U.S. ARMY RDECOM
ARL RESEARCH LABORATORY

U.S. ARMY RESEARCH, DEVELOPMENT AND ENGINEERING COMMAND

Physics-inspired placement of analytics services on heterogeneous resources for multisensor fusion

Brent Kraczek

US. Army Research Laboratory, Computational and Information Sciences Directorate

2018//11/06

THE TEAM

Brent Kraczek and Kevin Chan, US Army Research Lab

Theodoros Salonidis and Bongjun Ko, IBM Research

Prithwish Basu and Ali Sydney, Raytheon/BBN

Tom LaPorta, Pennsylvania State U

Funded as part of ARL DAIS-ITA Initial Program Plan, with additional support by ARL mission funds.

PROBLEM: DATA VOLUME

Current flood of data is only the beginning:

- Commercial side:
  Explosion in data collected
  (estimate: 44 billion cameras in use by 2020)

- Tactical Side:
  Future battlefields to contain millions of networked devices (Internet of Battlefield Things—IoBT)

Analyzing data from multitude of sensors will be essential to understand future battlefield from situational awareness to global strategy.

NEED FOR EDGE COMPUTING

Cloud computing poses difficulty for IoBT:

- Data transmission is costly in terms of time and energy
- Network is likely to be unreliable due to congestion, jamming, damage, etc.
- Centralized infrastructure creates potential points of failure, reducing system robustness.

Edge computing:

- Places hardware for analysis at or near sensors.
- Performs analysis as close to data as possible.
- Curates data to be sent upward, reducing network traffic.
- Reduces latency in responses to changes in device status.

MODELS OF EDGE COMPUTING

Different models of edge computing:

- Traditional “Edge Computing”:
  Analysis performed on-device:
  - \+ Minimal information needs to be transmitted
  - \- Requires sufficient processing power on-device (higher costs)
  - \- Data fusion accesses data from multiple sources

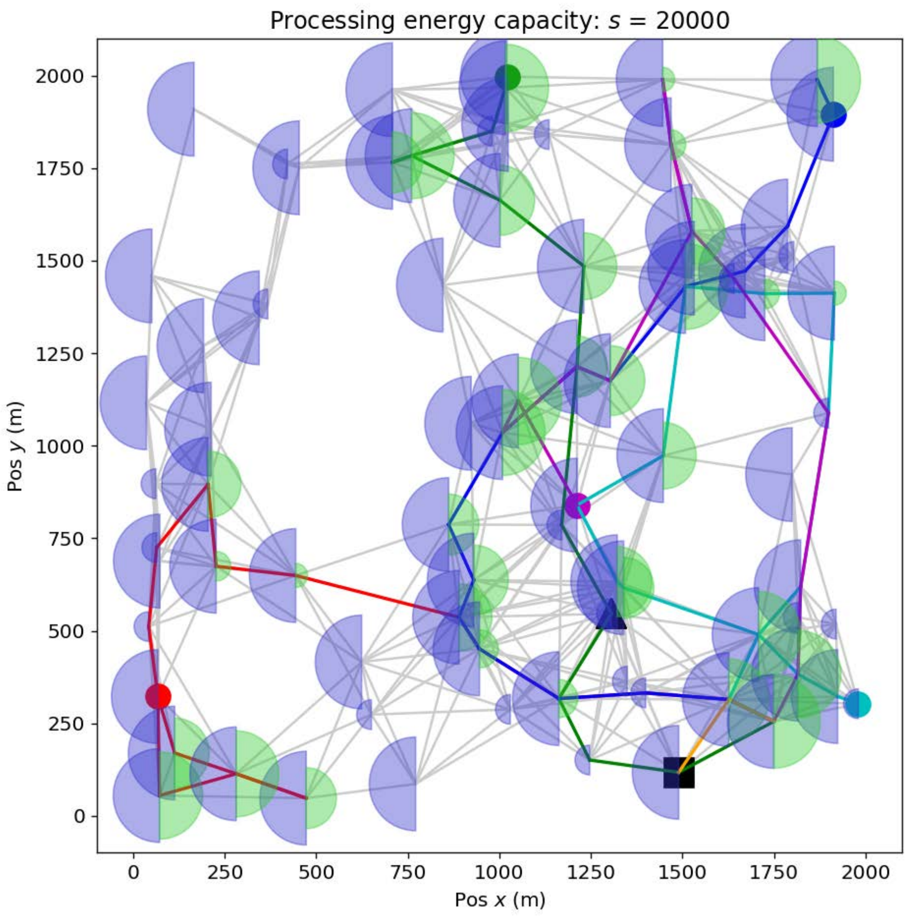
- “Fog Computing”:
  Network contains extra computing power, e.g. at network routers
  - \+ Individual devices can be inexpensive
  - \- Network routers are potential points of failure

- **Distributed model:**
  Many or all devices have some routing and computing capability.
  Set of devices act as a distributed computer
  Individual devices still inexpensive, fewer points of failure

TASK: MACHINE-OPTIMIZED COORDINATION OF ANALYTICS DISTRIBUTED ON SENSOR NETWORK

Use the network of devices to create sufficient computing power

- Distributed edge computing model
- Take advantage of each computing device on-network
- Optimization seeks to minimize energy costs, while simultaneously balancing energy usage.
- Handle constraints on available energy, RAM, etc.



