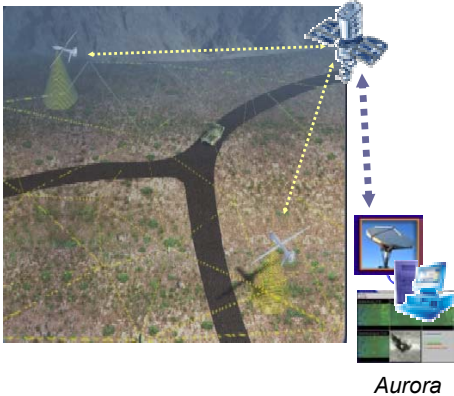


A Situational Awareness Application

Battlefield Monitoring



Aurora

- Military units on a battlefield send geo-position information through GPS
- A video of the battlefield is received through Satellite Images
- Aurora runs monitoring queries (alerts and warnings) defined by officers and displays the results on a visualizer

Monitoring under Resource Constraints

Load Shedding

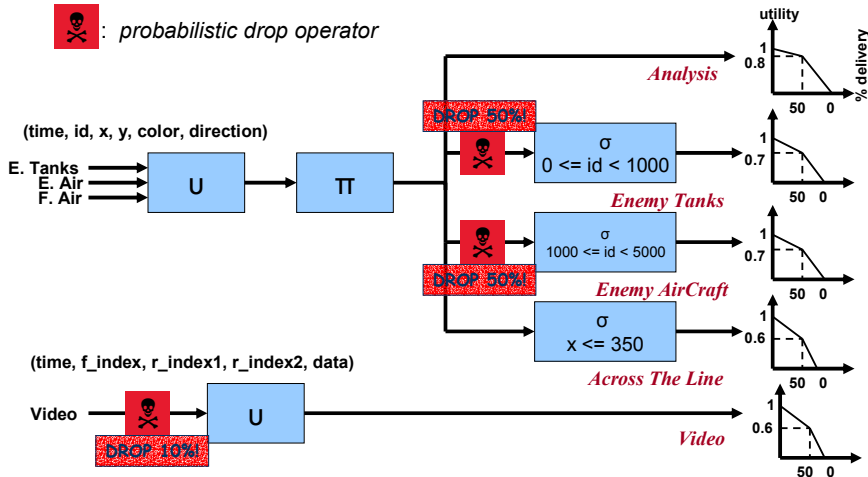
- Resources are commonly constrained
 - e.g., bandwidth between the ground units and the airplane
- Monitoring queries have varying service expectations
 - e.g., “closer” enemy units are more interesting
- Goal: Always deliver the “more” important results under resource constraints
- Approach: *Load shedding*
 - Selectively drop tuples such that
 - excess load is shed
 - loss in the perceived QoS is minimal
 - Accomplished by inserting *drop* operators into the operator network
 - *randomized* drop: probabilistically drop a given percentage of tuples
 - *semantic* drop: filter out tuples based on a drop predicate
 - Questions: Where, when, and how much to drop?

Battlefield Monitoring Network

w/ Randomized Load Shedding

- Goal: drop tuples from the *most loss-tolerant* outputs

: probabilistic drop operator



Battlefield Monitoring Network

w/ Semantic Load Shedding

- Goal: drop the *least important* tuples

: semantic drop operator

