

Modelling and Simulation of a Map Aided Inertial Navigation Algorithm for Land Vehicles

Tolga Sönmez

Haluk Erdem Bingöl

Navigation Division
TÜBİTAK SAGE
Turkey

tolga.sonmez@tubitak.gov.tr, erdem.bingol@tubitak.gov.tr

ABSTRACT

Sensing the terrain and using it for aiding the inertial navigation system has been widely used in air platforms since the 1960's. TERCOM and SITAN are well-known algorithms for sensing the terrain with a radar altimeter and calculating a correction for the navigation states according to a digital elevation map. In the absence of GPS signals, it is extremely important to be able to make positional fixes. In this paper, we developed a simulation environment for a conceptual application of TAN (Terrain Aided Navigation) for land vehicles. Basically, the test platform is always on the ground so we can assume a trivial zero terrain clearance measurement and apply well-known TAN algorithms. With this new idea, an inertial measurement unit (IMU), a digital elevation map and a barometer is sufficient to apply the TAN algorithms in land vehicles. A widely used navigation aid in land navigation, odometer, is also considered in this TAN application. Error models are developed for each sensor and a dynamic model is used to simulate the IMU data of a land vehicle which moves on the terrain surface. A Kalman filter is designed to track navigation states and as a reference, truth model data is used to find the error statistics of the navigation states via Monte Carlo simulations. This paper also discusses the requirements on the accuracy of sensors and the vehicle capability. The vehicle has to be able to move on a land that has characteristic features for a successful application of the TAN algorithms.

1.0 INTRODUCTION

The purpose of navigation systems is to track position, velocity and the attitude of the platform on which it is mounted. Each sensor that is integrated in the navigation system has advantages and disadvantages. For this reason, different sensors are used to build an optimized navigation system specifically designed for the application. Land navigation is one of these applications where the platform is bound to move on the terrain. Inertial sensors, GPS receiver, pseudolites and odometers are widely used land navigation sensors. In this paper, we propose the use of terrain elevation maps in order to make positional fixes in land navigation systems. The concept of terrain aided navigation is already applied in flying platforms such as missiles, helicopters, UAV and airplanes. Here, we are going to explore the possibility of applying the same concept to land navigation and compare with existing integrated land navigation algorithms.

Modelling and Simulation of a Map Aided Inertial Navigation Algorithm for Land Vehicles

Since the 1930's, inertial sensors are used for navigational purposes. Basically these sensors measure the rotation and acceleration and calculate the navigational variables. Since all the measurements are inertial, it is not susceptible to interference or jamming. As the navigation variables are calculated at a high rate, it is possible to control the platform. But as the error is also summed over during the calculation of position, the uncertainty in the position grows with time unboundedly ([6]).

Satellite based navigation systems have been functional since the beginnings of the 1990's. It is possible to make position and velocity fixes using the signals that are transmitted from the satellites around the world. Among the satellite navigation systems, GPS, GLONASS and Galileo (functional by 2011) are the well known systems. These navigation systems provide a position fix at each instant, so they are widely used to bind the errors of inertial navigation states. INS/GNSS integration can be done in different levels: loosely coupled, tightly coupled and deeply coupled. As satellite based navigation systems depend on receiving electromagnetic signals, they are vulnerable to interference and jamming ([6]).

In order to support inertial navigation system, terrain aided navigation is studied since the 1960's. Basically in this method, clearance from the terrain is measured with a radar altimeter or laser altimeter from an airborne platform flying at a constant altitude. This terrain clearance measurement is compared with DTED (Digital Terrain Elevation Data) maps. If the vertical channel of the INS is not precise enough, a barometric altimeter can be used as a reference. Well-known algorithms are Tercom, Sitan and Terprom ([2], [3], [4]).

In land navigation, the vehicle is required to be on the ground. This constraint provides a trivial terrain clearance of zero magnitude. With this trivial measurement, terrain aided navigation algorithms can be used in making position fixes. We also consider the use of odometer in the land navigation system and show the benefit provided by including it in the navigation system.

To analyze land navigation systems with different sensors, we first develop a simulation environment in Section 2. Using this environment, we can simulate navigation systems under different scenarios. In Section 3, we describe how the simulation data is created according to a scenario. In a scenario, we specify the limitations of land vehicle dynamics and the waypoints that the vehicle passes. In Section 4, error models of different sensors are given in detail. We explain the application of map aided navigation in land vehicles in Section 5. We give the results of our simulations and comparisons in Section 6.

2.0 SIMULATION ENVIRONMENT

In the simulation environment, a realistic movement of the land vehicle has to be imitated. A set of waypoints has to be determined. Selection of waypoints and the terrain, on which the land vehicle is moving, are discussed in section 3. A realistic error model for each sensor in the navigation system has to be determined. The block diagram of the simulation is given in Figure 1.

