Action Module Planning and its Application to an Experimental Climbing Robot

David M. Bevly
Mechanical Engineering
Stanford University
Stanford, CA 94305

Shane Farritor
Mechanical Engineering
University of Nebraska
Lincoln, NE 68516

Steven Dubowsky
Mechanical Engineering
Massachusetts Institute of Technology
Cambridge, MA 02139

Abstract

This paper presents the application of an action module planning method to an experimental climbing robot named LIBRA. The method searches for a sequence of physically realizable actions, called action modules, to produce a plan for a given task. The search is performed with a hierarchical selection process that uses task and configuration filters to reduce the action module inventory to a reasonable search space. Then, a genetic algorithm search finds a sequence of actions that allows the robot to complete the task without violating any physical constraints. The results for the LIBRA climbing robot show the method is able to produce effective plans.

1 Introduction

Mobile multi-limbed field robots could play an important role in such tasks as space exploration, nuclear site clean up, bomb disposal, and infrastructure inspection and maintenance. Currently the use of robotic systems in such unstructured environments has been limited for a number of reasons, including the lack of effective methods to automatically generate feasible robot plans.

Numerous planning methods have been proposed for robotics [1]. One method used for mobile robots is behavior control [2]. This layered technique was used on a climbing robot that uses friction to allow the robot to travel through ducts [3]. However in this work, task-specific (low-level) actions were necessary to allow the robot to maneuver around difficult sections of the duct.

The use of potential fields for motion planning and obstacle avoidance of mobile robots has also been proposed [4]. In such methods the physical capability of the robot such as actuator saturation are not specifically considered. Planning methods that provide a representation of physical constraints such as power consumption, actuator saturation, workspace, and obstacles have been developed [5,6]. A method that considers the physical constraints of the system and its environment to find feasible task plans has been proposed [7]. The method assembles action modules to produce a task plan for a robot in complex, unstructured environments. This requires knowledge of the environment, possibly from on-board sensors [8]. This method has similarities to an approach that searches for a sequence of foot placements to allow the robot to walk through an environment [9]. However the action module method considers more general tasks than walking, such as planetary space rovers gathering rock samples [10].

In the action module planning method a set, or inventory, of robot actions (called action modules) are assembled in a physically realizable sequence for a given task. Since the number of possible plans that can be assembled is extremely large, a hierarchical procedure is used to automatically search for a feasible plan with a reasonable amount of effort. Note that computational efficiency is important because the computational power carried by many field robotic systems is limited [11]. The final step in the method uses a genetic algorithm to search for an acceptable plan.

Here the action module planning methodology is applied to the MIT LIBRA laboratory climbing robot to study its practical validity [12]. LIBRA climbs a series of pegs and presents some challenging planning problems such as tight physical constraints. Previous work has shown that actuator saturation and kinematic reach present serious limitations to such systems [6]. Since these limitations are explicitly considered by the action module methodology it is able to produce effective plans for the LIBRA. The control of this system is also difficult [13,14]. However these issues are beyond the scope of this paper.

LIBRA is a planar, three legged climbing robot, see Figure 1. Each 32 cm limb of the 40 Newton robot, consists of two joints driven by highly geared motors. Each joint has 110° of rotation and an actuator saturation limit of 3000 oz-in. Specific details on the design and properties of the climbing robot LIBRA can be found in [12,13]. A pegboard was built such that the pegs, used for LIBRA to climb, could be placed in several patterns. This was done to study the planning of LIBRA on various tasks.

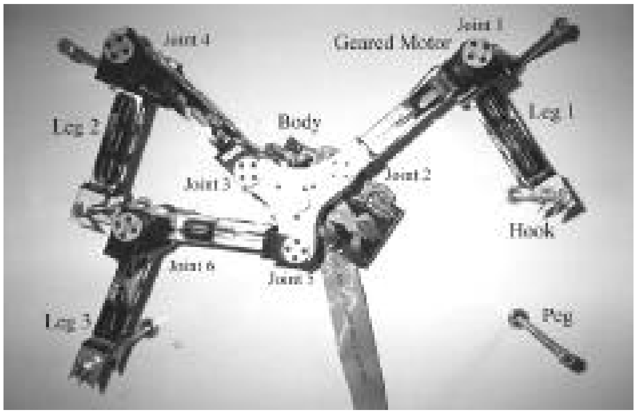

Figure 1. LIBRA

2 The Action Planning of LIBRA

Sets of simple tests and filters and a physics-based simulation are used by the hierarchical planning method to plan the actions of the LIBRA.

