

Scaling Issues

Challenge Problems, Science and Engineering Responses

Robert Neches, P/I

Pedro Szekely, Project Leader

University of Southern California

Information Sciences Institute

Marina del Rey, California

Scaling Issues in the ANTs Program

Requirements

Challenges

■ Three goals

- create real-time resource management systems
- that operate in highly decentralized environments,
- making maximum use of local information, providing solutions that are both good enough, and soon enough.

Tightening time restrictions
Increasing # of resources

Increasingly distributed (single machine, LANs, WANs...)

Larger amounts of information to consider
Interactions between local alternatives

■ Distinguishing characteristic of the ANT approach is the explicit time-bounds on calculation of actions.

Size and complexity of set of action calculations

Essential Elements Provided by DARPA Negotiation Technology

Issues unique to ANT



Challenges

■ Handling incomplete knowledge, context-sensitive decisions

- Preferences unknown until possibilities are known
- Desired tradeoffs change with situation
- Which plane to cannibalize depends on missions

of unknowns
of dependencies

■ Exploring tradeoffs between values

- Mathematical impossibility to satisfy a group w/o negotiation
- Systematic alternative generation uncovers issues and refines preferences

of positions to be resolved

■ Assuring user confidence

- Acceptance fostered by understandability
- Minimally disruptive replanning

Size of plan

Where the Challenges Matter Most: ***Convergence / Closure / Stability***

- What ensures that the system will reach agreement at all?
- What ensures that adequate agreements will be achieved within time bounds?
- What ensures that agreements can be adequately repaired within time bounds
 - if the external environment changes?
 - if there's failure / renegeing within the system?

What Constitutes a Good Answer?

(Alternatively: how do we interpret any answer?)

- What performance / quality characteristics are we trying to optimize in a negotiation control regime?
 - Beat the clock:
Fastest time to an acceptable result?
 - Use the clock:
Best result in time bounds?
 - Satisfice:
Probability of best answer in time bound?
 - ...????

Science and Engineering Agenda

- Strong claim: there may be no universally optimal simple negotiation control regime
- Science implication:
 - Need *comparable characterizations* of key concepts
 - Need to understand performance characteristics of alternatives wrt structural characteristics of problems
- Engineering implication:
 - Need handbook of when to use which method
 - Need to explore mechanisms for changing methods
 - Need to explore automatically changing methods

Presentations Following

- Three application examples:
 - CAMERA (ISI)
 - Autonomic Logistics (Vanderbilt)
 - ANT Testbed (Lockheed Martin)
- Our goal in presenting these three:
convey how they stress-test these issues
 - How are these projects addressing them?
 - What can you do to address them?

this foil still in progress -- may not be used -- ignore for now

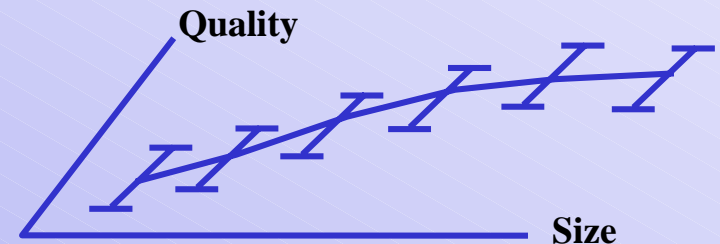
Dimensions

- Increase in number
 - Assumptions
 - Constraints
 - Agents
 - Alternatives
 - Solutions
- Decrease in time
 - Overall time avail.
 - Time per task
 - Between binding constraints and making decision

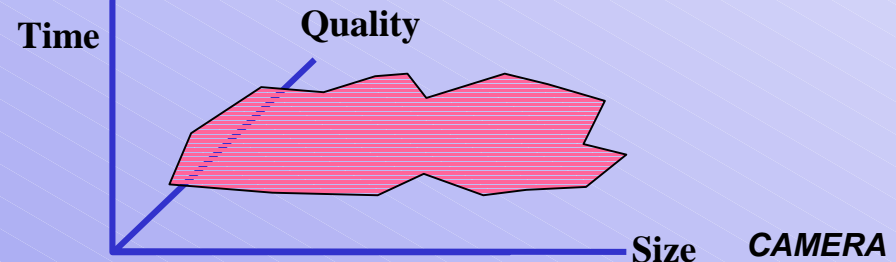
■ Solution Time?



