W911NF-08-1-0041 Project - Mid-term Report

Agent Technology Center, Czech Technical University in Prague

Michal Pechoucek

1 Accomplishments for the First 6 Months

1.1 Scenario and Demos

During the first six months of the course of the project we have created 3 major demonstration scenarios and a number of minor scenarios for testing purposes. All of the scenarios utilize the same environment, a hypothetical island with towns, harbors, airports, resource storages, roads etc. This virtual world is powered by the ACROSS2 simulation platform. Various parameters of all demos can be adjusted via XML configuration files.

The three major demo scenarios are:

- **Basic builders demo** – a group of builder agents/units all have the same task to build houses in a given town. Since builders cannot move by themselves, they have to find (hire) transporter agents/units to transport them to the town in order to build houses there. Virtual synchro-points are inserted into all units' plans transparently in order to ensure proper spatio-temporal synchronization of collaborating units.
- **Commander demo** – while in the previous demo all tasks were pre-assigned staticly, this demo introduces a commander agent/unit responsible for assigning tasks to his subordinate builder units dynamically at runtime. Upon receiving a new task, the builders have to react appropriately and update their plans. Moreover, if sand (building resource) is not available at the building site, the corresponding builder has to hire yet another transporter to deliver sand there from a sand repository before the builder can start working.
- **Air-transport (coordination) demo** – this demo introduces transport airplanes to illustrate multi-level coordination. This time, builders hire airplanes (as they are faster than trucks) to handle the transportation but since airplanes can only fly between airports, the airplanes in turn have to hire trucks to take care of the transportation of units or resources between airports and towns if necessary.

1.2 Integration and Software Development

The following sections summarize the key achievements of software integration and development during the first 6 months of the I-GLOBE project.

1.2.1 ACROSS2 Integration

To enable fast development, I-GLOBE is being built on top of existing ACROSS2 simulation platform. The platform had to be adjusted and extended in many ways to fit the needs of the I-GLOBE project. One of the most significant extensions is the new support of **turn-based simulation** allowing the entire (otherwise real-time) simulation to automatically pause itself and wait until all activities scheduled to be executed in a particular simulation clock tick are completed. This feature turned out to be crucial since the planning processes generally require significantly more time than other actions taking place in the virtual world, which is the exact opposite to the real-world situation we are trying to simulate.

1.2.2 I-Plan Integration

For the purposes of the demonstrator I-X was upgraded to I-X 4.5 which is the open source version of I-X that was released in March 2008.

All units/agents in the simulation use I-X planner I-Plan to do their strategic planning. Each unit/agent has its own instance of I-Plan together with a corresponding I-Plan domain description that was designed for each type of unit present in the scenario: a builder, truck, airplane and commander. It is important to point out that all existing (and even future) demos utilize *the same* set of domains for two main reasons: (i) to keep the domain maintenance as simple as possible and (ii) to demonstrate the versatility of the domain definitions. I-Plan problem definitions for each planning session are automatically derived from the current state of the world as percepted by a particular unit/agent.

1.2.3 Agents' Control Modules

In order to maximize the modularity of the system, agents' decision-making algorithms are implemented in the form of pluggable control modules. These modules can be groupped into hierarchic trees through which time-update notifications, sensory data and action feedbacks flow transparently, depending on the particular control module type. This way the behavior of each unit/agent can be defined by a combination of different control modules. For instance, one control module can be responsible for path planning, other for strategic planning, yet another can represent the agent's tactical layer etc.

1.2.4 Multi-layer Architecture

One of the main accomplishments has been the design and implementation of the multi-layer planning architecture. The architecture is based on a hierarchy of planning layers using the extended form of social commitments as the integration part. Because of the modularity of the layers, the system is ready for implementation of various planning methods and can be easily used for their comparison. The current implementation consists of three layers: strategic, tactical and individual.

- **Strategic layer:** The layer provides an overall strategic plan for middle- and long-term time horizon using I-Plan for high-level planning and distributed allocation based on multi CNP protocol and social commitments.
- **Tactical layer:** On this layer, the strategic plan is optimized using a commitment condensation algorithm. The algorithm searches for blank time slots in the plan and replaces them with later commitments (taking in account the time constraints of the commitments).

- **Individual layer:** On this layer, the units perform reactive behavior and path-finding based on the tactical plan (i.e. an ordered set of commitments).

Each agent uses all layers or their subset (e.g. the commander unit does not use the individual layer).