2.1 Action Module Inventory and Module Filters

The planning method uses an inventory of action modules. The modules are physically realizable actions of a robot, see Figure 2 and Table 1.

Table 1. LIBRA Action Module Inventory

#	Description	#	Description
001	Move Body +x	N00	Leg N Release Peg
002	Move Body –x	N91	Leg N Grab Nearest Open Peg
003	Move Body +y	N92	Leg N Grab 2nd Nearest Open Peg
004	Move Body –y	N93	Leg N Grab 3rd Nearest Open Peg
005	Rotate Body +θ	N99	Leg N Grab a Random Open Peg
006	Rotate Body -θ	NXX	Leg N Grab Peg XX

LIBRA with its three legs has 21+3p action modules, where p is the number of pegs in the task. For example, a climbing problem with 30 pegs will create an action module inventory of 111 actions.

The size of the search space is:

$$D=N^m \tag{1}$$

Where: D = number of possible action plans
N = number of available modules
m = number of action modules used

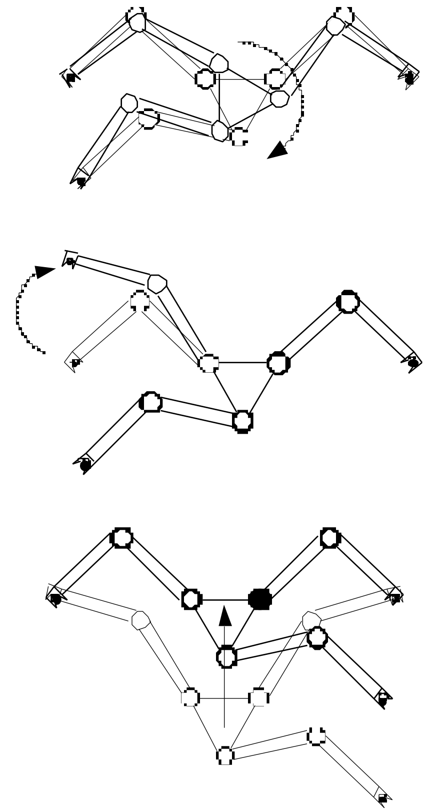

Figure 2. Examples of LIBRA Action Modules

A task that required 60 actions to complete, such as the task presented in Section 3.2, and used all 111 actions would correspond to a search space that contains over 10^{197} possible plans. Filters reduce the size of the search space by eliminating non-feasible actions from the inventory. For example, plans need only consider pegs within the kinematic reach of LIBRA at any time. For more complex systems the size of the action module inventory can be even larger. For such systems, the use of task and configuration filters used in a hierarchical fashion is essential [7]. Small reductions in the number of actions modules greatly reduces the number of plans that must be considered by the more computationally expensive simulation based tests later in the search. Table 1 represents the reduced inventory for LIBRA.

In the method, the action modules are assembled into a sequence to create an action plan as shown in Figure 3. The final assembly of the plan, from the reduced inventory, uses a genetic algorithm [10] as described in section 2.2.

The inventory of action modules can be improved by the use of learned high level modules. Repeated sequences of action modules are grouped into higher level modules such as those shown in Figure 3. Adding higher level modules to the inventory can drastically reduce the search space by decreasing the number of action modules (m in equation 1). The method also learns its own higher-level action modules once repeated sequences appear in the action plan. This repeated sequence can then be extracted and added to the inventory [8].