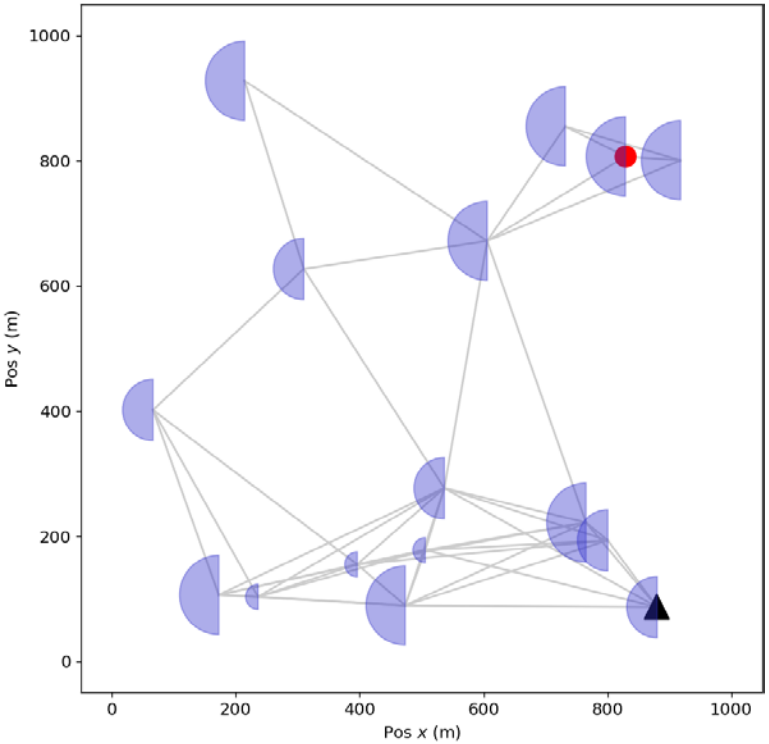
EXAMPLE: RANDOMLY DISTRIBUTED SENSOR NETWORK

Example:

- System of networked security cameras/sensors
- Physical nodes occupy geometric random graph on square kilometer.

- Each device consists of:
  - Sensor
  - Cell-phone-class processing/memory
  - Battery
  - Solar Cell
  - Radio (e.g. Wi-Fi) for communications

- Task:
  - Distribute analytics over devices
  - Stay within capacities of devices



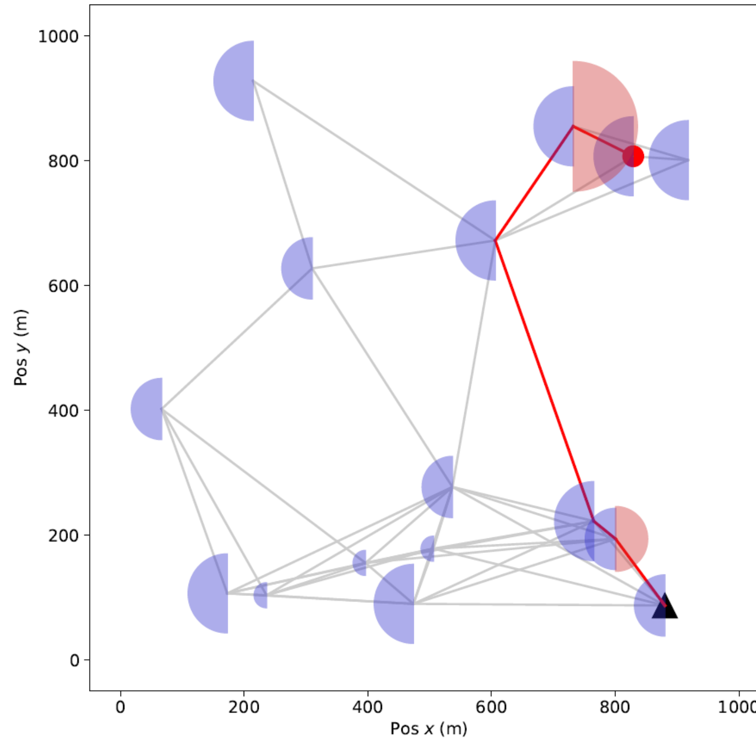
16 networked IoT devices on square km with varying capacities. Available communications links based on measured 802.11ac performance.

PROBLEM SET-UP: GRAPH EMBEDDING PROBLEM

Treat as optimal graph embedding problem (OGEB):

- Source (red circle) + observer (black triangle)
- Seek to "optimally" place analytics:
  - Minimize communications
  - Balance load on system
- Seek distributed solution (avoid single points of failure)

WARNING: OGEP is generally intractable.



Initial placement of analytics, based on minimum communications path. Blue semicircles for capacity, red semicircles for over-utilization.