■ Solution Quality?



■ Performance Profile?



Weekly Flight Schedules

Objective

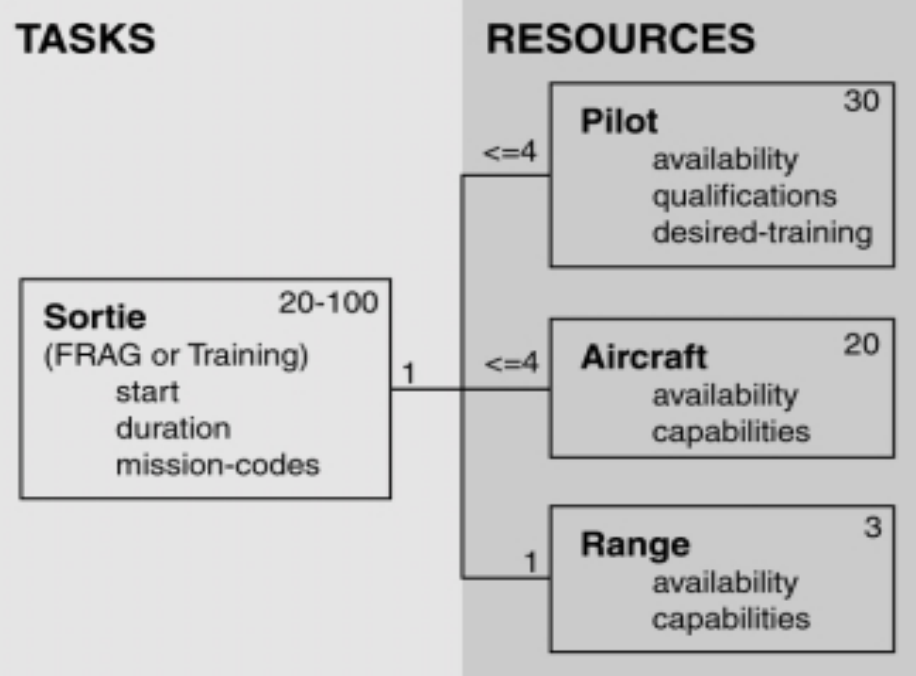
Build weekly flight schedule

■ Input

- FRAG sorties
 - Start, duration & mission codes
- Training goals for the week
 - Mission codes to emphasize

■ To do

- Define training sorties
 - How many, what mission codes
- Assign resources to all sorties
- Schedule training sorties



CONSTRAINTS

Pilot.qualifications *can-perform* **Sortie.mission-codes**
Pilot.availability *contains* **Sortie.start + duration**
Aircraft.capabilities *can-perform* **Sortie.mission-codes**
Aircraft.availability *contains* **Sortie.start + duration**
Range.capabilities *support* **Sortie.mission-codes**
Range.availability *contains* **Sortie.start + duration**
... lunar light, weather constraints