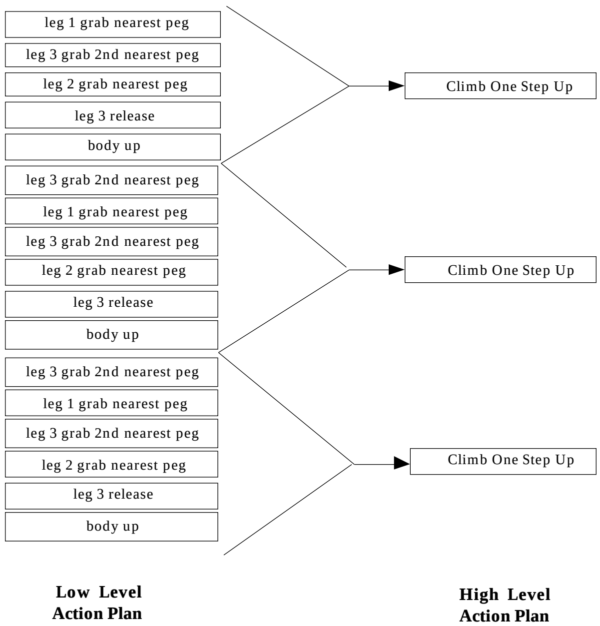

Figure 3. Low and High Level Action Modules

2.2 Plan Assembly by a GA Search

A genetic algorithm (GA) searches candidate action plans, represented as a chromosome in the form of a list of action modules. As in classical GA methodology, crossover and mutation are used to evolve a successful action plan [15].

Evolving plans are ranked among each GA generation by a fitness score based on its performance in a physics-based simulation. Plans for crossover "mating" are selected using their fitness score. This ensures that the best plans mate more often to create stronger plans. A well designed fitness function is crucial. The fitness function used for LIBRA is:

$$f = w_1|D_{body}|+w_2|D_{leg\ i}|-w_3(P)+w_4(d)-w_5(LP)-w_6(S) \quad (2)$$

where: D_{body} = the distance the body of LIBRA travels toward the target

$D_{leg\ i}$ = the distance leg i travels toward the target

P = power consumed

d = 1 if the target (or sub-goal) is reached and 0 otherwise

LP = length of plan

S = Stability factor

w_i = weighting factor

It awards plans that move the LIBRA's legs. This encourages the LIBRA to grasp pegs on its way toward the target, while the remainder of the function constrains such factors as power consumption and static stability.

Mating between two high-scoring fitness plans is accomplished by a method referred to as tail crossover, see Figure 5. Tail crossover acts to maintain the successful first part of each plan by switching the two lover halves of each plan. The crossover points C1 and C2 (shown in Figure 5) are chosen randomly near the point of failure. This allows the GA to operate near the critical region of each plan and build upon the successful portions of plans. However, this causes limited backtracking and creates the need for sub-goals or high-level action modules to prevent the planner from being caught in local minimum.

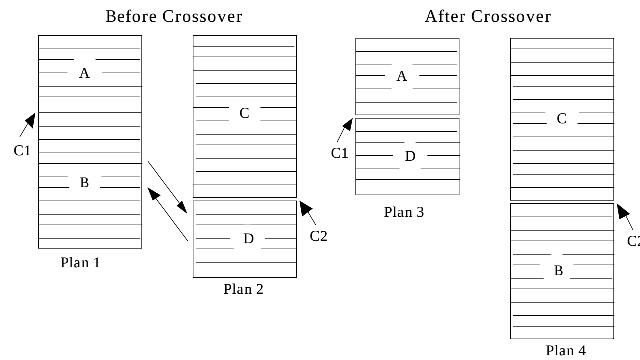

Figure 5. Genetic Crossover [8]

Mutation is a process used by the GA to insure population diversity. It was accomplished by replacing a random action module in a plan with another random action. Mutation was applied to approximately 5% of the plans in each generation.

2.3 Physics-Based Simulation

A physics-based simulation was written to check for the violation of physical constraints of the robot during execution of a plan and to assign a fitness score to each plan for the GA search. It checks physical constraints such as joint and kinematic reach limits, end effector forces, power consumption, limb interference, stability and interference between the limbs the body and the pegs. The simulation also has a graphical interface for animating execution of plans.

The actuator saturation constraints are modeled using the method in [6]. Power consumption is the calculated by assuming that it is dominated by the static motor torques required to support the system [5].

