Distributed Control for a Modular, Reconfigurable Cliff Robot

Paolo Pirjanian, Chris Leger, Erik Mumm*, Brett Kennedy, Mike Garrett, Hrand Aghazarian, Shane Farritor*, Paul Schenker

Jet Propulsion Laboratory, California Institute of Technology,
4800 Oak Grove Drive/MS 82-105
Pasadena, California 91109

*Department of Mechanical Engineering,
University of Nebraska-Lincoln,
Lincoln, NE 68502

Abstract

We have developed a cliff robot that is capable of descending into a cliff and autonomous navigation to way-points on the cliff wall. This aggressive mobility system consists of an ensemble of three tethered robots, which cooperate under tight coordinated control and collective state estimation. The distributed task is described as a behavior network, which consists of a network of controllers spread across the robots and which interact through communication links to achieve a collective control objective. Fielded experimental results show that the cliff robot is capable of navigating to designated way-points on a cliff wall using the proposed control scheme.

1 Introduction

In this paper, we describe a robotic system for access to high-risk locations on cliff-sides, where conventional robot designs fail. The system concept consists of a modular system that can break into three robotic entities, where the two serve as anchoring points at the top of the cliff and assist to lower and guide the third one down the cliff-side, using controlled tethers. This work is part of a larger endeavor, at JPL's Planetary Robotics Lab. in developing reconfigurable robotics systems for all-terrain exploration that consist of multiple modular robotic assets that can be reconfigured to perform challenging collective tasks under changing requirements [1]. Our research is focused on 'system-level' modularity, where each 'module' is a self-contained robot with its own sensors, actuators, computational resources, and power, and is capable of autonomous operations. The modules can separate and reconnect to form different configurations suitable for specific tasks. The focus of this work is different from the more conventional efforts in modular robotics, which pursue building a reconfigurable robot consisting of a small number of

Figure 1 Cliff-bot conceptual CAD drawings illustrating three phases of approach, separation, and deployment for cliff access.

simple modules that do not by themselves serve any interesting function (see [4, 5]). While component-level modularity has the potential to develop the ultimate reconfigurable systems, system-level modularity has more immediate practical use for many application areas including Mars exploration, search and rescue, and access to cliffs and craters. Yet the main research issues regarding self-reconfiguration, task allocation, planning and coordination of activities, collective estimation, distributed control, etc. are common to both approaches to modular robotics. Additional challenges for component-level modularity include micro-mechanisms for mechanical, data, and power connectors and interfaces between the modules. Issues regarding connectors are less of a challenge for system-level modularity due to the advantage gained by working with larger scale components. To some extent, the cliff-bot faces similar challenges to its priors such as CMU's Dante [7] but provides further value in its exploitation of modularity, multi-robot control, and self-configuration capabilities.

2 Cliff-bot conceptual overview

Recent orbital imagery of the Martian surface indicates potential for important science near cliff edges, where water may have once flowed. Access to these high-risk regions can have high science return value and requires aggressive mobility systems beyond that provided by conventional rover designs. Accessing the cliff wall from below requires a system that can safely climb steep slopes while maintaining tip over and traction stability. In fiscal year 2000, we developed a rover for all-terrain exploration (ATE), which is capable of traversing moderate slopes of up to 50 degrees. The ATE rover uses a kinematic and quasi-static model of the vehicle to shift its center of gravity in response to the traversed slope as analyzed by stereo-vision sensing and terrain modeling [1]. Climbing steeper slopes may require alternative robot designs that provide, e.g., legged locomotion, dexterity, etc. Accessing the cliff from above offers the opportunity for using a tether as a safety line and means of stabilization. This approach also falls within our research objectives in system-level modular and self-reconfigurable robotics.

Figure 1 illustrates the cliff-bot system concept using CAD drawings. The modules mate, using connectors that hold deployable tethers, for long-range traverse from the landing site to the cliff-edge. At the cliff-edge the ensemble of modules/robots finds a deployment site that allows safe access to the cliff wall. After getting in position, the robots at the two ends 'anchor' themselves and deploy their tethers, connected to the third robot, which 'rappels' down the cliff side under assistance from the two anchor-bots.

2.1 Hardware test-bed

This year, we have focused our work on the last, and probably most challenging, phase for cooperative cliff descent. In addition to descending the cliff wall, our objective was to enable access to any designated way-point on the cliff-bot, within the workspace defined by the position of the anchor-bots and the tether lengths. In other words, the cliff-bot can navigate around on the cliff face as other conventional rovers navigate on a flat surface. This capability is valuable since it enables access to interesting targets on the cliff.