RESOURCES: PHYSICAL GRAPH

Physical graph: Available resources

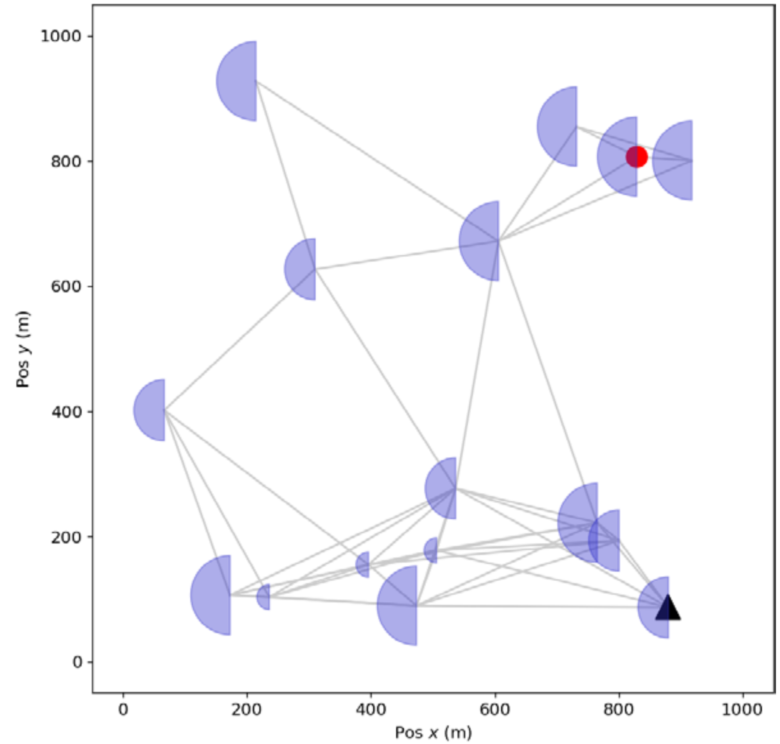
$$\mathcal{H} = (V_P, E_P, \underbrace{Q, R}, \omega_P^E)$$

Nodes (physical devices)

Edges (available communications)

System capacities (processing energy and RAM)

Edge communi-cations energy costs



Physical graph composed of networked devices. Connections set by wireless communications. Left-semicircles represent available capacity.

ANALYTICS: LOGICAL GRAPH

Logical graph: Analytics requirements

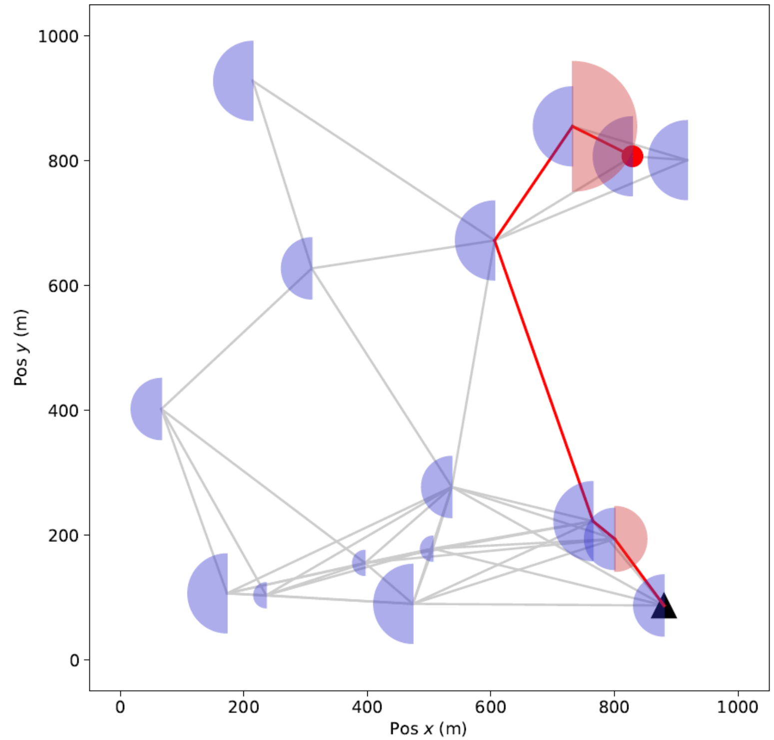
$$\mathcal{G} = (V_A, E_A, \underbrace{q, r}, \omega_A^E)$$

Nodes (analytics stages)

Edges (data connections between nodes)

System requirements (processing energy and RAM)

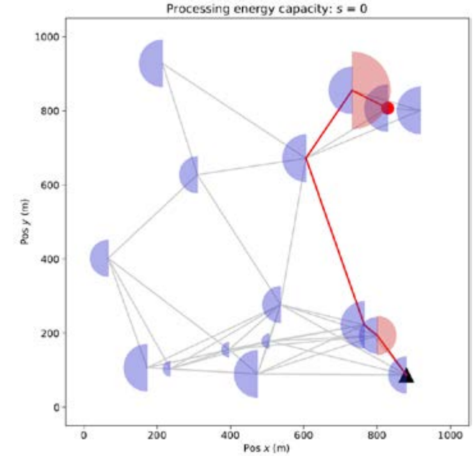
Edge communi-cations requirements

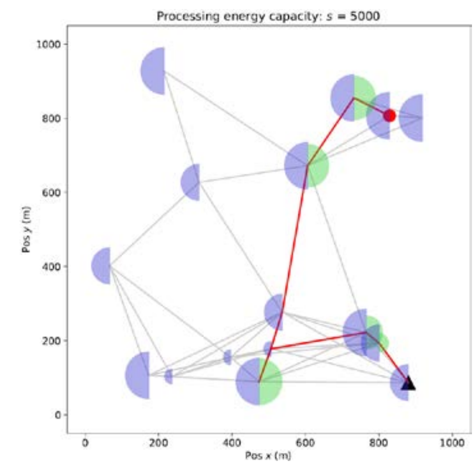


Logical graph: Right semi-circles show system requirements of analytics; heavy red line shows communications path