IMU (Inertial Measurement Unit) data involves 3 accelerometer and gyroscope data. IMU data is formed according to a scenario and vehicle dynamics and in the simulation it is read from a file. Strapdown INS (Inertial Navigation System) algorithm (See [6]) as shown in Figure 2 is applied to calculate navigation states (position, velocity, and attitude). Since the IMU data in the file is errorless, after applying INS algorithms in the upper branch, we get the truth model navigation states. We use the truth model states as a reference. In the lower branch, to make a realistic simulation, we also add an IMU error model before we apply INS algorithm. Here we get the simulated navigations states. Possible corrections from other sensors are also fed into the INS block.

Sensor data is based on truth model states, but the measurements are corrupted by the error model for each sensor. Error model for each sensor is listed in Section 4. Most common sensors for land navigations are: GPS or GNSS, odometer, barometer

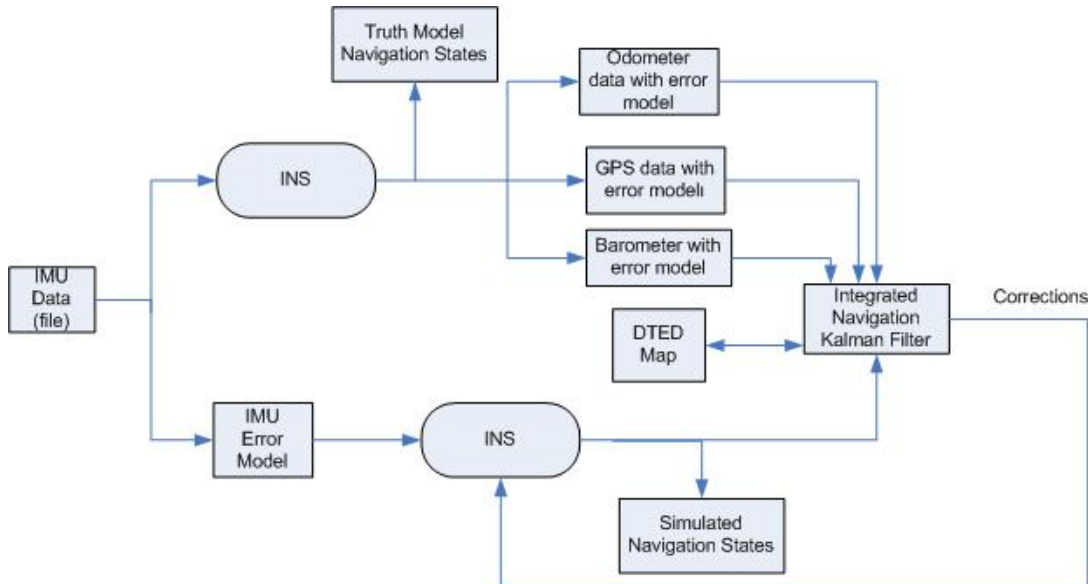


Figure 1: Block diagram for the multi-sensor navigation system simulation.

Each sensor has different characteristics and contributes to the observability of different states. It is crucial to integrate these sensors in an optimal way. Kalman filter is a good choice for this integration with many states. At each update cycle, INS states are corrected according to sensor data in Kalman context. Measurements are compared with Kalman apriori estimates and corrections are calculated. These corrections are used to improve navigation states. This simulation environment can be used to determine and compare the performance of different sensor configurations as the truth model data is available.

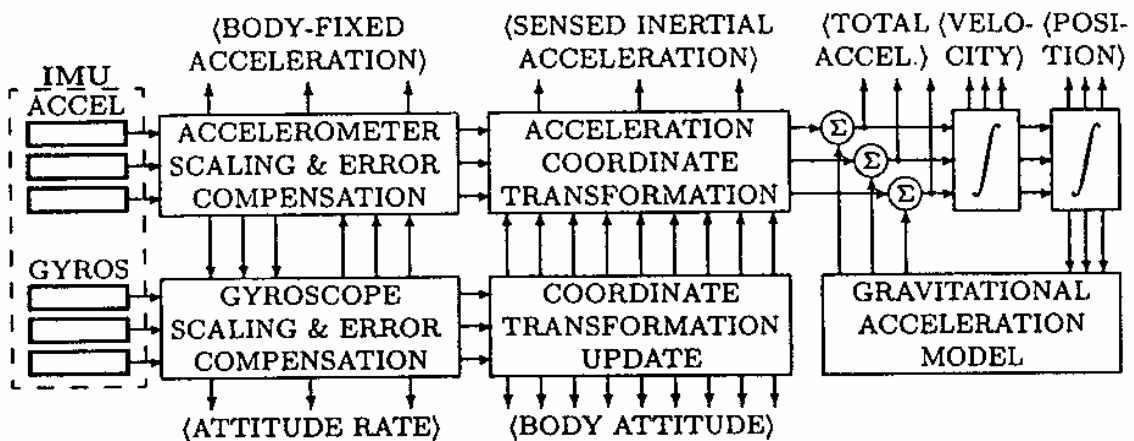


Figure 2 : Strapdown algorithm for inertial navigation system.