We used a physics-based simulator (see section 2.2) to investigate and drive our hardware design

Figure 2 Cliff-bot test-bed: Multi-robot ensemble of three tethered robots consisting of two anchor-bots plus a cliff-bot. The tethers from the anchor-bots to the cliff-bot are not visible in this photo. The ramp is used for indoor development and testing. Validation is performed in outdoor settings on unmodified cliff face.

requirements and control strategies (see section 3.2). The hardware test-bed, as depicted in figure 2, consists of a 4-wheeled robot used as the *cliff-bot* and two railed winch assemblies to emulate the *anchor-bots*. Each robot has a PC-104+ stack for on-board computation, a set of dedicated sensors, and radio modems for wireless communications. The anchor-bots have encoders to support closed-loop velocity and position control of the tethers. The tethers are attached to the cliff-bot at a gimbal instrumented with sensors to measure the direction vector of each vector. This information is accessed by the anchor-bots to close the feedback loop for coordinated velocity control.

2.2 Physics-based simulation

Initially, we prototyped several aspects of the rover-tether ensemble using Darwin2K [6], a dynamic simulation and automated design package. The rover and tethers were simulated using a full dynamic model: that is, the motion of the rover was computed based on its inertial properties, kinematics, and the forces exerted on the rover by drive and steering actuators, the two tethers, and wheel-ground contacts. We tested various tether attachment locations on the rover, observing the

Figure 3 Dynamic simulation of the cliff-bot and the two anchor-bots with Darwin2k, used for investigating hardware design requirements and distributed control strategies.

effects of tether location on heading and front-to-back stability Figure 3). For example, if the tethers do not connect to the rover at a single point, then unequal tether loading will apply a torque to the rover which will prevent the rover from hanging in a stable orientation as the rover moves horizontally across the cliff face, while tether attachment points that are too high or too low can cause the rover to easily tip forward or back if the tether and wheel velocities are not well-synchronized.

We also developed and validated the first version of the velocity-based tether controller using the simulator, and performed a sensitivity analysis to determine the minimum acceptable synchronization rate between the tethers and the rover.

3 Distributed estimation and control

For safe and stable navigation on the cliff-side the cliff-bot and the two anchor-bots are required to tightly coordinate their activities through collective estimation and distributed, synchronized control. We have used CAMPOUT [2,3], Control Architecture for Multi-robot Planetary OUTposts, as the underlying architecture of this work.

3.1 CAMPOUT

CAMPOUT consists of a number of key mechanisms and architectural components that facilitate development of multi-robot systems for cooperative and coordinated activities. These include the following:

Modular task decomposition: Following a behavior-based methodology, CAMPOUT provides fundamental building blocks for describing a system in terms of task-achieving modules known as a behavior-producing module or a behavior, for short. While a behavior provides a convenient and efficient architectural substrate to encapsulate perception and

Figure 4 Conceptual diagram of the behavior networks used for controlling an ensemble of robots within CAMPOUT.

action, it is its interactions with other behavior-producing modules that generate the final behavior of the system. In its current implementation, task decomposition is done by hand and encoded in a script/plan, which is then executed by the agents. We are currently working towards extending CAMPOUT with automated planning of joint team activities.

Behavior coordination mechanisms: A system's behavior is described as a network of behaviors (Fig. 4) that interact with each other and with the environment through sensors and effectors. The behavior interactions are regulated through behavior coordination mechanisms (BCMs). The BCMs are used to restrict and control the behavioral interactions so that the system can operate according to its specifications. In other words, the BCMs are used to ensure that the behaviors interact in a desired and consistent manner.

Group coordination: CAMPOUT uses the same task decomposition scheme and representation to describe group activities, the deference being that the nodes (behaviors) of the network are distributed across the group of robots and connected through implicit (supported by external sensing) or explicit (radio) communication.

Communications infrastructure: Relying on sensing for communication is not a feasible solution because sensing can be unreliable, the sensory envelope is often more range-limited than radio communication, and it requires more computation for processing than is usually the case with radio communication. CAMPOUT provides a software infrastructure that

allows transparent inter-robot communication, which enables the robots to share state information, sensors, actuators, etc. The communications infrastructure allows a behavior network to be seamlessly distributed across a network of robots.

Distributed, multi-robot control is encoded in a *behavior network* (figure 4 and 5), which is a network of behaviors that span over the robots, taking input from any robot's sensors and controlling the action of one or more of the robots. The links between behaviors across different robots is provided through the (wireless) communications infrastructure. Lower-level behaviors are combined into higher-level behaviors using behavior coordination mechanisms. Thus a higher-level behavior can be coordinating the activities of multiple behaviors that run on different robots. In the following section, we see how these mechanisms are used to implement the coordination necessary for the cliff-bot.