OPTIMIZATION APPROACH: ADAPT DISTRIBUTED MOLECULAR MODELING ALGORITHMS

- Treat nodes of physical and logical graphs particles, similar to particles in molecular simulations.
- Use physics-inspired interactions to obtain objective function.
- Optimize using Metropolis Monte Carlo + Simulated Annealing to find optimal placements.
- Placements are not global optima, but typically quite good and fast.





OPTIMIZATION PROBLEM: OBJECTIVE FUNCTION 1

Compute placements $\pi: V_A \rightarrow V_P$, the mapping of logical nodes to physical nodes, that minimizes

$$\Phi = \Phi_{\mathrm{comms}} + c_{\mathrm{RAM}} \Phi_{\mathrm{RAM}} + c_{\mathrm{proc}} \Phi_{\mathrm{proc}}$$

where c_{RAM} and c_{proc} are adjustable parameters,

Φ_{comms} is actual energy cost of communications,

$$\Phi_{\mathrm{comms}} = E^{\mathrm{comms}} \,,$$

$$E^{\mathrm{comms}} = \sum_{(u,v) \in E_A} \omega_A^E (u,v)\, \omega_P^E (\pi(u), \pi(v))$$

OPTIMIZATION PROBLEM: OBJECTIVE FUNCTION 2

Φ_{RAM} is a penalty function (soft constraint) to ensure RAM capacity is not exceeded

$$\Phi_{\text{RAM}} = \sum_{z \in V_P} \mathbb{1}\left\{R_z + \sum_{u \in V_A} r_u \mathbb{1}\left\{\pi(u) = z\right\} > 0\right\}$$

and Φ_{proc} is a physics-inspired, Coulomb-like potential

$$\Phi_{\text{proc}} = \sum_{z \in V_P} \sum_{u \in V_A} Q_z q_u \; f\left(z, \pi(u)\right) + \sum_{(u,v) \in V_A \times V_A} q_u q_v \; f\left(\pi(u), \pi(v)\right)$$

Φ_{proc} serves dual role:

- Acts as soft constraint on processing energy limits.
- Balances utilization.

U.S. ARMY RDECOM ARL RESEARCH LABORATORY

OPTIMIZATION ALGORITHM: METROPOLIS MONTE CARLO + SIMULATED ANNEALING

Algorithm:

Init:

1. Set physical graph
2. Embed logical graph
3. Set cooling schedule $\tau(s)$
4. Compute initial Φ

Iterate for s=1 to $s_{\max}$:

1. Trial move (randomly move one node of logical graph)
2. Compute $\Delta\Phi$.
3. If $\Delta\Phi < 0$: Accept trial move
4. Else if $e^{-\Delta\Phi/\tau(s)} > \mathrm{rand}(0,1)$
   Accept trial move
5. Else: Reject trial move

Comments:

- Φ is additive, so only $\Delta\Phi$ needs to be computed.
- $\Delta\Phi$ can be computed at any physical node with only local information of other nodes
- Parameters c_{RAM} and c_{proc} and $\tau(s)$ depend on physical and logical graph—can be preset.

BALANCED UTILIZATION INDEX

Local final capacity given by

initial capacity (“negative charge”) + utilization (“positive charge”)

$$\bar{Q}_z = Q_z + \sum_{u \in V_A} q_u \mathbb{1}\{\pi(u) = z\}$$

Use shift to ensure quantities are non-negative

$$Q'_z = \bar{Q}_z - Q_{\min}$$

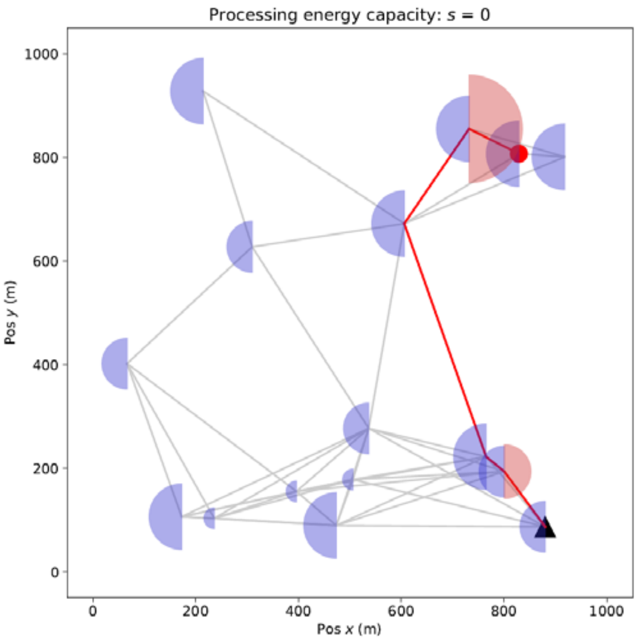
Balanced Utilization Index (modification of Jain’s fairness index) measures balance of final capacity:

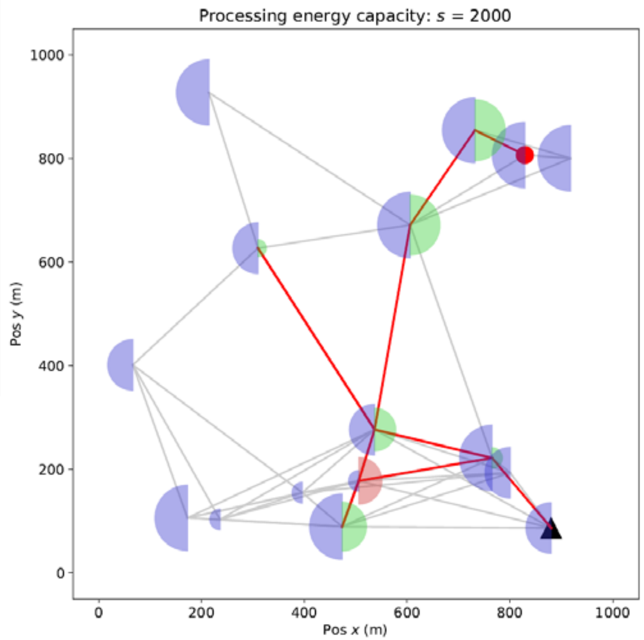
$$I_{BU} = \frac{\left(\sum_{z=1}^{n} Q'_z\right)^2}{n \cdot \sum_{z=1}^{n} \left(Q'_z\right)^2}$$

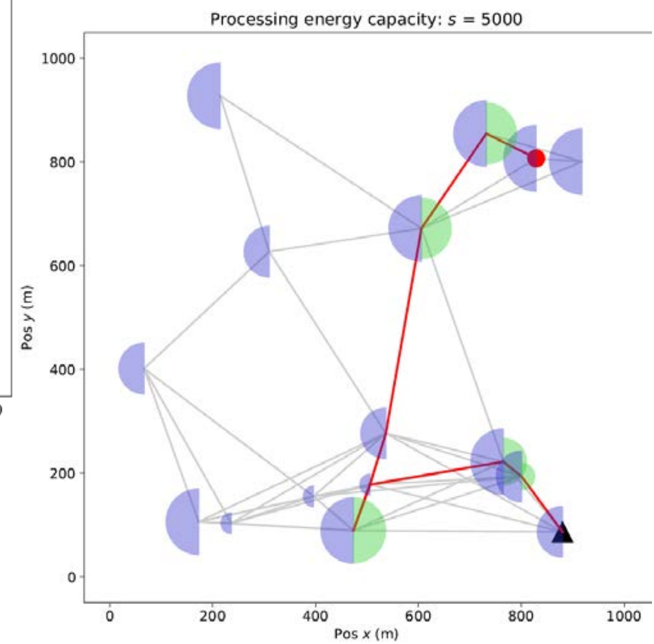
U.S. ARMY RDECOM ARL RESEARCH LABORATORY