3. Action Module Planning Results

3.1 Ladder Task

Figure 6 shows the nominal gait developed by the method for climbing the ladder pattern of pegs. Although other gaits exist, this gait provides higher stability and lower power consumption.

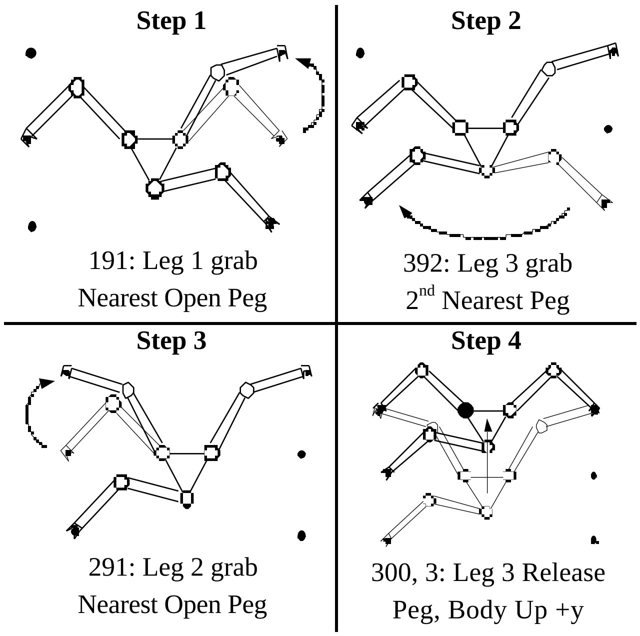

Figure 6. Nominal Gait for the Ladder Climb Task

This gait was implemented on the experimental system and the results are shown in Figure 7. This experimental data shows the path taken by each limb and the center of the LIBRA body during execution.

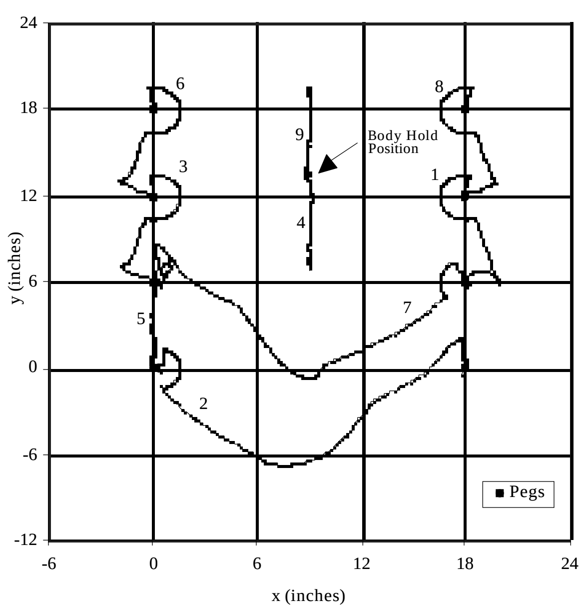

Figure 7. Experimental Implementation

3.2 Modified Ladder Task

Figure 8 shows a modified ladder task, that has four pegs removed from the right side. The method could not find a plan to perform the climb when body rotation action modules (005 and 006) are not included in the inventory. However, with modules 005 and 006 a successful plan is found.

Figure 8. Modified Ladder Task

This shows the importance of a well designed action module inventory. Also, if a 6 inch step is used for the "move body +y" (003) action module, a solution can not be found. The ability to move the body to precise locations between the missing pegs is required. As the step size is decreased, plan flexibility is increased at the cost of an increased search space. However, through learning, the methodology would assemble larger y motions of the body as necessary.

Figure 9 shows the evolution of the plan (shown in Table 2) for the modified ladder task. The long flat section of the graph corresponds to the time required to evolve a plan to get past the missing pegs. Once the planning method increased the reach of the robot by rotating the body, it is able to quickly find a plan to move up the ladder. This can be seen by the large jumps in fitness score after the flat portion of the graph.