3.2 Collective way-point navigation

Without coordinated assistance from the anchor-bots, the cliff-bot will not be able to traverse the side of a cliff with slopes of up to 75 degrees, due to tip-over and loss of traction considerations. Most mobility platforms are not able to safely traverse moderate slopes of as low as 35-40 degrees without adaptive CG and traction control. In fact, we have previously developed such techniques for traversing slopes of up to 50 degrees [1].

To navigate on a steep cliff-side wall, the cliff-bot and the two anchor-bots are required to tightly coordinate their activities. Each anchor-bot must adjust the velocity of its tether to the velocity of the cliff-bot. The relationship between the velocity of the tethers is derived from the kinematics constraints of the system and is determined by the projection of the cliff-bot's velocity, v onto the directional vector of each tether r_{left} and r_{right} . The velocity vector v is easily estimated using the cliff-bot's position encoders and inertial navigation system but has to be shared with the anchor-bots. Also the tether vectors r_{left} and r_{right} are estimated based on sensor data on the cliff-bot, which is instrumented with resolvers to measure the tilt and pan (or pitch and yaw) angles of the attached tethers. So the yaw and pitch angles must be shared with the anchor-bots, which is transparently provided through CAMPOUT's communications infrastructure. Using CAMPOUT, each anchor-bot can access the sensor readings (tether vectors) or fused estimates (e.g., velocity vector) on any robot. To control the tether velocities each anchor-bot is instrumented with one encoder on the driving motor and another on the pay-out mechanism that measures the tether length.

In theory, the coordination of the velocities of anchor-bots with the cliff-bot boils down to a straight-forward projection of velocities onto the tether vectors. However, to realize this capability in practice the three robots are required to perform distributed control and estimation. This requires access to data that is distributed across the robots,

Figure 5 A subset of the behavior network for collective cliff-descent illustrating sub-system for controlling the velocity of anchor-bot 1. The arrows represent data links between local blocks as well as remote components (behaviors, sensors, actuators) thus spanning a behavior network across the team of robots.

Figure 6 Snapshots taken during the field-trials. Cliff-bot descending down the cliff to depths of about 15m.

which is facilitated by CAMPOUT. Further, there are a number of failure modes that are not handled by this straight-forward approach. One particular failure mode is when the cliff-bot attempts to ascend a steep slope but cannot create enough torque to create a momentum, which implies that its velocity never exceeds 0. This means that the projected velocities for the winches remain 0. When this occurs, we would like the anchor-bots to proactively, rather than reactively, assist the cliff-bot to initiate its velocity by hauling it for a short distance. Once the cliff-bot is in motion the anchor-bots will resume adjusting their velocities in a reactive manner. CAMPOUT's task decomposition and distributed task description capabilities facilitate the development of such distributed control strategies using powerful tools to simplify the problem. Many other failure modes exist that again require both collective estimation and distributed control. The architectural facilities provided by CAMPOUT support the development of such complex systems.

The block diagram in Figure 5 shows a portion of the behavior network that is used for controlling the velocity of anchor-bot 1. The portion used for controlling anchor-bot 2 is the same but with different links between the behaviors, sensors, and actuators. The four main behaviors that control the velocity of the tether are *Match Velocity*, *Haul*, *Maintain (tether) Tension*, and *Stability*. *Match Velocity* adjusts the tether velocity based on the cliff-bot's velocity. *Haul* is triggered either when the other anchor-bot is triggered or when the velocity of the cliff-bot remains zero after a time-out period

Figure 7 Snapshots taken during the field-trials. Cliff-bot moving diagonally towards a designated way-point and assisted by the two anchor-bots shown in the background.

from the time where a navigation command was issued. When *Haul* is triggered, it gives the cliff-bot a "push" (or actually pull) in the right direction to help it initiate motion in the up-hill direction. *Maintain Tension* controls the tether velocity to maintain a safe tension on the tether. *Stability* consists of a number of modules that check various stability requirements (not covered here) and triggers a safe mode where all commands are aborted. These behaviors are in part coordinated using a priority-based arbitration mechanism (denoted with the circle labeled "P") and in part by a summation operator. The boxes with numbers 1, 2, and 3 indicate the priority of each behavior with *Stability* having the highest priority of 1. The links to the behaviors indicate exchange of sensory, perceptual, and other state information across the network of behaviors and robots, as supported by CAMPOUT's communications facilities. In fact, for the behaviors there is no perceivable distinction between local and remote data since all data transfer occurs transparently once the transfer has been initiated using the publish and subscribe methods.