As described before, in I-GLOBE there are several independent entities acting in the scenarios. The coordination and cooperation of such actors refers to a complex distributed planning and allocation problems. Although the planning is provided by a set of I-X planners, the distributed resource allocation has to be solved with respect to the individual actors' constraints and abilities.

We have designed and implemented mechanisms for distributed allocation based on recursively refined hierarchical plans based on allocation. This approach enables efficient coordination of the distributed autonomous units. The basic idea is to create a raw plan to fulfill a task assigned to the agent. This plan contains activities that can be performed by the agent and some activities out of the scope of the agent's capabilities. So, the agent has to find other agents to assign such activities to them. The allocation is achieved by the means of negotiation between agents in the contract-net-protocol manner. During the negotiation all the responding agents refine requested activities (i.e. create a more detailed plan) and potentially generate new requests in the lower level of the plan hierarchy. This process is recursively repeated until all the atomic activities are assigned to the particular actors.

For capturing the frame of negotiation we have introduced the commitment representation as a context of agents' agreements. The distributed allocation (and also planning) is then represented by the negotiation over commitments. The agent is able to commit itself to perform requested tasks within the scope of the agreed commitment. The plan to fulfill the task is prepared by the agent locally. For the parts of the plan that the agent is not able to perform by itself, the contract-net-protocol is initialized to find another agent that commits to it. This approach transforms the distributed allocation problem to negotiation over commitments. The whole distributed plan can be then viewed as a commitment graph.

1.2.5 AgentFly Integration

The AgentFly system was integrated as one of the possible individual layers in the planning architecture. The AgentFly system is a simulation air-traffic-control system developed by ATG, focusing on a multi-agent control of unmanned aerial vehicles and adopting the free flight concept.

In order to seamlessly integrate the two existing systems an AgentFly wrapper was implemented. On one hand, the wrapper fulfills the needs of the multi-layer planning architecture and on the other it acts as a control module of the AgentFly pilot agent. Using this wrapper, the I-GLOBE system can transparently control the missions of the simulated UAVs and does not need to implement the plane's behavior and simulation on its own. The planner wrapper is based on an agent-to-agent protocol and currently it can handle the initial definition of the mission and feedback sent from the AgentFly system to I-GLOBE once the mission point is reached.

1.3 Research

We have analyzed the state of the art in the field of distributed planning and social commitments.

We have extended the well-known formal model of social commitment by prof. Wooldridge and proposed its recursive form, usable for nesting of commitments and a more expressive formal model of the commitment convention. This model allows us to describe the decommitment rules, which describe the mentioned commitment modifications within the commitment and thus within the plan. The modifications are:

- **Restriction** – the commitments can be negotiated with flexible or loose constraints. The flexibility decreases with time and when the commitment is about to be executed it has to be fully fixed.
- **Relaxation** – when the committed constraints cannot be kept, the constraints should be renegotiated to preserve the commitment.
- **Delegation** – in case of a failed relaxation, the commitment may be delegated to another actor.
- **Decommitment** – when all the previous possibilities fail and the commitment cannot be executed with the agreed constraints it can be dropped. In this case, the reallocation and potentially replanning takes place.

All the above modifications of commitments can also affect the violation links and thus the whole commitment graph can be affected. Our goal is to minimize the impact of commitments modification and minimize the changes in overall commitment graph.

Using the extended form of the social commitment we proposed a graph notation of the commitments. The mutual bindings and commitments form a commitment graph. The commitment graph describes the same properties of the mutual decommitting as the logical notation. The graph notation can be used to describe the process of successive solving of the exceptional states. The process is based on traversing through the commitment graph. The traversing starts with the first violated commitment. One of the decommitment rules is then triggered, according to the violation type. The traversing process propagates the decommitment requests through the commitment graph.

2 Planned Action in the Next 6 Months

2.1 Scenario and Demos

The first of the planned scenarios is the UAV Surveillance Scenario. The scenario will be based on the draft from Mark Adkins. The scenario will contain UAVs with various abilities, including taking of snapshots, image strips, tracking of movable objects and large-area surveillance. Initial missions will be recieved by the appropriate UAVs from the aerial commander unit. The UAVs will be selected using multi-agent distributed resource allocation methods. The mission will describe which action should be performed at what time and location. Later during runtime, other UAVs will enter the scenario. These ones will be able to adopt surveillance tasks from other entities. The delegation process will be initiated by the UAVs currently carrying out the mission, and will use the delegation decommitment rule of the mission commitments. The delegation process will use the multi-agent negotiation methods to preserve the common social welfare. The process of tracking will invoke relaxation of the commitments. The plan will be shifted into the future until the delegation of the tracking takes place, or tracking is withdrawn by the commander.

