Multi-optimization Criteria-based Robot Behavioral Adaptability and Motion Planning

DOE Project Number: 82794

PI: Dr. François G. Pin, pin@ornl.gov, (865) 574-6130

Research Objectives:

Our overall objective is the development of a generalized methodology and code for the automated generation of the kinematics equations of robots and for the analytical solution of their motion planning equations subject to time-varying constraints, behavioral objectives, and modular configuration.

Robotic tasks are typically defined in Task Space (e.g., the 3-D World), whereas robots are controlled in Joint Space (motors). The transformation from Task Space to Joint Space (through the inverse kinematics equations) must consider the task objectives (e.g., high precision, strength optimization, torque optimization), the task constraints (e.g., obstacles, joint limits, non-holonomic constraints, contact or tool task constraints), and the robot kinematics configuration (e.g., tools, type of joints, mobile platform, manipulator, modular additions, locked joints). Commercially available robots are optimized for a specific set of tasks, objectives and constraints and, therefore, their control codes are extremely specific to a particular set of conditions. Thus, there exist a multiplicity of codes, each handling a particular set of conditions, but none suitable for use on robots with widely varying tasks, objectives, constraints, or environments. On the other hand, most DOE missions and tasks are typically "batches of one". Attempting to use commercial codes for such work requires significant personnel and schedule costs for re-programming or adding code to the robots whenever a change in task objective, robot configuration, number and type of constraint, etc. occurs.

The specific goal of our project is to develop a single "generic code" that can automatically generate the inverse kinematics equations of robots, is useable for all types of robots (mobile robots, manipulators, mobile manipulators, etc.) with no limitation on the number of joints and the number of controlled Task-Space variables, and can solve the motion planning equations in analytical form with real time (at loop rate) changes in number and type of constraints and in task objectives, while also adapting to changes in kinematics configurations (change of module, change of tool, joint failure adaptation, etc.).

Our overall approach is based on a novel method for the resolution of under-specified systems of algebraic equations, involving both a novel formulation of the general solution as:

$$S = \left\{ \Delta\bar{q} \in \Re^{m},\ \Delta\bar{q}(t_1, \ldots, t_{m-n+1}) = \sum_{i=1}^{m-n+1} t_i \bar{g}_i,\ \sum_{k=1}^{m-n+1} t_k = 1 \right\}$$

and a Lagrangian-based constrained optimization

$$L(t_i, \lambda, \lambda_j) = Q(t_i) + \lambda \left[\sum_{i=1}^{m-n+1} t_i - 1 \right] + \sum_{j=1}^{r} \lambda_j C^j(t_i)$$

where

$$Q(\Delta\bar{q}(t_i)),\ i = 1, m-n+1,$$

is the general criterion to be optimized, and

$$C^j(\bar{q}, \Delta\bar{q}(t_i)) = 0,\ j = 1, r;$$

represents a set of r general constraints. The unique aspect of this approach is that it leads to analytical solutions that simultaneously fulfill the optimization criterion and all constraints applicable at the particular time-step (loop rate) [1, 2, 3, 4].

Research Progress and Implications:

As of the second year and a half of this 3-year project, we have completed the methodology development and corresponding code for four major modules of the overall software system. The first is the automatic generation of the forward kinematics equations for robots with arbitrary numbers of actuated joints. From the Denavit-Hartenburh (DH) parameters representing each link-joint module, the transformation matrices representing the modules are automatically generated to produce the global transformation matrix of the complete system, from which the forward kinematics equations are extracted, in symbolic form. From this symbolic representation, automated differentiation is performed to automatically generate, in symbolic form, the Jacobian matrix for the overall system [5]. This approach is valid for any link-joint system and, therefore, it applies to tools

as well as manipulators and mobile manipulators. Because the automated generation can be invoked at any time during operation, immediate adaptation of the kinematics and motion planning equations is now possible when changes of configuration occur (e.g., modular robots, change of tools, etc.) [6]. Because it impacts the entire field of robotics by allowing automated generation of robots equations, this module, which has been named the "JFKengine" (Jacobian and Forward Kinematics engine) has been made available for wide dissemination, with full source code, via a public website [7]. The more-than-two-thousand visits to this site and more-than-five-hundred e-mails received during only the first two months of activity of the site provide an indication of the interest and impact of this code on the robotics community.

The second aspect of the overall methodology that has been completed involves a dynamics simulation system allowing visualization with full 3D geometry. It models Newtonian mechanics with coulomb friction. By exploiting recently developed rigid-body simulation techniques, it is able to simulate complex systems (even with multiple interacting robots) in real-time. This dynamic simulation capability opens the possibility of research using hybrid real and simulated systems – 'hardware-in-the-loop' simulations - and will allow us to explore the issues involved with use of the same motion planning code to drive real robots or their simulation, in particular during changes of configurations, constraints, and task objectives.