DEMONSTRATION: EVOLUTION OF PLACEMENT SOLUTION

Processing energy capacity + utilization





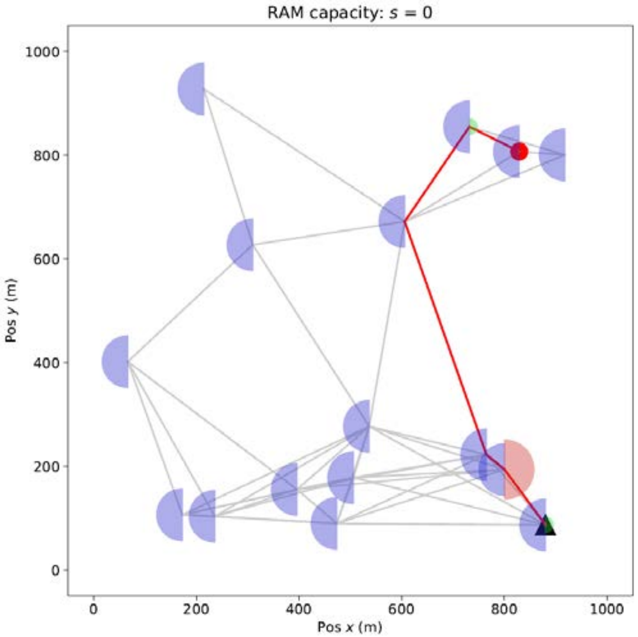


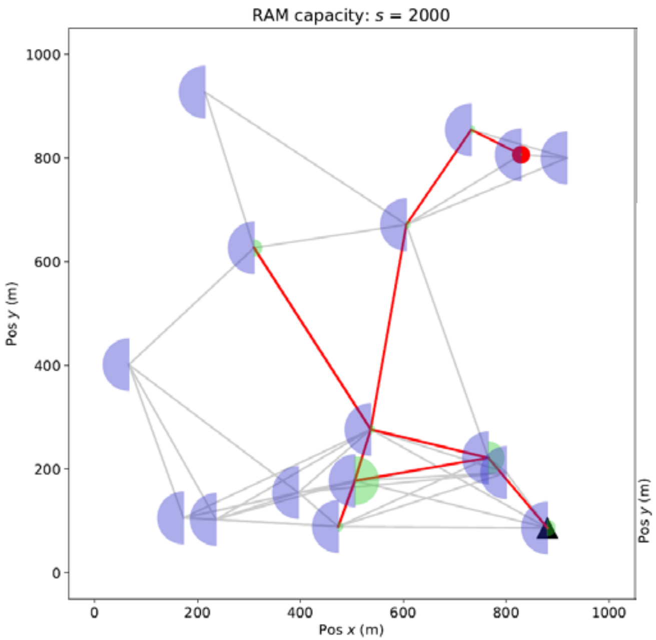
- Only final configuration is important.
- Intermediate configurations show evolution of optimization solution.

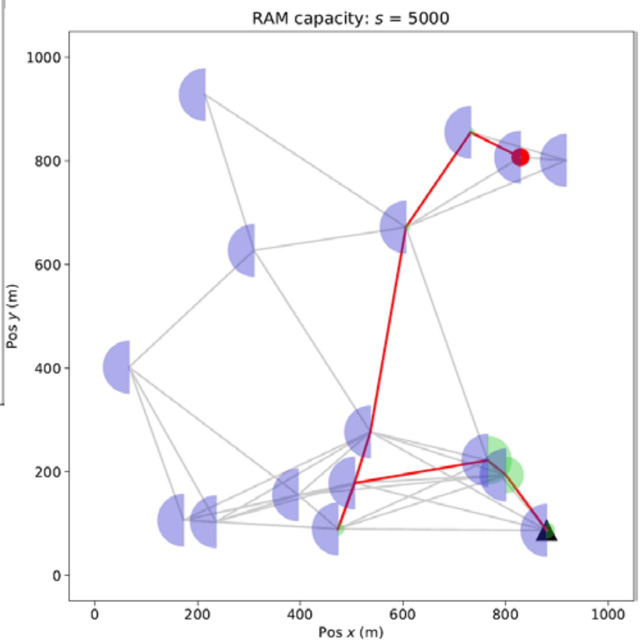
U.S. ARMY RDECOM ARL RESEARCH LABORATORY

DEMONSTRATION: EVOLUTION OF PLACEMENT SOLUTION

RAM capacity + utilization



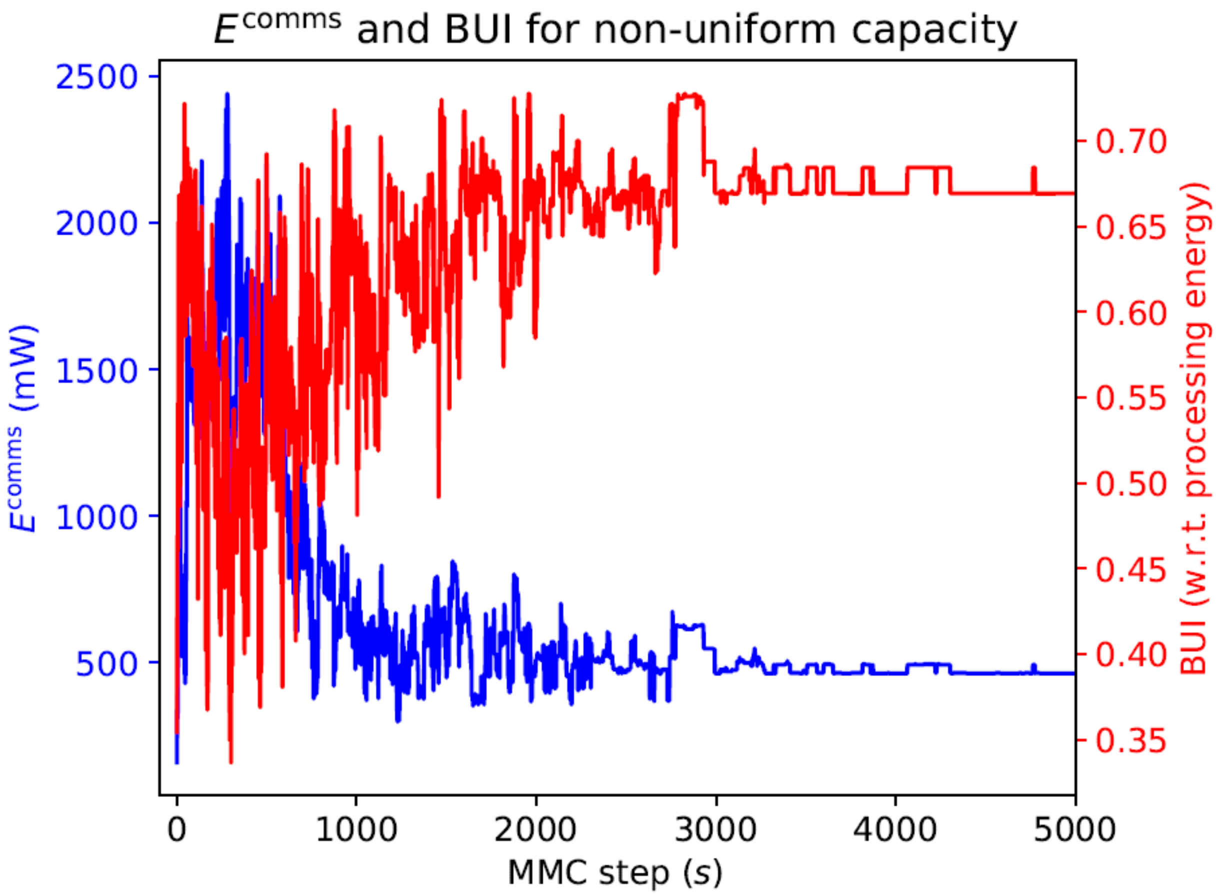




- RAM capacity constraint is robust:
  Satisfied in all calculations in this work