The second of the planned scenarios will show re-planning abilities of the planning architecture. Several non-deterministic effects will be tested including disabling of the units, exploration of the unknown area and changing the topology of the island's traffic infrastructure and the key locations (resource repositories, airfields, cities, etc.).

Another task is to extend scenarios to include UAVs and temporal planning at the strategic layer, i.e. to add time constraint manager to I-X.

The last planned scenario will integrate all the introduced aspects of the I-GLOBE project, showing them in a single complex domain – Disaster Relief Scenario. The scenario will start with several tasks assigned to medical units. These units will have to deal with the injured in various designated areas. The units will negotiate over the transportation which must be covered by the UAV surveillance. Later, during the execution of the scenario, other tasks will occur (dealing with the injured, surveillance, transportation, building medical tents, etc.) and the units will have to re-plan seamlessly to continue solving the recent tasks and the new tasks simultaneously.

2.2 Integration and Software Development

For the second 6 months period we plan to implement the following features:

Monitoring of commitments execution – the local intra-agent monitoring of the commitments execution and overall inter-agent commitments execution monitoring will be designed and implemented. The Commitment Visualizator will be improved to enable commitments execution visualization.

Feedback propagation (positive and negative) – when a commitment execution is finished (successfully or by failure), the feedback has to be propagated back. The mechanisms based on synchro-points and violation links will be developed.

Commitments modifications – in the case of a non-deterministic scenario the world is changing during commitment execution in a non-predictable way. Some events can lead to the need of commitment modification. We will focus on the following cases: *Restriction*,*Relaxation*, *Delegation*, *Decommitment*. All the above modifications of commitments can also affect the violation links and thus the entire commitment graph can be affected. Our goal is to minimize the impact of commitment modifications and to minimize the changes in the commitment graph in general.

I-P2 user interface– extending scenarios such that commander agent has I-P2 user interface to allow for mixed initiative planning

State subscription – develop and implement interfaces and protocols to allow state monitoring for the commander through a selective subscription mechanism

The AgentFly wrapper will be extended with the re-planning ability. I-GLOBE will be able to change a part of the currently processed AgentFly plan. This process must take into account the time delays between the two planning systems and must be able to handle the re-planning requests from both sides.

It is expected that more complex future scenarios may require further extensions and improvements to the existing ACROSS 2 platform, I-GLOBE multi-layer planning architecture and flexible commitments representation.

2.3 Research

We plan to focuse on following research targets:

Multi-attribute negotiation about commitments – We will study different methods of negotiation over the commitments. The negotiation will be based on multi-attribute commitment descriptions.

Reasoning about the decomitment rules – We will analyze how agents perform reasoning about decommitment rules under various conditions, world states and predictions of future world state evolution.

Resource exploitation – Resource exploitation impact will be compared for different scenario settings with various sets of decommitment rules.

Volatility minimization – Volatility minimization efficiency validation will be performed by comparing measurement results of the system with and without decommitment rules applied.

Custom tailored reactive plan – The research topic is based on the observation that the rich action description (i.e. decommitment rules in the commitment) forms a reactive plan, tailored according to the predicted future of the evolving world state.

Distributed planning – Using the I-Space tool in I-X to model other agents.

Planning to a time horizon – Develop an agent's plan only up to a fixed time horizon. This should maximize UAV utilization given the uncertainty in the environment. This leads to a nice linking of research agendas as this technique may require decommitting as the time horizon lengthens.

3 Technical Issues Encountered

We realized the importance of planning with quantitative time for fully automated planning (e.g. within UAVs) but also in mixed-initiative planning (e.g. with the commander).

We had to extend the multi-agent simulation system A-Globe Simulation with the time scalability of the simulation: now the simulation can be arbitrary set to work in any value of the interval: turn-based simulation – real-time simulation.

ACROSS 2 had to be extended with new action processing, handling real duration of the actions and ability to stop currently processed action any time. The linkage system (linkage of the entities, e.g. if the truck transports the builder) had to be rewrited, to be able to work in the turn-based simulation.

The integration of the AgentFly system required complete new ad-hoc protocol designed to fulfill the needs of both project.