3.0 SCENARIO PLANNING, FORMATION OF EXPERIMENTAL DATA

For scenario planning, it is assumed that the land vehicle is moving on the terrain in a lateral direction. The terrain information is taken from a digital elevation map (DTED 2 which has approximately 30 meters resolution). The land vehicle passes through a list of waypoints selected by the scenario planner. The vehicle has a constant velocity (V) in the roll axis direction. The vehicle roll axis is guided to the next waypoint with turns around the yaw axis. Pitch axis turn rate is determined according to the slope of the terrain. Roll axis turn is assumed to be zero since the platform is on the surface. Maximum available turn rate is also specified parametrically according to the capabilities of the land vehicle. Turn rates at each axis specifies C_{bn} (body to NED) coordinate transformation matrix at each time instant. After determining the rotation of the platform, the accelerations in North-East-Down frame are calculated as follows ([1]):

First, we calculate velocity at NED frame ($V_n(k+1)$) at next time step using rotation matrix,

$$V_n(k+1) = C_{bn}(k) \cdot \begin{pmatrix} V \\ 0 \\ 0 \end{pmatrix} \quad (1)$$

Here, we use the already calculated C_{bn} (body to NED) coordinate transformation matrix by the aforementioned method. Secondly we calculate the accelerations in NED frame:

$$\frac{V_n(k+1) - V_n(k)}{\Delta t} = C_{bn} \cdot f_b - (2w_{ie} + w_{en}) \times v_n + g_n \quad (2)$$

Where f_b is the body acceleration, w_{ie} is the earth rotation rate, w_{en} is the navigation frame rotation rate, g_n is the gravitational acceleration and Δt is the discretization time step. Lastly we calculate the accelerations observed by accelerometers in body frame:

$$f_b = \frac{V_n(k+1) - V_n(k)}{\Delta t} + (2w_{ie} + w_{en}) \times v_n - g_n \quad (3)$$

4.0 MODELLING OF SENSORS

Each sensor shown in Figure 1 for the integrated navigation system has to be modeled for a realistic simulation. Error models and measurement characteristics are given below.

Inertial Measurement Unit: IMU is formed by 3 accelerometers and 3 gyroscopes that are strapped on the platform orthogonally. In calculating position and velocity, integration of sensor data is performed; therefore error in the IMU data is also summed and the error in position and velocity grows. This type of systems is called dead reckoning.

Mathematical model for the sensor data is given below:

Gyroscopes:

$$\begin{bmatrix} \delta\omega_x \\ \delta\omega_y \\ \delta\omega_z \end{bmatrix} = B_G + B_g \begin{bmatrix} \alpha_x \\ \alpha_y \\ \alpha_z \end{bmatrix} + S_G \begin{bmatrix} \omega_x \\ \omega_y \\ \omega_z \end{bmatrix} + M_G \begin{bmatrix} \omega_x \\ \omega_y \\ \omega_z \end{bmatrix} + w_G \quad (4)$$

Accelerometers:

$$\begin{bmatrix} \delta \dot{x} \\ \delta \dot{y} \\ \delta \dot{z} \end{bmatrix} = B_A + S_A \begin{bmatrix} \alpha_x \\ \alpha_y \\ \alpha_z \end{bmatrix} + M_A \begin{bmatrix} \alpha_x \\ \alpha_y \\ \alpha_z \end{bmatrix} + w_A \quad (5)$$

Here, $\alpha_x, \alpha_y, \alpha_z$ denotes the acceleration and $\omega_x, \omega_y, \omega_z$ denotes the rotation rate in body axis of the land vehicle. Also B_G and B_A represents gyroscope drift and accelerometer bias, B_g is the dynamic drift matrix in gyroscopes, S_A and S_G are the scale factor error matrix, M_A and M_G are the misalignment matrices, w_A and w_G are the white noise in the sensor data.