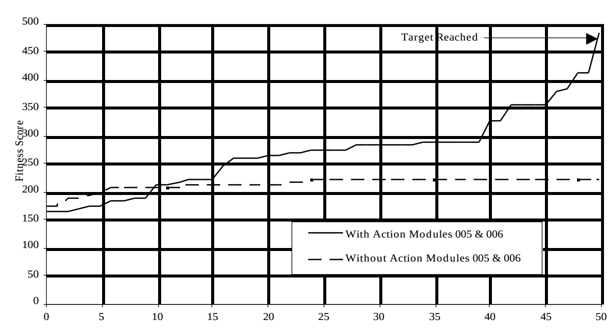

Figure 9. Fitness Score for the Evolution of the Modified Ladder Plan

Table 2. Action Plan for the Modified Ladder Task

Step #	Action	Step #	Action	Step #	Action
1	3	21	3	41	3
2	291	22	3	42	3
3	6	23	3	43	3
4	3	24	3	44	291
5	3	25	3	45	5
6	391	26	3	46	3
7	3	27	5	47	3
8	291	28	3	48	391
9	3	29	3	49	3
10	6	30	3	50	3
11	3	31	5	51	3
12	3	32	3	52	3
13	6	33	2	53	193
14	3	34	3	54	3
15	3	35	193	55	3
16	293	36	292	56	3
17	392	37	392	57	300
18	100	38	3	58	3
19	3	39	1	59	3
20	3	40	3	60	3

3.3 The "H" Task

A human was asked to derive a plan to move the LIBRA from point A to point B in the H task shown in Figure 10 while minimizing joint torques. The maximum torque required was approximately 1900 oz-in, occurring at the location of LIBRA in Figure 10a. The action module methodology was then used to produce a plan that maintained joint torques below 1/2 the saturation limit (3000 oz-in). Sub-goals (SG) were required to prevent the planner from getting caught in a local minimum as mentioned previously. The method was able to find a solution (shown in Figure 10b) with a maximum torque less than 1500 oz-in. Figure 11 shows the time history of the maximum joint torques for both the human and the GA-derived plans. This shows that the planner can explicitly and effectively consider parameters such as joint torques and power consumption that are not obvious to a human planner.

4. Summary and Conclusions

This paper described the application of an action module planning methodology to an experimental climbing robot named LIBRA. Details of the action planning methodology and the genetic algorithm search method used, as well as specific details for their application to the climbing robot, were presented. A physics based Plans simulation was used for the application of the planning methodology to ensure that no physical constraints of the system were violated. The methodology's ability to use the full physical capabilities of the robot to accomplish tasks that are extremely challenging for the robot was shown. Additional results were provided that showed the importance of the design of the action module inventory.