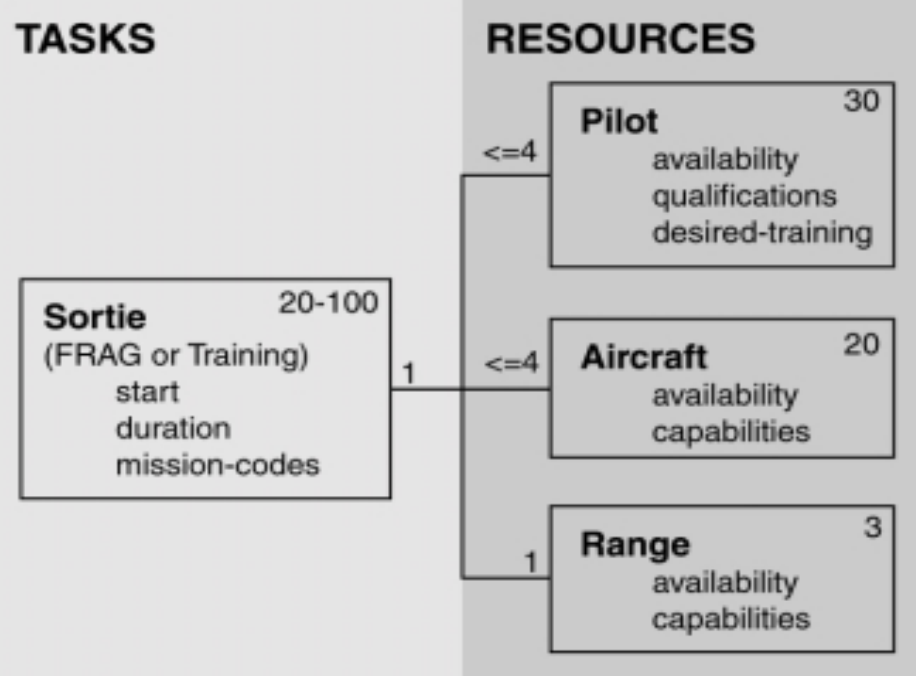
EVALUATION/UTILITIES

Schedule/ Sortie	<ol style="list-style-type: none">1) Minimize risk: depends on pilot qualifications, recent flight history and mission codes2) Perform all FRAG sorties3) Maximize readiness (train pilots)
Pilots	<ol style="list-style-type: none">1) Make fast progress in curriculum (fly desired mission codes)2) Don't let qualifications expire
Aircraft	<ol style="list-style-type: none">1) Maximize usage

Weekly Flight Schedules

Why is it hard

- Incomplete resource information
 - Only rough estimates of number and capabilities of aircraft
- Resources break down
 - Aircraft problems (very common)
 - Pilot snivel
- Option to change the problem rather than solve it
 - Can change resources
 - Convince maintenance to make other aircraft available
 - Can change tasks
 - Simplify training sorties
 - Do other (training) sorties
 - Question: how to identify that it is time to change the problem



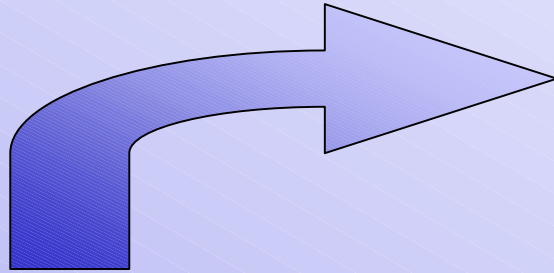
CONSTRAINTS

Pilot.qualifications *can-perform* **Sortie.mission-codes**
Pilot.availability *contains* **Sortie.start + duration**
Aircraft.capabilities *can-perform* **Sortie.mission-codes**
Aircraft.availability *contains* **Sortie.start + duration**
Range.capabilities *support* **Sortie.mission-codes**
Range.availability *contains* **Sortie.start + duration**
... lunar light, weather constraints

EVALUATION/UTILITIES

Schedule/ Sortie	<ol style="list-style-type: none">1) Minimize risk: depends on pilot qualifications, recent flight history and mission codes2) Perform all FRAG sorties3) Maximize readiness (train pilots)
Pilots	<ol style="list-style-type: none">1) Make fast progress in curriculum (fly desired mission codes)2) Don't let qualifications expire
Aircraft	<ol style="list-style-type: none">1) Maximize usage

Approach



CAMERA's Negotiation Model

- Resembles approach humans use to solve this problem
- Focuses on repairing solutions, localizing disruptions
- *Gathers information that is useful later to fix problems*

Camera Approach

- Design architecture for constraint relaxation within time bounds
- Investigate a range of soft negotiation protocols & strategies
- Run simulations to understand suitability of protocols & strategies to problem features
- Build a framework that makes it easy to program agents, protocols & strategies