There is also correlated noise in bias and drift of the sensors. This can be written as

$$\delta \dot{f}_d^b = -\frac{1}{\tau_a} \delta f_d^b + w_a \quad (6)$$

$$\delta \dot{\omega}_{ib_d}^b = -\frac{1}{\tau_g} \delta \omega_{ib_d}^b + w_g$$

where τ_a and τ_g are the time constants for the random noise. Error in the acceleration/rotation measurement is the difference between the reading and the truth model value in each axis. In the equations b denotes the body coordinate system and i denote the inertial coordinate system. Realistic measurements are formed by adding the aforementioned errors to the true values as follows:

$$\tilde{f}^b = f^b + \delta f^b + \delta f_d^b \quad (7)$$

$$\tilde{\omega}_{ib}^b = \omega_{ib}^b + \delta \omega_{ib}^b + \delta \omega_{ib_d}^b.$$

GPS Receiver: GPS is one of the available global navigation satellite systems. Here, in this simulation we will give the error model for the Global Positioning System but it can be generalized to any GNSS. GPS system enables the user to find its location anywhere that is close to the World. Positioning signals are transmitted by 24 satellites, which are turning around a circular orbit. These signals are received by the GPS receiver and satellite position, distance to user (pseudo-range) and precise timing information is extracted. But the pseudo-range information is distorted by the following (See [5]):

- User clock error (δR_{clk_u}),
- Code loop error (δR_{code}),
- Satellite position error (δr_{sv}),
- Troposphere delay error (δR_{trop}),
- Ionosphere delay error (δR_{ion}).

The pseudo-range is $R_t = |r_{sv} - r_u|$ where r_u is the user position and r_{sv} is the satellite position. In this case, the whole error in the pseudo-range becomes

$$\delta R_t = \delta R_{clk_u} + \delta R_{code} + \delta R_{trop} + \delta R_{ion} + \delta R_{clk_{sv}} + \delta r_{sv} + v,$$

including white noise process v . Now let's combine the similar error terms to get a more compact error expression,

Modelling and Simulation of a Map Aided Inertial Navigation Algorithm for Land Vehicles

$$\delta R_t = \delta R_{clk_u} + \delta R_{sozdeuzaklık} + v \quad (8)$$

The first term includes the receiver clock phase and frequency error and the second term is the combination of other errors. Receiver clock phase and frequency error can be written as:

$$\delta \dot{R}_{clk_u} = \dot{x}_{clk_u} = \begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ 0 & -1/\tau_c \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \quad (9)$$

In the simulation, GPS data set is formed in a block that is fed by true receiver position, receiver antenna attitude and time (See Figure 3). And according to the error model we defined, a realistic error is added in the second block. Finally we achieve raw GPS data satellite pseudo-ranges and satellite position.

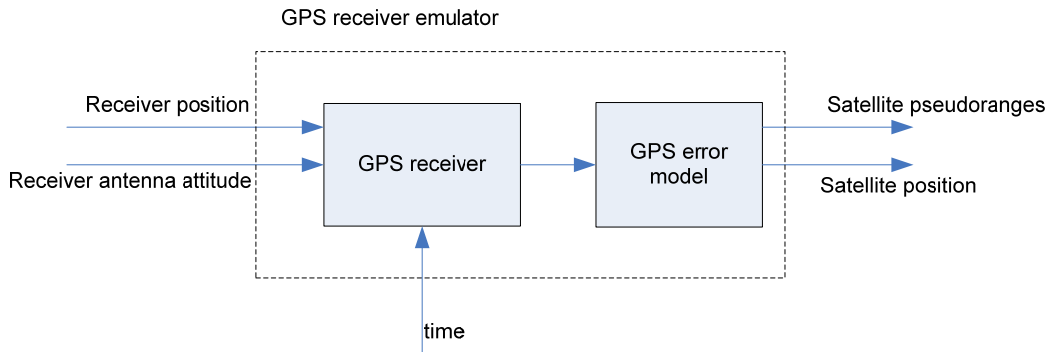


Figure 3: GPS emulator block diagram

Barometric Altimeter: Barometric altimeter provides altitude measurement by comparing the air pressure at the current altitude relative to the sea level air pressure. This altitude measurement from sea level also is also effected from various error sources.

$$h_b = h_a + \delta h_a + \delta h_b + h_a * \delta h_{scf} + v$$

- δh_{pco} : atmosphere change related error
- δh_a : error because of device delay
- δh_b : bias error
- δh_{scf} : scale factor error
- h_b : barometric altimeter measurement
- h_a : true altitude
- v : white noise measurement error

Error model can be expressed in state space form ([7]).

$$\begin{bmatrix} \delta \dot{h}_{pco} \\ \delta \dot{h}_a \\ \delta \dot{h}_{sf} \\ \delta \dot{h}_b \end{bmatrix} = \begin{bmatrix} -1 & 0 & 0 & 0 \\ \tau_{pco} & 1 & -1 & 0 \\ \tau_{dy} & \tau_{dy} & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} \delta h_{pco} \\ \delta h_a \\ \delta h_{sf} \\ \delta h_b \end{bmatrix} + \begin{bmatrix} K_1 / \tau_{pco} \\ 0 \\ 0 \\ 0 \end{bmatrix} \quad (10)$$

Here τ_{dy} is the time constant for device delay, τ_{pco} is the time constant for atmospheric change. In the simulations we used 1sec, and 500 sec for these variables. Bias $1-\sigma$ value is 5m typically, scale factor error $1-\sigma$ value is taken as 0.01.