DEMONSTRATION: EVOLUTION OF ANALYTICS PLACEMENT

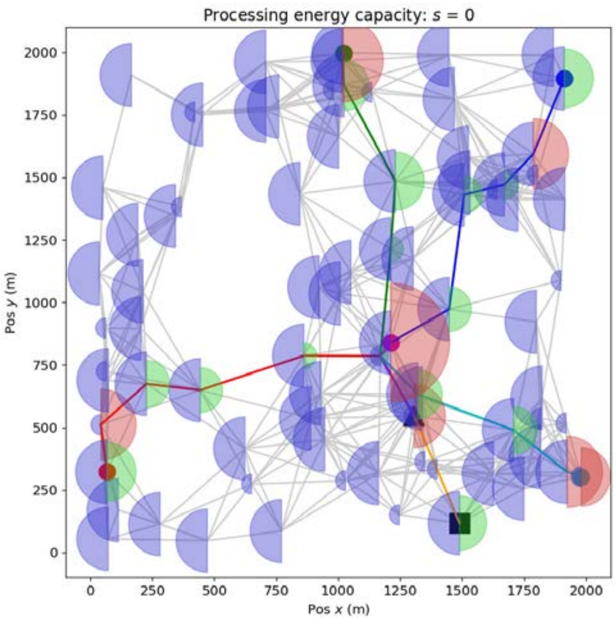
Evolution of E_{comms} and BUI:

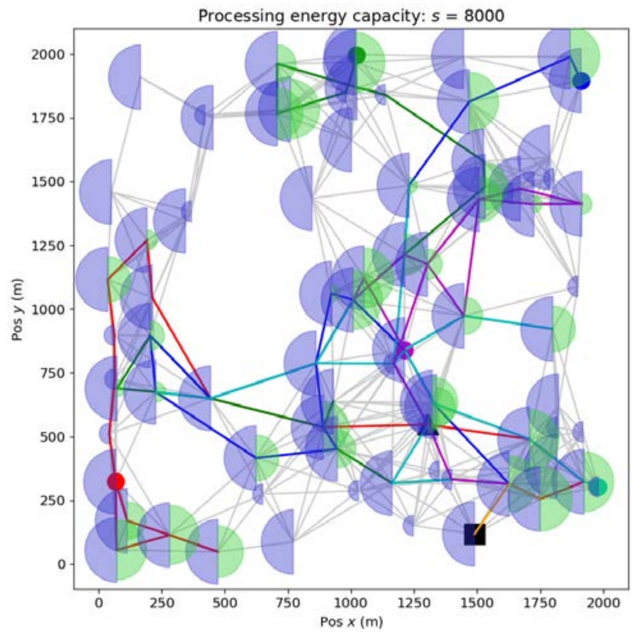


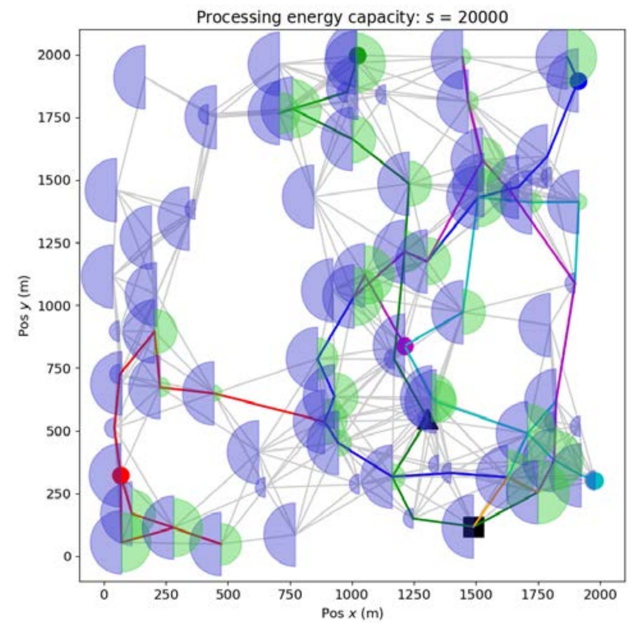
Algorithm is simultaneously optimizing E_{comms} and BUI (minimizing and maximizing, respectively).

