRIBUTION OF UNCLASSIFIED RTO PUBLICATIONS

AGARD & RTO publications are sometimes available from the National Distribution Centres listed below. If you wish to receive all RTO reports, or just those relating to one or more specific RTO Panels, they may be willing to include you (or your Organisation) in their distribution.

RTO and AGARD reports may also be purchased from the Sales Agencies listed below.

Requests for RTO or AGARD documents should include the word 'RTO' or 'AGARD', as appropriate, followed by the serial number. Collateral information such as title and publication date is desirable.

If you wish to receive electronic notification of RTO reports as they are published, please visit our website (www.rto.nato.int) from where you can register for this service.

NATIONAL DISTRIBUTION CENTRES

BELGIUM
Royal High Institute for Defence – KHID/IRSD/RHID
Management of Scientific & Technological Research for Defence, National RTO Coordinator
Royal Military Academy – Campus Renaissance
Renaissancelaan 30
1000 Brussels

CANADA
DRDKIM2 – Knowledge Resources Librarian
Defence R&D Canada
Department of National Defence
305 Rideau Street, 9th Floor
Ottawa, Ontario K1A 0K2

CZECH REPUBLIC
LOM PRAHA s. p.
o. z. VTÚLaPVO
Mladoboleslavská 944
PO Box 18
197 21 Praha 9

DENMARK
Danish Acquisition and Logistics Organization (DALO)
Lautrupbjerg 1-5
2750 Ballerup

FRANCE
O.N.E.R.A. (ISP)
29, Avenue de la Division Leclerc
BP 72, 92322 Châtillon Cedex

GERMANY
Streitkräfteamt / Abteilung III
Fachinformationszentrum der Bundeswehr (FIZBw)
Gorch-Fock-Straße 7
D-53229 Bonn

GREECE (Point of Contact)
Defence Industry & Research General Directorate
Research Directorate, Fakinos Base Camp
S.T.G. 1020
Holargos, Athens

HUNGARY
Department for Scientific Analysis
Institute of Military Technology
Ministry of Defence
P O Box 26
H-1525 Budapest

ICELAND
Director of Aviation
c/o Flugrad, Reykjavik

ITALY
General Secretariat of Defence and National Armaments Directorate
5th Department – Technological Research
Via XX Settembre 123
00187 Roma

LUXEMBOURG
See Belgium

NETHERLANDS
Royal Netherlands Military Academy Library
P.O. Box 90.002
4800 PA Breda

NORWAY
Norwegian Defence Research Establishment
Attn: Biblioteket
P.O. Box 25
NO-2007 Kjeller

POLAND
Centralny Ośrodek Naukowej Informacji Wojskowej
Al. Jerozolimskie 97
00-909 Warszawa

PORTUGAL
Estado Maior da Força Aérea
SDFA – Centro de Documentação
Alfragide
P-2720 Amadora

ROMANIA
Romanian National Distribution Centre
Armaments Department
9-11, Drumul Taberei Street
Sector 6, 061353, Bucharest

SLOVENIA
Ministry of Defence
Central Registry for EU and NATO
Vojkova 55
1000 Ljubljana

SPAIN
SDG TECEN / DGAM
C/ Arturo Soria 289
Madrid 28033

TURKEY
Milli Savunma Bakanlığı (MSB)
ARGE ve Teknoloji Dairesi Başkanlığı
06650 Bakanliklar – Ankara

UNITED KINGDOM
Dstl Knowledge Services
Information Centre
Building 247
Dstl Porton Down
Salisbury, Wiltshire SP4 0JQ

UNITED STATES
NASA Center for AeroSpace Information (CASI)
7115 Standard Drive
Hanover, MD 21076-1320

SALES AGENCIES

NASA Center for AeroSpace Information (CASI)
7115 Standard Drive
Hanover, MD 21076-1320
UNITED STATES

The British Library Document Supply Centre
Boston Spa, Wetherby
West Yorkshire LS23 7BQ
UNITED KINGDOM

Canada Institute for Scientific and Technical Information (CISTI)
National Research Council Acquisitions
Montreal Road, Building M-55
Ottawa K1A 0S2, CANADA

Requests for RTO or AGARD documents should include the word 'RTO' or 'AGARD', as appropriate, followed by the serial number (for example AGARD-AG-315). Collateral information such as title and publication date is desirable. Full bibliographical references and abstracts of RTO and AGARD publications are given in the following journals:

Scientific and Technical Aerospace Reports (STAR)
STAR is available on-line at the following uniform resource locator: http://www.sti.nasa.gov/Pubs/star/Star.html
STAR is published by CASI for the NASA Scientific and Technical Information (STI) Program
STI Program Office, MS 157A
NASA Langley Research Center
Hampton, Virginia 23681-0001
UNITED STATES

Government Reports Announcements & Index (GRA&I)
published by the National Technical Information Service
Springfield
Virginia 2216
UNITED STATES
(also available online in the NTIS Bibliographic Database or on CD-ROM)

ISBN 978-92-837-0076-0