A Distributed Collaboration Architecture for Global Optimization

Tom Castelli  
+1 (508) 490-2512  
Thomas_J_Castelli@raytheon.com

Josh Lee  
+1 (508) 490-4105  
leejm@raytheon.com

Waseem Naqvi  
+1 (508) 490-3635  
Waseem_Naqvi@raytheon.com

Raytheon Company Network Centric Systems  
1001 Boston Post Road, M/S 3-1-3250  
Marlborough, Massachusetts 01752 USA

11th International Command and Control Research and Technology Symposium  
Cambridge, UK, September 26, 2006
Why Optimization?

• We need to determine a way to control the distributed, large-scale C2 systems of the future.
• These systems will need increased autonomy.
• Each subsystem will have its own goal or objective.
• The overall system will have a greater goal that must be satisfied in order to enable the subsystems to succeed. This goal may or may not conflict with individual subsystem goals.
• For example, we all (presumably) drive to work each morning. We obey traffic lights and follow other rules of the road to orchestrate each individual vehicle’s desire to get to different locations at different times.
Motivation

- Distributed C2 systems deal with complex, large-scale, changing environments. This is different than the C2 systems of the past. Information needs to be readily available for autonomous control.

- Provide greater autonomy to individual subsystems and reduce the bottlenecks of centralized control, while still achieving the global mission.

- We have developed the Global Optimizer And Local Strategizer (GOALS) framework. GOALS is an optimization framework that is easily adaptable to any problem involving a dynamic, distributed environment.
  - Intended for use by future NCS projects.
  - Small, hand-solvable problem in a known domain (air traffic).
Key Issues

- Large-scale, dynamic, distributed agent systems
  - Examples: network packet routing, Air Traffic Management systems, cargo shipment networks, first responders, and more
  - Each agent (participant or actor in the system) has its own local goals, which may or may not conflict with those of other agents. (Examples: minimize fuel usage, minimize travel time, … )
  - Need a means to optimize the global state of the world, as “greedy” agents may “starve” other agents.

- Global optimization in a distributed environment
  - A global optimum is achieved by getting as close as possible to satisfying all local optima.
  - Local agents have little or no knowledge of the world at large.
  - A single, “omnipotent” agent is often impractical.

- How do we achieve global optimality in a distributed, dynamic system?
The GOALS Approach

- Autonomous entities are modeled as software agents
  - Local goals can be modeled as graphs, for example
  - Agents first locally optimize, then negotiate for needed resources
- Negotiation amongst local agents can take many forms
  - Negotiation leads to global optimization
  - Reviewed different negotiation schemes
- Achieving a global optimized state can be tackled in numerous ways
  - For an adaptive system, we chose to implement a game theory/auction theory based approach
- We determined a need for a generic solution
  - C2 and net centric applications present different problems, but with common elements
  - GOALS is a generic framework which can be applied to many problem domains
Resource Markets

- Get as close as possible to local goals.
  - Let agents first be “greedy” and optimize locally.
  - Needed resources for the locally optimal solution are now known.

- Attempt to obtain resources in a market.
  - In order to obtain resources, agents must interact with each other. We model this interaction as a resource market.
  - The market can be auction-based or another type of market; we selected an auction-based market strategy.
  - If resources are not obtainable, the local agent falls back to the next suboptimal solution.
  - Result: A globally optimal solution, which may or may not be locally optimal for all agents.
Resource Allocation/Negotiation Example

• Local Goal: Get from Boston to Chicago.

• Shortest Path: Dashed green arrows.

• Resources won: Solid green arrows.

• Resource lost: Dashed red arrow.

• Now new resources (blue arrows) are needed for locally suboptimal solution fallback.

What does the agent do with the “useless” resource it just won?
Agent Frameworks

- Cybele / Cybele Pro
  - FIPA Compliant
  - Used in prior work for the Airspace Concepts Evaluation System (ACES)
  - We plan to integrate our framework with ACES at a later date
- IBM’s ABLE (Agent Building and Learning Environment)
  - FIPA Compliant
  - Java implementation available (public domain)
  - Allows for machine learning
- JADE (Java Agent DEvelopment Framework)
  - FIPA Compliant
  - Public Domain
- Cougaar
  - Not FIPA Compliant
  - Used by DARPA in previous projects
- We chose Cybele
  - Implementation in progress.
Possible Agent Negotiation and Reasoning Approaches

- Belief, Desire, Intention (BDI)
- Partially Observable Markov Decision Processes (POMDPs)
- Distributed Constraint Optimization (DCO)
- Use of Genetic Algorithms and other Machine Learning Techniques
  - Agents will learn better negotiation techniques.
  - Better performance of agents in the marketplace

**Goal: Combine DCO with learning for better performance.**
Market Mechanism Auctions

- **Auctions**
  - **English**
    - Most familiar to people: highest bid wins
  - **Dutch**
    - \(n\) identical items auctioned simultaneously, highest \(n\) bids win one item
  - **Second Price (Vickrey/Clarke/Groves or VCG Auction)**
    - Highest bid wins, but winner pays the second highest bid
    - Fairness and honesty is ensured – no agent has incentive to bid too high or sell too low
  - **\(n\)th Price**
The GOALS Framework