The third major activity has focused on enhancing the earlier version of the basic solver code, i.e., the code generating the entire solution space for the under-specified system of equations and performing the constrained optimization. The code was verified for proper resolution in comparisons with the standard Moore-Penrose pseudo-inverse in sample problems involving manipulators and mobile manipulators with various configurations and degrees of freedom (d. o. f.), and for proper handling of constraints that could change at loop-rate, including avoidance of multiple obstacles and joint limits. In addition, analytical solution methods for linear or planar motion constraints for tools or non-holonomic constraints for mobile platforms have been developed. For example, Fig 1 shows a sample problem involving a mobile platform with a car-like kinematics, i.e., with a very strong non-holonomic constraint, supporting a manipulator that moves to acquire a tool, performs tasks with the tool, and then releases the tool. The significance of the approach successfully handling non-holonomic constraints (which due to their non-integrable nature cannot be handled by direct unconstrained approaches) simultaneously with configuration changes through acquisition of tools and time-varying constraints on the manipulator and/or tool reinforces the capability of the novel constrained optimization scheme to handle widely varying types and numbers of constraints and criteria, that can change at loop rate, through a single resolution code. A journal article presenting the developed methodologies in this constrained optimization area is in review for publication [8].

The fourth area of emphasis has involved the completion of the capability of simultaneously handling constrained motion planning for multiple robots, possibly interacting through their respective time-varying constraints, for example when some of the robots become "obstacles" for the others due to their motion in a common workspace, thereby generating time-varying obstacle-avoidance constraints that affect the robots' motions in real-time. Figure 2 shows an example of such a multi-manipulator resolution by the constrained motion planning system, in which two of the manipulators (bases at the top of the workspace cell) represent surgical manipulators remotely operated by a surgeon (and therefore unpredictable in their actual motion) while two other manipulators are autonomous "assistants," responsible for tasks such as holding supplies (sponges, forceps, etc.) and tools while the surgeon performs surgical tasks. As such, the assistants must avoid collisions with the surgeon-driven arms, while also maintaining specific end-effector positions and/or orientations to perform their holding tasks. In this particular example, the surgeon's remote manipulators include 8 d. o. f. each, while the assistants' manipulators are 12 d. o. f. each, illustrating the capability of the code to now handle simultaneously multiple systems with a very large combined number of degrees of freedom and constraints. These capabilities were presented this year in two special sessions of dedicated conferences [9, 10].

On-going activities are focusing on our fifth area of emphasis, which involves non-linear task objective criteria. We have developed analytical frameworks for such non-linear approaches and corresponding analytical resolution strategies. Coding of the algorithms is proceeding as planned, with experimental investigations of complexity and bounded time capability through comparative computational experiments to assess preferred resolution approaches for these objective functions.

Planned activities:

During the remainder of this third year of activities, our focus will be on the continuation of the investigations related to non-linear criteria handling, in particular for conditions representative of systems for which no ready solutions exist today, for example in the area of flexible manipulators and hydraulically powered systems. We are also focusing on proof-of-principle experiments using kinematics configurations of existing D&D robots at ORNL, including implementation on an actual robot control system to demonstrate pervasiveness of the single-code

approach. Figure 3 shows a simulated Titan II robot (widely used in DOE D&D activities) on which the developed algorithms are implemented to investigate performance of tasks that otherwise were found impossible to perform.