Altitude measurement from barometric altimeter is used to stabilize the vertical channel of INS.

Odometer: Odometer measures the displacement of a land vehicle. Total displacement can be calculated with the help of a device that is mounted on the front wheels. By measuring the displacement in a unit time, velocity can also be calculated. Error growth of inertial navigation system can be lessened by these velocity measurements. But only the rate of error increase will be lessened as the dead reckoning in position still exists. Odometer is a cost effective sensor. Odometer error model can be expressed as:

$$\begin{aligned} v_{measurement} &= |V| + n \\ n &\approx N(0, \sigma^2) \end{aligned} \quad (11)$$

- n : Velocity reading error, assumed to be Gaussian (m/sn)
- σ : Standard deviation of measurements noise (m/sn)
- $v_{measurement}$: Velocity measurement (m/s)
- $|V|$: True velocity (m/s)

5.0 MAP AIDED INERTIAL NAVIGATION

TERCOM is based on correlating terrain clearance measurements with the barometer and a digital elevation map (DTED). Ideally for correct i and j shifts of the estimated position, the following equation holds,

$$h_{DTED}(L(k) + i\Delta L, \lambda(k) + j\Delta L) = h_b(k) - h_{rad}(k) \quad \text{for } k := 1 \dots N$$

where ΔL is the length of a cell edge, L is the latitude, λ is the longitude, h_b is the barometric height, h_{rad} is the radar altimeter measurement and h_{DTED} is the DTED map terrain height. Realistically, exact values for these variables are not known, so statistical methods are used to correct the estimated position. A profile of altitude data is collected and used in the correlation process.

For land navigation radar altimeter measurements will be trivially zero since the vehicle will always be on the surface of the terrain. With this pseudo measurement we can apply the same TERCOM algorithm in land navigation. In this paper we applied the basic TAN algorithm TERCOM, and showed it has a good

Modelling and Simulation of a Map Aided Inertial Navigation Algorithm for Land Vehicles

performance in our scenarios. For the detailed explanation of TAN algorithms see [2], [3], [4]. For the Kalman filter, a model with 17 states (position, velocity, accelerometer bias, gyroscope drifts, GPC clock bias and frequency) is used.

6.0 SIMULATION RESULTS

In our simulations, the land vehicle has a speed of 30m/sec. The IMU data is simulated according to section 3. The sensor error models are used as explained in section 4. Truth model data is used as reference for error calculation. We used two different types of terrain for performance comparison. We also simulated two versions of vehicle: one with odometer and one without odometer. We also tried the effect of the quality of inertial sensors. High quality inertial sensors have 0.1mg accelerometer and 0.1deg/hr gyroscopes whereas low quality inertial sensors have 1 mg accelerometers and 1deg/hr gyroscopes. 100 Monte Carlo simulations are performed; we analyzed the root mean square error and the number of tracks that converge. Tracks with positional error greater than 300m in any axis are assumed to be diverged.

Firstly, performance of the odometer was evaluated. The following two cases are simulated under the same conditions:

- 1) INS with odometer update for $0 \leq t < 250$ sec, then Tercom and odometer update is applied for $250\text{sec} \leq t \leq 650$ sec,
- 2) Only INS for $0 \leq t < 250$ sec, then Tercom algorithm is applied for $250\text{sec} \leq t \leq 650$ sec.

The histogram of the final CEP errors is shown below.

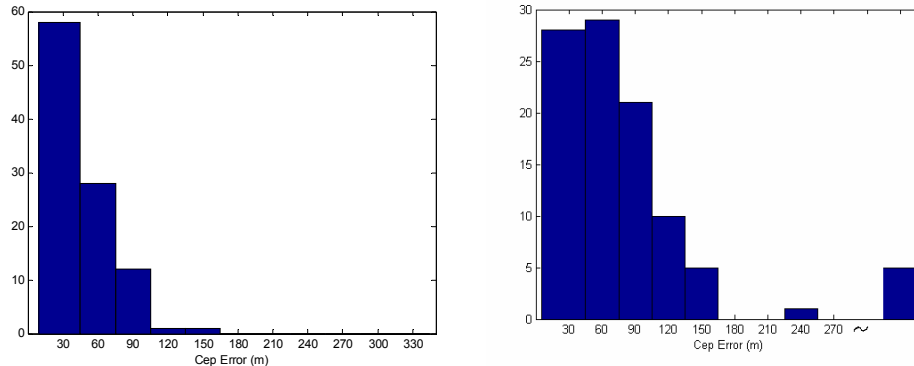


Figure 4- (left) Histogram of the CEP Error with odometer, (right) Histogram of the CEP error without odometer