-Implemented a distributed optimization framework in Java.
  - Collector and Distributor agents for resources.
  - Goals are modeled as graph problems – shortest path for collectors, maximum flow for distributors.
  - Each agent locally optimizes its goals, then bids for needed resources at auction.
  - Implemented an English auction method.
  - If needed resources are not obtained, collector agents fall back to the next locally suboptimal solution and try again.
  - Other local goals and market methods may be implemented in the future.
  - jGraphT library (public domain, graph structures and algorithms).
The GOALS Framework (continued)

- Developed a “prototype” Air Traffic Management scenario to be simulated using the framework as proof of concept.
  - Aircraft (collector) and Center (distributor) agents.
  - Graphs represent paths between cities / NAVAIDs.
  - Graph edge weights represent the “cost” to the aircraft agent of flying that segment (fuel used, distance, time in flight, etc.).
  - Resources represent the right to fly an airway segment starting at a given time.
  - Center agents might want to maximize the flows through their sector (due to fees charged for overflight, for example), but are subject to constraints on the amount of total traffic.
GOALS Framework Diagram

- Agents
- Markets
- LocalGoals
- DomainModels
- AgentFrameworkServices
- Resources
- JGraphT
Mission Assurance

- Our framework ensures that agents come as close as possible to their locally optimal solutions. This in turn will yield a globally optimal solution.
- Many agents working together – if one fails, several others are working on the problem as well.
- Applications in the C2 domain
  - ATM
  - Network routing
  - First responder communications
  - Autonomous vehicle command (e.g. in hostile environments)
  - Other C2 applications
Challenges We Faced

- Validity period of resources was an issue
  - Simultaneous auctions allow a collector agent to obtain resources which may be dependent on others that are not obtained.
  - Complexity of resources – auctioning off not only paths but the right to use them at specific times.

- Winning of unnecessary resources
  - If auctions are simultaneously entered, it is possible to obtain a resource which is dependent on one which is not obtained (due to parallel auctions).
  - Collector then needs to become a distributor, or return the resource to the marketplace.
Benefits

- Our work has demonstrated that by automating the collaboration of multiple entities with various (and conflicting) goals we have been able to reach a globally optimized state – A Win-Win for all entities

- The GOALS Framework can be applied to many problem domains
Conclusions

- We implemented a general optimization framework for distributed, dynamic systems, providing a modeling and simulation framework to understand distributed flows.
- We have created a small prototype application which uses and tests this framework.
- We envision our framework being used for future C2 systems.
- We envision our framework being used in other domains (network traffic routing, intelligent transportation, first responder situations).
Future Work

- Implement other kinds of auctions (Dutch, VCG, e.g.) and alternative negotiation methods
- Implement non-graph goals
- Integrate with other agent frameworks
- Create a complex, “real world” domain model (such as integration with ACES)
- Implement machine learning in agents
  - Genetic Algorithms
  - Bayesian learning
Biographies

Tom Castelli has been a software engineer with Raytheon Network Centric Systems (NCS) for five years, working primarily on air traffic control programs with some experience in satellite communications. He has most recently been working in research and development under Dr. Naqvi for NCS's Airspace Management and Homeland Security (AMHS) business at the Marlborough, MA facility. Tom received his MS in Electrical and Computer Engineering from Boston University in 2003, earned on the Raytheon Advanced Study Program, with project work on facial feature recognition in sign language videos. He also received his BS in Computer Science from Cornell University in 2000.
Biographies

- Joshua M Lee is a Sr. Software Engineer I in the Software Engineering Center at Raytheon's Network Centric Systems facility in Marlborough, Massachusetts. He has 5 years experience in software development on various programs. Josh is currently working on the Network Enabled Operations Joint Airspace Security (NEO) Program. Josh earned a B.S. degree in Computer Engineering from Tufts University and is currently pursuing an M.S. in Computer Science at Worcester Polytechnic Institute.
Biographies

Dr. Waseem Naqvi, Technology Director, is responsible for Contract Research and Development (CRAD) Capture for the Air Space Management and Homeland Security business within Raytheon Company’s Network Centric Systems (NCS) division. Previous to this role, Waseem was the NCS Technology Director for the Northeast Site.

Waseem is an active member of many industry standards groups including: the Object Management Group’s (OMG) Transportation, C4I, and Agent Working Groups; the Air Force Airborne Moving Target Indicators (AMTI)/Ground Moving Target Indicators (GMTI) Working Group; the FAA/Eurocontrol Flight Object Working Group, and the FAA’s System Wide Information Management (SWIM) Policy Working Group.

Waseem has been at Raytheon for over 8 years, starting his career as a software engineer and advancing through project management, technical line management, and strategic direction levels to his current position.