PLACEMENTS FOR MULTISENSOR FUSION PROBLEM

Data from 5 sensors, joined at triangle; fusion completed at square:

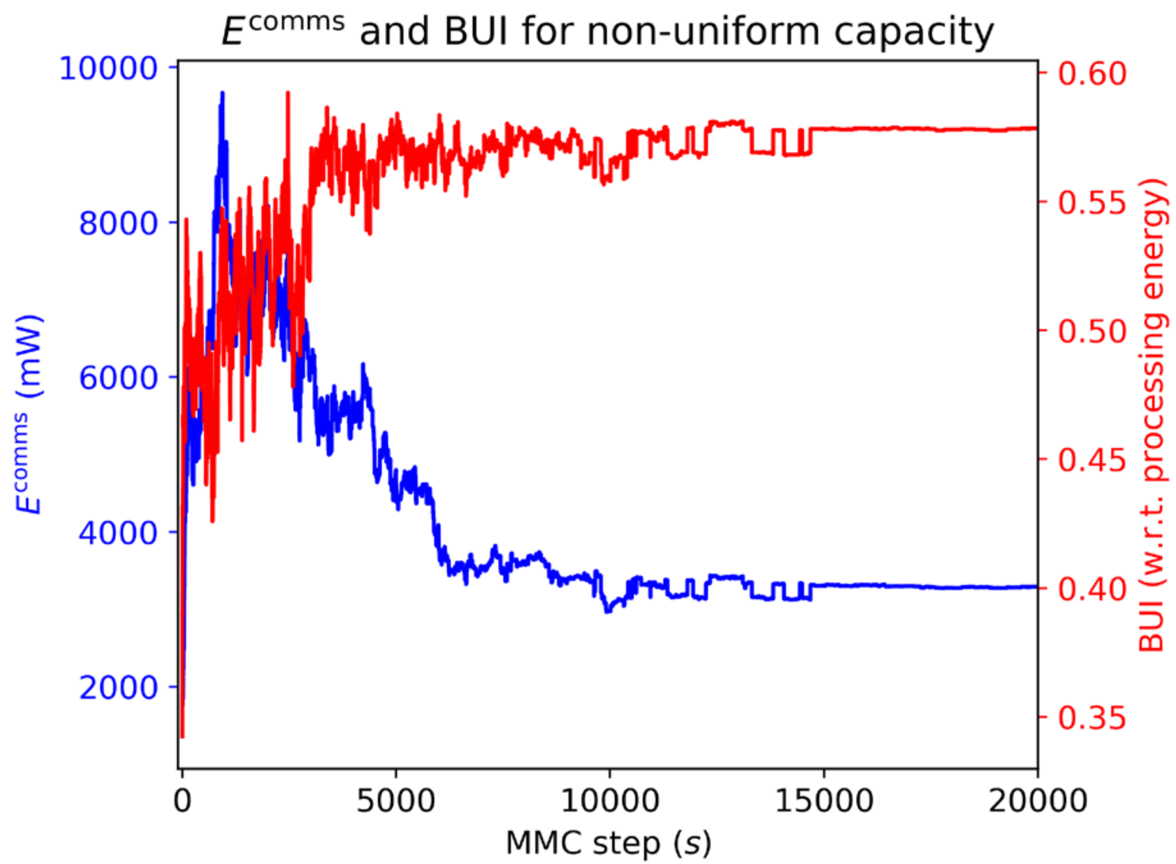






MULTISENSOR FUSION: EVOLUTION OF METRICS VS. MMC STEP

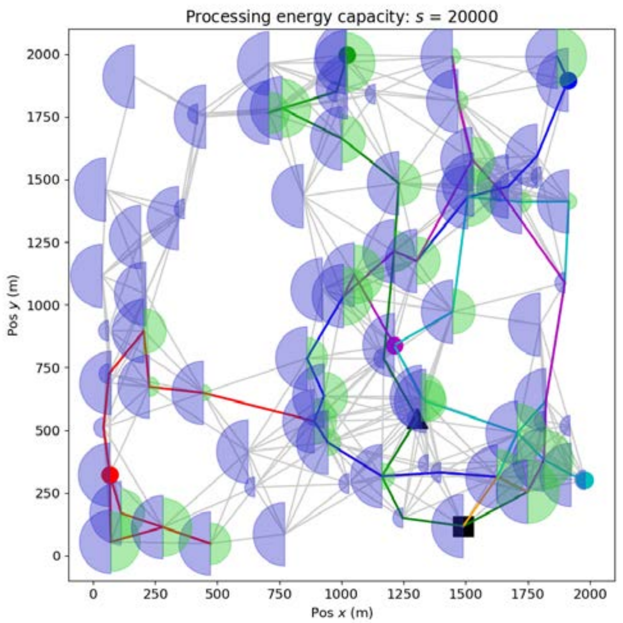
Evolution of E_{comms} and BUI:



Algorithm is simultaneously optimizing E_{comms} and BUI (minimizing and maximizing, respectively).

CONCLUSIONS

- Multisensor fusion will be essential to analyzing incoming flood of data.
- On tactical networks, distributed edge computing has potential to replace distant, cloud-based analysis.
- Placement problem for components of distributed computation can be abstracted as optimal graph embedding problem (OGEP).
- Objective function defined in terms of physics-inspired interactions.
- Metropolis Monte Carlo-Simulated Annealing can find optimal placements that balance utilization while minimizing energy costs.



More details: Kraczek et al., *SPIE Proceedings: Disruptive Technologies in Information Sciences*, **10652** (2018); related work submitted to INFOCOM.