Literature Review for SMORES Verification

Tarik Tosun

January 12, 2015

In this paper, we present a system that allows users to construct modular robot behaviors by composing existing ones. The system verifies new behaviors while they are being developed, providing warnings that allowing the user to quickly identify common problems before performing an expensive dynamic simulation. The problems the system can identify are self-collision, loss of quasi-static stability, and the presence of unexpected behaviors.

Composition and Verification in Modular Robotics

Distributed Watchpoints: Debugging Large Multi-Robot Systems [3]

Presents a message-passing scheme that implements watchpoints (conditional breakpoints) on a distributed system of modular robots. Watchpoints are specified in a description language based on LTL. Focuses on detecting errors during execution of a program on a distributed system.

A Self-Reconfigurable Modular Robot: Reconfiguration Planning and Experiments [7]

Presents a layered motion planner for modular robot cluster flow ("cluster flow" is a mode of locomotion). Cube-shaped metamodules move from the rear of a cluster to the front. At the lowest layer, individual module movements are verified by checking for self-collision and cluster connectivity.

Reconfiguration Planning for Modular Self-reconfigurable Robots [2]

This is the Ph.D thesis of Arancha Casal, who Mark worked with at PARC. Unfortunately I have not been able to find a copy of it yet, but Mark told me it covers self-collision and quasi-static stability detection for modular robots, so it is very relevant to our paper. I have emailed her old advisor (Jean-Claude Latombe, who also advised Mark) asking for a copy.

Composition and Verification in Robotics

PPR

The PPR (Printable Programmable Robots) project is developing manufacturing, design, and programming techniques to allow novice users to easily create robots. Design techniques focus on composing new robots from modular elements in a library. Verification is not stressed.

A Design Environment for the Rapid Specification and Fabrication of Printable Robots [5]

Presents a design environment to allow casual users to quickly and easily create custom robots. New robots are built by composing mechanical, electrical, and software subsystems. Supporting software is automatically co-generated when the user composes functional elements (such as legs and a body).

Verification of new designs and behaviors is not stressed. Elements are connected at prescribed parametric interfaces. Composition focuses on structure (e.g. combine motors, propellers, and microprocessors to make a quadrotor) rather than task-facing functionality (e.g. build me a robot that maintains quasi-static stability while walking, and can get up onto a table).

While the system has modular components, the robots are not "modular robots" in our sense of the word. Robots in this paper are manufactured by cutting and folding plastic and paper sheet material. However, the design environment is not specifically limited to these design techniques.

Demo abstract: ROSLab - A modular programming environment for robotic applications [1]

This brief abstract introduces ROSLab, the block-based modular programming language developed for PPR and referenced in the above paper.

On Embeddability of Modular Robot Designs [4]

This is my paper (to appear in ICRA 2015 (hopefully)). We formalize the notion of design embeddability, and present an algorithm to efficiently detect it. If a group of modules capable of a kinematic task (for example, a planar arm) can be realized in a larger group of modules configured for a different kinematic task, we say that the larger group *embeds* the smaller. If one design embeds another, it is guaranteed to replicate the kinematics of the other design. However, workspace constraints (self-collision) and dynamics/forces are not considered.

Techniques and Algorithms

FCL: A General Purpose Library for Collision and Proximity Detection [6]

The Flexible Collision Library (FCL) is used by the ROS MoveIt package, and seems to be one of the best freely available packages for collision detection. The paper provides a nice overview of state-of-the-art collision detection techniques, many of which are provided by FCL.

References

[1] Nicola Bezzo, Junkil Park, Andrew King, Peter Gebhard, Radoslav Ivanov, and Insup Lee. Demo abstract: Roslaba modular programming environment for robotic applications. In *Cyber-Physical Systems (ICCPS), 2014 ACM/IEEE International Conference on*, pages 214–214. IEEE, 2014.

[2] Aranzazu Casal. *Reconfiguration Planning for Modular Self-reconfigurable Robots*. PhD thesis, Stanford University, Stanford, CA, USA, 2002. AAI3038069.

[3] Michael De Rosa, Seth Goldstein, Peter Lee, Jason Campbell, and Padmanabhan Pillai. Distributed watchpoints: Debugging large modular robot systems. *The International Journal of Robotics Research*, 27(3-4):315–329, 2008.

[4] Yannis Mantzouratos, Tarik Tosun, Yim Khanna, Sanjeev, and Mark. On embeddability of modular robot designs. In *IEEE International Conference on Robotics and Automation*, 2014.

[5] Ankur Mehta, Nicola Bezzo, Peter Gebhard, Byoungkwon An, Vijay Kumar, Insup Lee, and Daniela Rus. A design environment for the rapid specification and fabrication of printable robots. In *nternational Symposium on Experimental Robotics (ISER), Marrakech, Morocco*, 2014.

[6] Jia Pan, Sachin Chitta, and Dinesh Manocha. Fcl: A general purpose library for collision and proximity queries. In *Robotics and Automation (ICRA), 2012 IEEE International Conference on*, pages 3859–3866. IEEE, 2012.

[7] Eiichi Yoshida, Satoshi Matura, Akiya Kamimura, Kohji Tomita, Haruhisa Kurokawa, and Shigeru Kokaji. A self-reconfigurable modular robot: Reconfiguration planning and experiments. *The International Journal of Robotics Research*, 21(10-11):903–915, 2002.