Naval Research Laboratory
Washington, DC 20375-5320

Final Report: Combining robotic behaviors for safe, intelligent execution

Laura M. Hiatt
Email: laura.hiatt@nrl.navy.mil

Navy Center for Applied Research in Artificial Intelligence (NCARAI)
Information Technology Division

Keywords: Autonomous systems, cognitive architectures, robot planning, robot execution, motion planning, temporal goal networks

September 8, 2021

This **NCARAI Technical Note** is the final project report for the NRL Base Program project "Combining robotic behaviors for safe, intelligent execution."

Introduction and Objective

The goal of this project was to develop a framework in which different cognitive skills and behaviors can be combined to produce intelligent and safe robotic behavior. The DoD Autonomy COI has recently identified a problem in autonomy and artificial intelligence research; namely, that the majority of the AI behaviors being produced work largely independently, and are unable to be combined with other behaviors or skills without significant research and development efforts. In this project, we took steps towards addressing this problem by developing a framework in which the many successful cognitive skills and behaviors we have developed over the years can be safely combined to produce intelligent robotic behavior.

To illustrate, consider a robot whose job it is to patrol around a secure facility with a few simple tasks: ensuring that everyone it sees is authorized to be there, and scanning the building to ensure that laboratory and office doors are always closed and secure when no one is inside. Now, suppose that the director of the facility comes up to the robot and, walking alongside it, asks it to report how its day is going. What should the robot do? The robot has access to relevant behaviors and knowledge (it knows how to patrol, it knows how to go down a hallway with a human, and its individual behaviors know what they have done that day), but it has not been explicitly engineered to do these things together.

Fully handling this situation requires the robot to move beyond the paradigm of executing isolated, separate behaviors and combine its component behaviors both in terms of task execution and in terms of knowledge. Its reasoning about what to do, for example, requires the robot to consider the relative utilities of either talking to the director or pursuing its patrol deadline. The highest utility course of action, for example, is to pursue both potential goals simultaneously talking with the director as it continues down the hallway to patrol; but this interleaving of the involved behaviors raises potential safety concerns that would need to be taken into account when developing its plan of action (such as being sure to not run into the human while turning to look into a doorway). Its report to the director of its day requires the robot to have aggregated knowledge from the behaviors it has executed that day into a comprehensive knowledge base it can draw upon to give an intelligent, meaningful report. As we move towards the goal of competent, tactical robots working in the field, these issues will be even more important to solve in order to ensure that the robots can safely and intelligently assist the Warfighter.

In this work, our objective was to develop a framework that will take steps towards combining individual behaviors and skills by researching these two important questions: (1) how do we select what behaviors and skills to execute and interleave at any given time, considering both utility and safety? and (2) how can knowledge gained during the execution of unrelated skills and behaviors be meaningfully combined to support intelligent behavior?

Technical Approach

In this work, there were two ways in which behaviors needed to be combined: their physical execution, and their internal knowledge. We next discuss each of these contributions in more detail.

Combining Behaviors' Physical Execution. The first part of the project involved how to best determine which skills and behaviors can physically be executed and interleaved at any given time. Literature in motion planning typically treats robots as unary resource, precluding interleaved or multitasked execution. Therefore, the driving force behind this work was to enable independent and asynchronous motion planning for different kinematic chains (such as two arms) of a robot.

One of the key challenges for this asynchronous motion planning was accounting for the other moving objects in the environment (such as the other kinematic chain moving asynchronously). In other words, when planning the motion for one arm, the plan needed to ensure that it won't collide with the other arm that is moving independently. Classical motion planning approaches such as Probabilistic Road Maps (PRMs) assume a static environment at plan time, which precludes asynchronous motion. Approaches such as dynamic Rapidly- exploring Random Tree (RRT) have the ability to account for other dynamic objects in the environment, but at the cost of the occasional failure (such as if an arm can't get around a dynamic object in its way).

Our approach had the benefit, however, of knowing, at plan time, the dynamics of the other kinematic chains / arms currently moving in the environment. Here, then, we took advantage of this to expand PRMs to not only plan in 3D space, but also to plan with time as a dimension. When checking for a valid path, instead of checking for static objects at any given point in space, it checks for static objects at any given point and *at a specific time*. This allows the motion planner to plan around other joints of the robots that are moving independently to generate a correct, asynchronous motion.