As seen from the histograms %5 of the solutions were diverged when the odometer is not used. North and east channel errors for convergent solutions are given in Figure 6. The vertical channel is not given as it is trivially computed from DTED map. Though odometer can not make a position fix, error growth rate of INS decreases since the velocity of the platform is measured. As it can be seen from the figures, decrease in error rate by the aid of the odometer helps the correlation process of the Tercom algorithm.

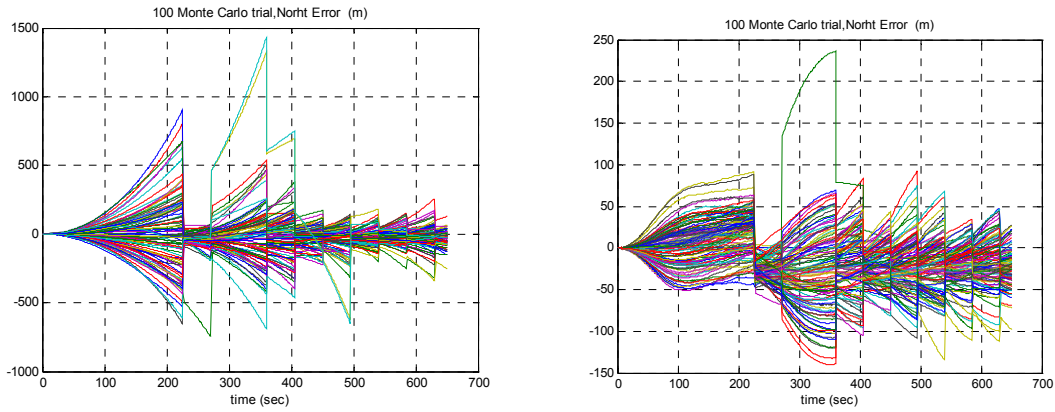


Figure 5: Error in position estimates in north direction for land vehicle (left) without odometer and (right) with odometer.

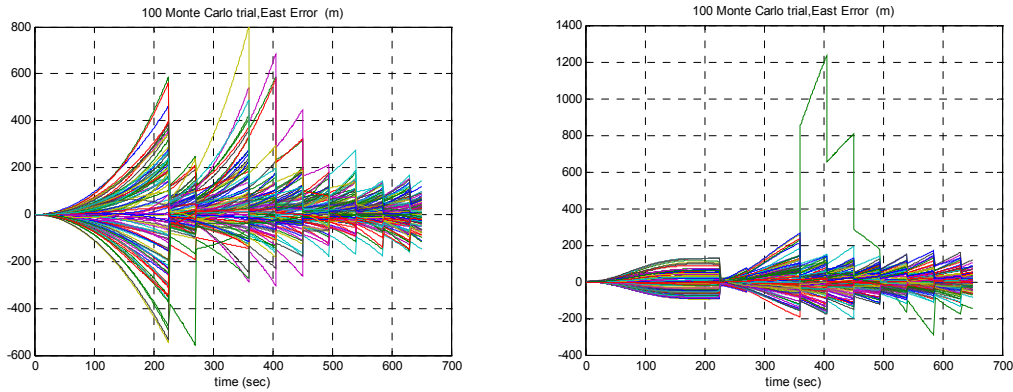


Figure 6: Error in position estimates in east direction for land vehicle (left) without odometer and (right) with odometer.

Secondly two simulations are performed using two different terrains. In these simulations for $0 \leq t \leq 750$ sec Tercom algorithm is applied. Terrain profiles are shown in Figure 7. It can be seen in Figure 8 that pitch angles of the land vehicle does not exceed 30 degrees which is critical for the stability of the platform. According to the simulations, 55 tracks converge for the Terrain 1, and 98 tracks converge for Terrain 2 out of 100 tracks. North errors are shown in Figure 9 (The east error component is neglected as it is similar).