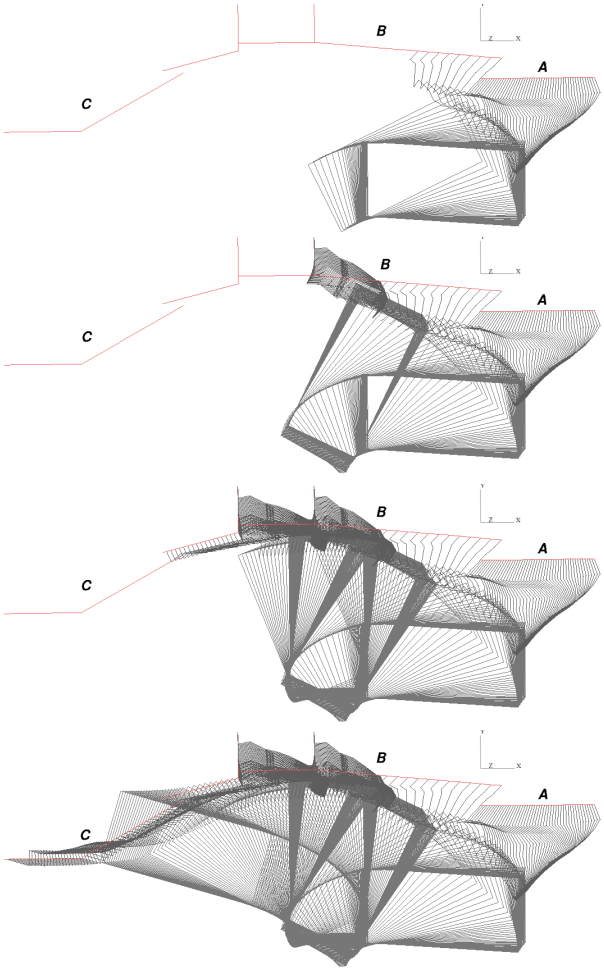

Figure 1 – Motion of a 13 d. o. f. mobile manipulator system with a non-holonomic constraint on the mobile platform, during a hole-drilling task with tool change and tool constraints. The resulting platform motion exhibits the typical cusp-like motion of car-like systems, while the overall system adapts in real-time to the changes of tools and tool-imposed constraints.

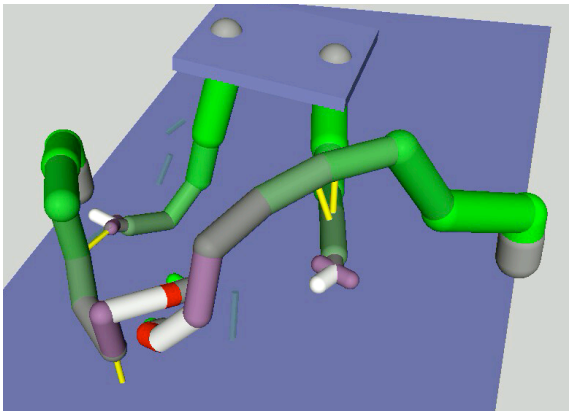

Figure 2 – Example of the simultaneous motion planning of 4 manipulators in the context of robotic surgery. Two 8 d. o. f. manipulators are surgical systems remotely operated by a surgeon, while the other two 13 d. o. f. manipulators are autonomous "assistant" arms that must maintain tool or supply holding tasks while avoiding collisions with the surgeon-driven arms. The yellow "ticks" indicate obstacle avoidance constraints that are active at the particular time-step due to the respective motions of the 4 manipulators.

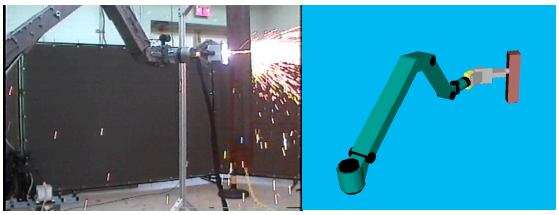

Figure 3 – Real and simulated Titan II manipulator during a plasma-cutting task typical of DOE D&D activities. Using the additional d. o. f. provided by the addition of an articulated tool, the system performs a surface-following task, which otherwise would not be feasible.

Information access:

Further background and details of the methodologies can be found in the following publications:

1 - F. G. Pin et al., *Journal of Robotic Systems* **13** (11), 723–736 (1996).

2 - F. G. Pin and F. A. Tulloch. Proceedings of the 1996 IEEE International Conference on Robotics and Automation, Minneapolis, Minnesota, April 22–28, 1996, Vol. 1, 468–473.

3 - F. G. Pin et al. Proceedings of the IEEE International Conference on Robotics and Automation, San Diego, California, May 8–13, 1994, 96–102.

4 - F. G. Pin et al. Proceedings of the 1997 IEEE International Conference on Robotics and Automation, Albuquerque, New Mexico, April 21–27, 1997, Vol. 4, 2914–2919.

5 - K. N. Fisher, D. L. Jung, F. G. Pin and W. E. Dixon, "JFKengine: A Jacobian and Forward Kinematics Generator," ORNL/TM-2002/230 (2002).

6 - F. G. Pin, K. N. Fisher and D. L. Jung, "Automated Generation of the Forward Kinematics Equations and Jacobian for Modular and Re-configurable Robots", in revision for *IEEE Trans. Robotics and Automation*.
7 - http://pobox.com/~djung/jfkengine
8 - F. G. Pin and D. L. Jung, "Motion Planning of Kinematically Redundant Robots with Time-varying Constraints," in review for *IEEE Trans. Robotics and Automation*.
9 - D. L. Jung, W. E. Dixon, and F. G. Pin, "Automated Kinematic Generator for Surgical Robotics Systems," Medecine Meets Virtual Reality, Vol. 12, J. D. Westwood et al. (Eds.), IOS Press, 2004.
10 - F. G. Pin, L. J. Love, and D. L. Jung, "Automated Kinematic Equations Generation and Constrained Motion Planning Resolution for Modular and Reconfigurable Robots." Proceedings of the American Chemical Society Conference, Anaheim, California, March 28-April 1, 2004.