Combining Unrelated Behaviors for Overall Execution. The next part of the project involved how to best plan for which behaviors should be physically executed and interleaved at any given time. Answering this general question is the driving force behind research in goal reasoning, which develops plans for autonomous agents by reasoning about the utility of sets of possible goals and ways to execute them. Goal reasoning, at a high level, has a 4-step goal lifecycle: (1) select goals; (2) select a set of behaviors that will accomplish those goals; (3) schedule the selected behaviors; (4) execute, and address any inconsistencies that arise during execution by returning to steps 1-3, as appropriate, to repair the plan.

A key challenge for considering goal reasoning here, however, was supporting interleaved behaviors. This requires reasoning, as part of the planning loop, about the constraints that the interleaved behaviors place on each other, both temporal (e.g., ensuring that the robot can meet its deadline even if it adds a behavior to its schedule) and resource (e.g., ensuring that an interleaved behavior doesn't violate the safety of another).

To accomplish this, goal reasoning needed to be extended to support *temporal goal reasoning*, where behaviors can place temporal constraints upon one another. This allowed the goal management of the system to more tightly couple knowledge about the constraints of the platform to produce more flexible and online planning. We accomplished this by extending canonical goal networks to *temporal goal networks* (TGNs), where time can also be represented.

Importantly, however, goal reasoning does not know about the geometry of the world; in other words, it does not know which motions can physically be combined and which cannot. To address this, we first developed plans that assumed maximal multi-tasking is possible. We generated these plans by taking plans output by temporal goal reasoning, and converting them into flexible partial-order plans that allow for overlap between different tasks in the plan. Then, we developed an executive loop that, at any given time during execution, tried to generate motion plans for any tasks that were "available" for execution using the products from part 1, above. If such an asynchronous motion plan is found, it sent that task off to be executed.

We showed that combining these approaches will result in more flexible and efficient execution of tasks on humanoid robots. We demonstrated this on several manipulation tasks using the bimanual Hubo robots in simulation.

Technical Progress

FY18:
Robot progress:

- We have motion planning and collision avoidance working for the Hubo robot in simulation.
- We have inverse kinematics working for the Hubo robot.

Software/planning/reasoning progress:

- ActorSim was in the process of being extended to support constraint networks and temporal planning.
- We were in the process of investigating how to represent and reason about initial abstractions of motion constraints using the Cozmo robot and path planning (a subspace of motion planning).
- We developed an approach to shared control that combined human input and robot task knowledge while a person is teleoperating a robot.

FY19:
Robot progress:

- We designed, and were working implement, different basic behaviors for the Hubo robots such as pointing and reaching for an object.
- We have an early version of navigation code for the Hubo.
- We expanded our approach to motion planning to include constraint-based motion planning.
- We have an implemented and published approach for constraint- and optimization-based bimanual teleoperation for the Hubo. The evaluation showed that this approach makes human teleoperation much more effective than other approaches. We drew upon the knowledge garnered in this work as we continued the project.

Software/planning/reasoning progress:

- We extended our existing planner to be able to reason about delivering objects to others in a buildilng; we were also working on finalize the goal structure for this scenario.
- We extended ActorSim to include simple goal orderings, and we were working on including in more sophisticated temporal reasoning. These steps were necessary to reason about combining and overlapping tasks, since time is a critical aspect of that type of reasoning.
- We have an initial implementation of a clustering heuristic for determine which motions can be combined while path planning (a subspace of motion planning).

FY20:
Robot progress:

- We spent a substantial amount of time developing a new motion planner that can move two arms asynchronously while also safely checking for collisions. Motion planners for humanoid robots assume that all movements are synchronous: that is, that movements occur in a known temporal relationship to one another and so can be planned out in advance. In our project, however, our goal of concurrency and interleaving required a motion planner that can safely move a humanoid's arms and body asynchronously, where movements may be planned or and/or started at different times.

- Given COVID, we also invested in improving our simulation capabilities for the robot, and are working on ramping up full simulation capabilities in Gazebo, a physics-based 3D robot simulation environment.