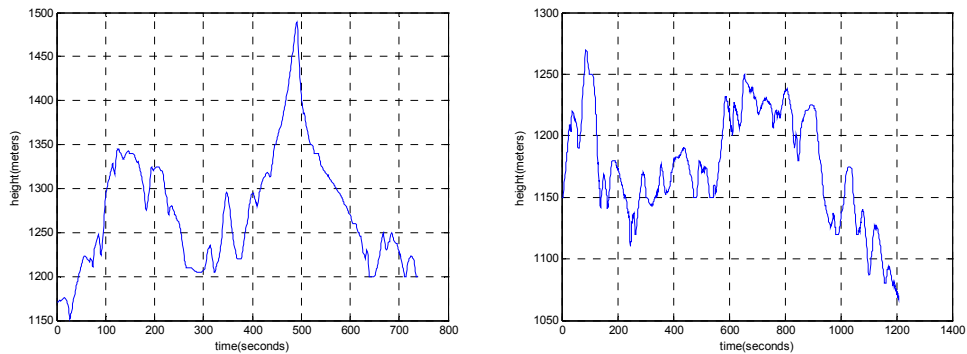


Figure 7: (left) Terrain 1 height profile, (right) Terrain 2 height profile.

Modelling and Simulation of a Map Aided Inertial Navigation Algorithm for Land Vehicles

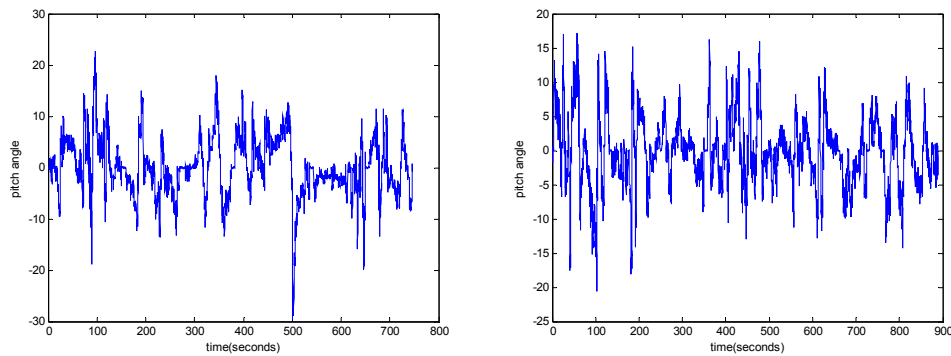


Figure 8: Land vehicle pitch angles for (left) Terrain 1 and (right) Terrain 2.

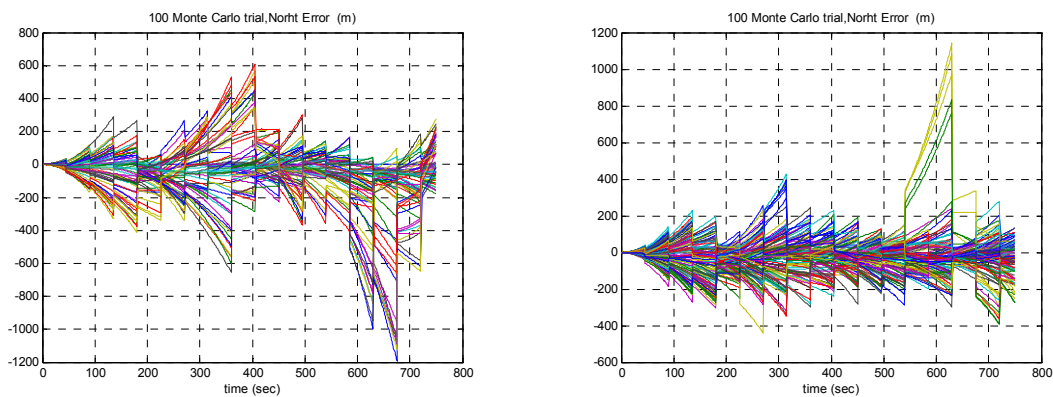


Figure 9: Simulation results for (left) Terrain 1 and (right) Terrain 2.

The roughness of terrain is a critical issue in the application of TAN algorithms. Widely accepted roughness measures for terrain are σ_t and σ_z (for details see [3]). The roughness values for the two terrains along a profile are shown in Table 1.

	σ_t	σ_z	σ_z (area)
Terrain 1	65.80	3.09	1.28
Terrain 2	44.08	2.28	1.97

Table 1- σ_t and σ_z values for Terrain-1 and Terrain-2

The Terrain 2 has lower σ_t and σ_z values which means that it has a lower roughness and a less characteristic terrain with respect to Terrain 1. Checking with Figure 9 and Table 2 we can see that performance is much better in Terrain 2 against the simple intuition from the roughness measures along the profiles. This is because the neighborhood of Terrain 1 bears periodical similarities in two dimensions. When σ_z values are calculated including neighbourhood (σ_z (area)), it is seen that the second terrain has a higher value. This shows that the change in the neighbourhood of the Terrain 2 is more than the Terrain 1.