4 Experimental results

A large number of successful experiments were conducted in the Arroyo Seco at JPL on natural, challenging hill-sides (see Figures 6, 7). The experiments included way-point based navigation on slopes > 70 degrees over distances of 10-15 meters (distance as determined by physical site access restrictions). The "anchor-bots" worked under a collective estimation and distributed control with the

descending "cliff-bot" to enable a robust, fault-free traverse in arbitrary directions.

In the experiments, we demonstrated autonomous cliff descent using the proposed decentralized control scheme under CAMPOUT. During the field-trials, the cliff-bot accessed all way-points by driving toward a designated way-point while being assisted by the two anchor-bots. These results are based on quantitative evaluation of the results. Qualitative experimental results are underway. We also ran several experiments without the implemented control loop, which verified that lacking a tight coordination of robot actions and exchange of sensor knowledge, the cliff-bot would be unstable and error-prone. These experiments verify that cliff navigation fails in the absence of effective synchronized, distributed control between the cliff-bot and its two tethered anchor-bots.

5 Conclusions and future work

We have used the cliff-bot concept as a test-bed to investigate research issues in system-level modular robotics, self-reconfiguration, and multi-robot coordination. The cliff-bot test-bed and its application to tasks such as cliff access, search and rescue, etc. provide a rich set of challenges that we are planning to investigate. Short-term plans include development of robust behaviors for autonomous obstacle avoidance, target approach, path planning, etc., which will be based on the core capabilities for autonomous navigation on the cliff-face. Interesting issues to investigate include collective estimation and data fusion for terrain mapping based on visual (or other) data collected from all robots. The anchor-bots, for instance, can be used for wide-baseline stereo vision and enable high-accuracy 3D terrain modeling. These capabilities will be valuable and key for search and rescue and similar applications, which require access to difficult terrain.

On the long-term basis, we are planning to investigate approaches for dynamic task allocation, distributed task description, and more. Currently, CAMPOUT provides communication facilities for sharing of state information across robots and it uses a behavior network for representation and execution of group activities in the same way that it represents the activities of a single robot. In our research, we have shown that CAMPOUT helps to bridge the gap

between multiple robots and provides a level of abstraction that enables us to develop multi-robot software in a manner much similar to what we use for single robot software development.

For future work, we are interested in further bridging this gap by extending CAMPOUT with task planning capabilities and automation of group activity generation. Currently, we use CAMPOUT's facilities to handcraft the behavior network that represents a group activity. We are investigating approaches to automate this process so that behavior networks can be formed and implemented automatically.

6 Acknowledgements

The research described in this paper was carried out at the Jet Propulsion Laboratory, California Institute of Technology, under a contract with the Cross Enterprise Technology Development Program of the National Aeronautics and Space Administration. The authors would like to thank the many people at the Planetary Robotics Lab, JPL that have provided valuable contributions to the described work.

References

- [1] P. S. Schenker, P. Pirjanian, et. al., "Reconfigurable robots for all terrain exploration," in SPIE Vol. 4196, Sensor Fusion and Decentralized Control in Robotic Systems III, 15 pp., Nov. 5-8, 2000.
- [2] P. Pirjanian, T.L. Huntsberger, A. Trebi-Ollennu, H. Aghazarian, H. Das, S. Joshi, and P.S. Schenker, "CAMPOUT: A control architecture for multi-robot planetary outposts," in Proc. SPIE Sensor Fusion and Decentralized Control in Robotic Systems III, Vol. 4196, Boston, MA, Nov. 2000, pp. 221-230.
- [3] P. Pirjanian, T.L. Huntsberger, Paul S. Schenker, "Development of CAMPOUT and its further applications to planetary rover operations: a multi-robot control architecture," in Proc. SPIE on Sensor Fusion and Decentralized Control Nov. 2001.
- [4] M. Yim et al. "PolyBot: A modular reconfigurable robot," IEEE Int. Conference on Robotics and Automation (ICRA) 2000.
- [5] D. Rus, "Self-reconfiguring robots." IEEE Intelligent Systems 134 (July-August): 2-4, 1998.
- [6] C. Leger. "Darwin2K: An Evolutionary Approach to Automated Design for Robotics". Kluwer Academic Publishers, 2000.
- [7] D. Apostolopoulos, and J. Bares, "Locomotion Configuration of a Robust Rappelling Robot," Proceedings of the 1995 IEEE/RSJ IROS '95, Vol. 3, August, 1995, pp. 280 - 284.