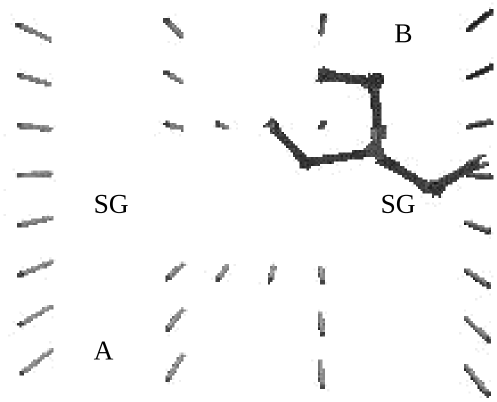

Figure 10a) Human-Derived Plan

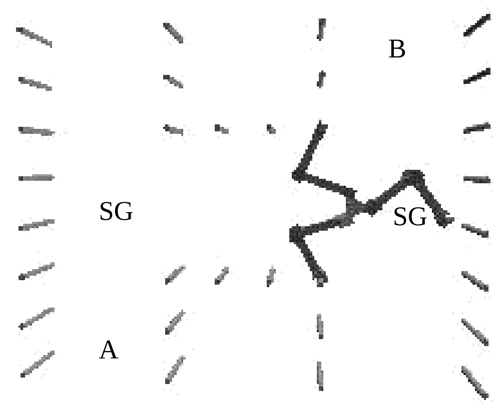

Figure 10b) GA-Derived Plan

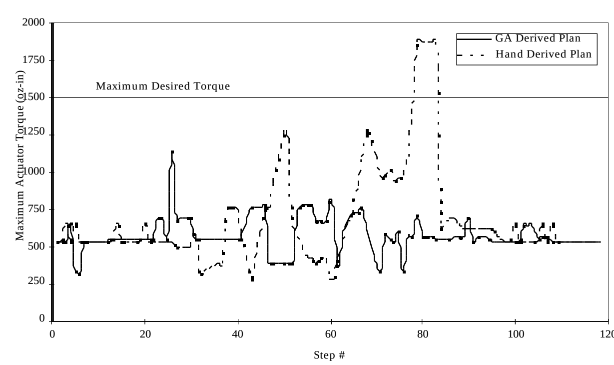

Figure 11. Maximum Actuator Torques for the Two Different H Task

5. Acknowledgments

This work is supported by the NASA Jet Propulsion Laboratory.

6. References

[1] Latombe, J.C., *Robot Motion Planning*, Kluwer Academic Publisher, 1991.

[2] Brooks, R., "A Robust Layered Control System for a mobile robot," *IEEE Journal of Robotics and Automation*, Vol. RA-2, No. 1, 1986, pp. 14-23.

[3] Neibauer, W., "Locomotion with Articulated Legs in Pipes or Ducts," *Robotics and Autonomous Systems*, Vol. 11, 1993, pp. 163-169.

[4] Khatib, O. "Real-time obstacle avoidance for manipulators and mobile robots," *International Journal of Robotics Research*, Vol. 5, No. 1, 1986, pp. 90-98.

[5] Dubowsky, S., Moore, C., and Sunada, C., "On the Design and Task Planning of Power Efficient Field Robotic Systems," *Proceedings of the ANS 6th Topical Meeting on Robotics and Remote Systems*, Monterey, CA, 1995, pp. 501-508.

[6] Madhani, A. and Dubowsky, S., "The Force Workspace: A Tool for the Design and Motion Planning of Multi-Limb Robotic Systems", *ASME Journal of Mechanical Design*, Vol. 119, No. 2, June 1997, pp. 218-225.

[7] Farritor, S. and Dubowsky, S., "A Self-Planning Methodology for Planetary Robotic Explorers," *Proceedings of the 8th International Conference on Advanced Robotics*, Monterey, CA, July 1997, pp. 499-504.

[8] Krotkov, E. and Hoffman, R., "Terrain Mapping for a Walking Planetary Rover," *IEEE Transactions on Robotics and Automation*, Vol. 10., No. 6, December 1994, pp. 728-739.

[9] Chen, C-H. and Kumar, V., "Motion Planning of Walking Robots in Environments with Uncertainty," *Proceedings of the IEEE International Conference on Robotics and Automation*, Minneapolis, MN, Vol. 4, 1996, pp. 3277-3282.

[10] Farritor S., Hacot H. and Dubowsky S., "Physics-Based Planning for Planetary Exploration," IEEE International Conference on Robotic and Automation, 1998.

[11] Schenker, P., et. al., "Lightweight Rovers for Mars Science Exploration and Sample Return," Intelligent Robots and Computer Vision XVI, SPIE Proc. 3208, Pittsburg, PA, October, 1997.

[12] Argaez, D.A., *An Analytical and Experimental Study of the Simultaneous Control of Motion and Force of a Climbing Robot*, M.S. Thesis, Department of Mechanical Engineering, Massachusetts Institute of Technology, Cambridge, USA, June 1993.

[13] Dubowsky, D., Sunada, C. and Mavroidis, C., "Coordinated Motion and Force Control of Mobile Multi-Limbed Robotic Systems," *Journal of Autonomous Robots*," vol. 6, no. 1, 1999.

[14] Bevly, D., Dubowsky, D., Mavroidis, C, "A Simplified Cartesian Computed Torque Controller and its Application to an Experimental Climbing Robot," Accepted for publication in the *Journal of Dynamic Systems, Measurement and Controls*, April 1999.

[15] Goldberg, D. Genetic Algorithms in Search, Optimization, and Machine Learning, Addison-Wesley, Reading, MA, 1989.