The simulations are repeated with high quality inertial sensors. It is observed that all of the solutions converge. Corresponding position errors are shown in Figure 10.

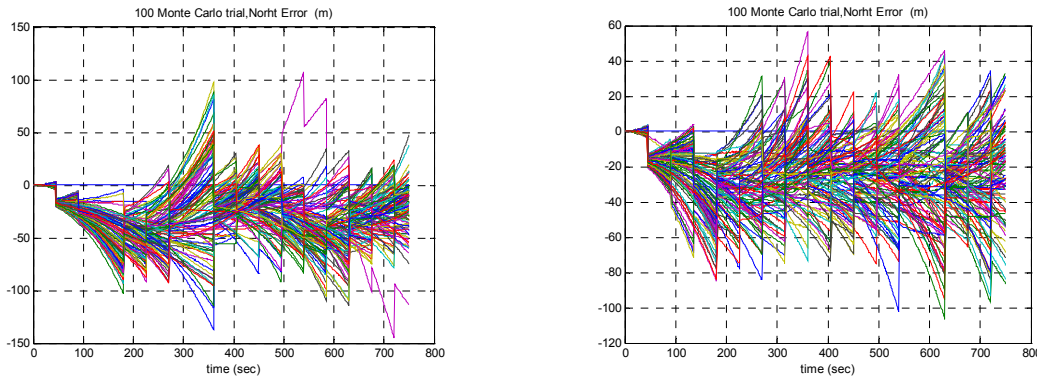


Figure 10: North errors for the navigation systems with (left) low quality and (right) high quality inertial sensors

The summary of the simulation results is shown in Table-2. Here the mean errors are calculated using the convergent solutions. Also, for with and without odometer simulation mean errors are calculated between $250\text{sec} \leq t \leq 650 \text{ sec}$ (when the Tercom algorithm is applied).

	Number of tracks that converge	North Error Mean (m)	East Error Mean (m)	Down Error Mean (m)
With Odometer	100	30.5	41.5	22.5
Without Odometer	94	72.8	56.1	21.2
Terrain-1 with low performance IMU	55	107.4	85.8	10.9
Terrain-2 with low performance IMU	98	65.5	42.4	10.13
Terrain-1 with high performance IMU	100	30.5	8.9	3.16
Terrain-2 with high performance IMU	100	24.7	13.7	2.4

Table 2- Summary of the Simulation Results

7.0 DISCUSSION AND CONCLUSION

In this paper we discuss a conceptual application of well known terrain aided navigation algorithms in land vehicles. These algorithms have been used in airborne platforms since the 1960's. Land vehicles have some further constraints such as they can not move too fast; there can be limitations on the type of terrain they can move. Given that these constraints are satisfied, we modeled a land vehicle movement and simulated a map aided inertial navigation algorithm for this vehicle. We have shown that the vehicle will benefit from the map aided navigation. We also compared the results for a land vehicle with an odometer. With odometer and map aided navigation, the performance is much better. Also the quality of sensors improves the performance quite a lot as it can be seen in Table 2. Terrain roughness is a crucial aspect for the performance of the algorithm but it requires more capable land vehicles in order to bear with the rougher terrain. As a future study, land vehicle capabilities for different types such as tanks, jeeps and etc. have to be studied and the feasibility of map aided inertial navigation should be further investigated.

Modelling and Simulation of a Map Aided Inertial Navigation Algorithm for Land Vehicles

REFERENCES

- [1] Zdislaw H. Lewantowicz “Deep Integration of GPS,INS SAR and Other Sensor Information” *AGARD Aerospace Navigation Systems*, June 1995 s.231.
- [2] Larry D. Hostetler, “Nonlinear Kalman Filtering Techniques for Terrain Aided Navigation” *IEEE Tran. On AC Vol. AC-28, No:3*, 1983.
- [3] George M. Siouris, “Missile Guidance and Control Systems”, Chap 7, Springer 2004.
- [4] Joe P. Golden, “Terrain Contour Matching: a Cruise Missile Guidance Aid” *SPIE Vol. 238 Image Processing for Missile Guidance* 1980.
- [5] E. Günaydin, P. Oguz, T.Sonmez, “Implementing a GPS Emulator for Integrated Navigation Systems” *AIAC Conference*, 2005.
- [6] Grewal, M.S., Weil, L.R., Andrews, A. “*Positioning Systems, Inertial Navigation, Integration*”, John Wiley&Sons, Inc., 2001.
- [7] Boasman N., Clarke D., Davison S., Stokes R., “*Advanced Test Methods for Inertial Navigation Systems*”, Royal Institute of Navigation, 2005.