Software/planning/reasoning progress:

- We integrated our goal reasoning system with a translator that can convert the output of an off-the-shelf planner into a flexible partial-order plan that we used during execution to maximize multi-tasking.
- We developed a draft implementation of temporal goal networks and were integrating it with the flexible partial-order plan described above.
- We also explored heuristics for inserting new items into constrained plans such as the ones we reason about here. Our work here included investigating how humans do such reasoning, and then implementing the heuristics computationally to evaluate their performance. Our work showed that using human-based heuristics to insert items into a temporal plan leads to faster performance while still resulting in the optimal solution.
- We developed and implemented an execution lifecycle as part of our TGN development. Our Execution Lifecycle synthesized state-of-the-art execution tracking from the planning-and-execution literature while blending it with concepts from ActorSim's existing Goal Lifecycle. Execution nodes transition through several states -- Inactive, Waiting, Executing, Completing, IterationEnded, and Completed -- based on conditions that are established as part of the goal in the TGN (e.g., a deadline) *or* conditions from the execution platform (e.g., a safety constraint). These execution nodes served as a key communication point between the task planning and physical platform concerns, which our approach required.

FY21:

Robot progress:

- We developed an approach, based on the canonical motion planning approach Probabilistic Road Maps (PRMs), that incorporates time as an added dimension to the motion planning problem. In other words, instead of checking for collisions during motion planning in only three dimensions, we also checked for collisions in a certain place *at a certain time*. This allowed us to plan around other objects (such as another arm) that we knew was moving around in the environment. This was a huge success for our project and is a significant contribution to the research community.
- We completed incorporating Gazebo (a 3D robotics simulator) into our workflow to allow for higher fidelity testing of our work in simulation.

Software/planning/reasoning progress:

- As the final contribution to this project, we wrote an executive for the robot that ties together planning via temporal goal networks with the new, improved motion planner. The executive leveraged both the flexibility from our partial-order plan representation as well as the motion planner's support for asynchronous motion to maximize multi-tasking during execution. At any given time, it checked the partial-order plan to see which tasks were "eligible" for execution. Then, in an asynchronous fashion, it attempted to develop a motion plan for them. If one was found for a task, then that task could be successfully executed and it was sent to the robot to physically achieve. Otherwise, the task was held back for later, such as when other arm's motions allowed for it to be physically executed. We finished this executive and have a full, end-to-end demonstration working in simulation.

Naval / Marine Corps Needs

The DoD Autonomy COI recently identified a problem in autonomy and artificial intelligence research; namely, that the majority of the AI behaviors being produced work largely independently, and are unable to be combined with other behaviors or skills without significant research and development efforts. Our work addresses this problem.

Dual Use

While still research, in the long term our work will improve robotic autonomous systems' abilities to support and extend human partners by allowing them to more effectively and more efficiently operate in the world around them. This is true for applications both within, and outside of, the DoD.

Transition Plan

Many of the products from this project will be used in a follow-on FY22-25 6.2 base funding project, "Learning Contextually-Dependent Robotic Skills."

Publications

Rakita, D., Mutlu, B., Gleicher, M., and Hiatt, L. M. (2018). Shared dynamic curves: A shared-control telemanipulation method for motor task training. In Proceedings of the 2018 ACM/IEEE International Conference on Human-Robot Interaction, pages 23–31.

Rakita, D., Mutlu, B., Gleicher, M., and Hiatt, L. M. (2019). Shared control-based bimanual robot manipulation. Science Robotics, 4(30).

Hiatt, L. M., Lawson, W. E., and Roberts, M. (2018). Learning context through cognitive priming. In Computational context: The value, theory and application of context with AI. CRC Press.

Rabkina, I., Kathnaraju, P., Roberts, M., Wilson, J., Forbus, K., and Hiatt, L. (2020). Recognizing the goals of uninspectable agents. In AAAI Workshops (PAIR).

Rabkina, I., Kathnaraju, P., Roberts, M., Wilson, J., Forbus, K., and Hiatt, L. (2020). Recognizing the goals of uninspectable agents. In Advances in Cognitive Systems.

Rosenthal, S. and Hiatt, L. M. (2020). Human-centered decision support for agenda scheduling. In International Conference on Autonomous Agents and Multi-Agent Systems.

Rabkina, I., Kathnaraju, P., Wilson, J., Roberts, M., and Hiatt, L. M. (2021). Comparing hierarchical goal recognition via HTN, CCG and analogy. In AAAI Workshops (PAIR).

Roberts, M., Hiatt, L. M., Shetty, V., Brumback, B., Enochs, B., and Jampathom, P. (2021). Goal lifecycle networks for robotics. In FLAIRS.