Design of a Low-Cost, Highly Mobile Urban Search and Rescue Robot

Bradley E. Bishop*
Frederick L. Crabbe
Bryan M. Hudock

United States Naval Academy
105 Maryland Ave (Stop 14a)
Annapolis, MD 21402
bishop@usna.edu

Keywords: Rescue Robotics, Mobile Robotics, Locomotion, Physical Simulation, Genetic Algorithms

Abstract— In this paper, we discuss the design of a novel robotic platform for Urban Search and Rescue (USAR). The system developed possesses unique mobility capabilities based on a new adjustable compliance mechanism and overall locomotive morphology. The main facets of this work involve the morphological concepts, initial design and construction of a prototype vehicle, and a physical simulation to be used for developing controllers for semi-autonomous (supervisory) operation.

I. INTRODUCTION

Recovering survivors from a collapsed building has proven to be one of the more daunting challenges that face rescue workers in today's world. Survivors trapped in a rubble pile generally have 48 hours before they will succumb to dehydration and the elements [1]. Unfortunately, the environment of urban search and rescue (USAR) does not lend itself to speedy reconnaissance or retrieval. The terrain is extremely unstable and the spaces for exploration are often irregular in nature and very confined. Though these challenges often make human rescue efforts deep within the rubble pile prohibitive, a robot designed for urban search and rescue would be very well suited to the problem.

Robots have already proven their worth in urban search and rescue, most notably in the aftermath of the September 11th, 2001 disaster. The combined efforts of Professor Robin Murphy, a computer scientist at the University of South Florida, and Lt. Col (ret) John Blitch, culminated in the creation of the Center for Robot Assisted Search and Rescue (CRASAR), which coordinates and assists robotic search and rescue efforts [2]. Though these and other undertakings have shown the potential of a robot in the urban search and rescue environment, there is still room for significant improvement (see the papers in [3] and associated references). This paper focuses on the need for a unique structure and method of mobility, which could result in a vehicle much improved over existing urban search and rescue robots in terms of range and ability to overcome obstacles.

The primary goal of this project was to develop a robot that was highly mobile, lightweight, and easy to control. While many robots used in search and rescue missions are modified commercial products (see the robots shown in [4], e.g.), the reported system was designed from the ground up with specific features to combat the irregular terrain found in an urban search and rescue environment. Specifically, the robot was designed to overcome three pre-selected obstacles that represent a sample of the types of impediments found in a collapsed building: non-cohesive irregular terrain (such as piles of loose rubble), an open canyon or deep void, and a ladder or open stair (see Figure 1). These obstacles are indicative of the more challenging aspects of standard urban search and rescue environments [5,6].

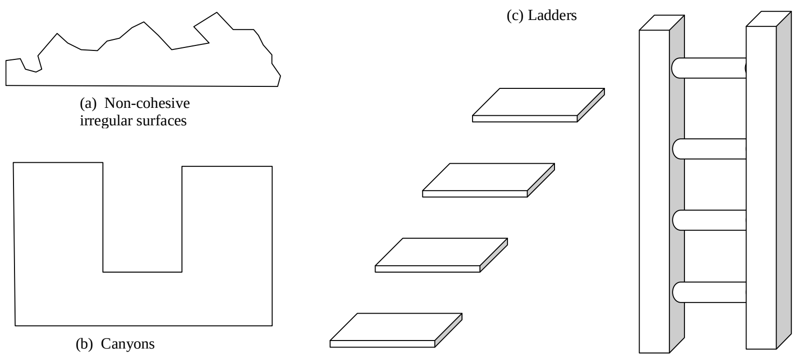


Figure 1: Selected obstacle types

The secondary goal of the project was to investigate the requirements for developing controllers for the system that would reduce the burden on a human operator by making complex coordination of many degrees of freedom transparent to the user, who uses simply labeled commands such as “move forward” and “turn left” to drive the robot. This reduction in burden on the operator is an important component for addressing the human interface problem in the urban search and rescue domain [4]. Unfortunately, such controllers are non-intuitive for a system with many degrees of freedom in a typical urban search and rescue environment. In order to build appropriate controllers, evolutionary methods were applied to a dynamic simulation of the physical prototype and its environment.

The remainder of the paper is organized as follows: Section II contains a description of the basic functionality and robot morphology selected, the simulation environment and controller development, and the reasoning behind the design decisions. Section III includes details of the physical prototype and a discussion of the associated design concepts. Results of testing for the prototype are discussed in Section IV. Conclusions and a discussion of important research directions for controller development are given in Section V.

II. ROBOT DESIGN

The underlying goal of this research effort was to isolate and identify basic locomotive methodologies that were effective, reliable, and easy to implement in the environment of robot-assisted urban search and rescue. A variety of robot morphologies have been fielded based on standard differential-drive and/or tank-style locomotion (see [7, 8] for an overview of these basic vehicle types), augmented tracks [9] and snake-like systems [1,10][1]. The basic characteristics that drive these selections involve the ease of use, ease of repair, and locomotion capabilities for the types of terrain that are anticipated in an urban disaster situation. Based on reported success from [9, 10], the addition of appendages and extra degrees of freedom has the benefit of substantially increasing locomotive capability over uneven terrain, but also increases the overall burden on the rescue worker and the complexity of the system from a repair and maintenance standpoint.

[1] Here we do not include distributed or multi-robot systems, but rather focus on single-unit approaches. The techniques used for cooperative approaches might be applied to almost any morphology. See [11-13] for examples of interesting multi-robot approaches to search and rescue.

The major considerations that went into the proposed design focused on the difficulties associated with the novel terrain types shown in Figure 1. Single-segment units face significant difficulties with all of the obstacles shown (depending on the relative size of the vehicle and the obstacle). The limiting consideration for such vehicles is that they must have dimensions that are compatible with the largest obstacle they are intended to overcome. If a single-segment vehicle is intended to cross a gap of one meter, it must be longer than one meter. This length then dictates the agility of the system. A meter long robot would have substantial problems navigating a right-angle turn in a 0.5 meter wide void unless it was very narrow, which presents stability issues. Thus, we focus on segmented vehicles of the nature of [10].

Segmented vehicles have their own set of difficulties. Primary among these is that such systems have a large number of degrees of freedom, which results in problems relating to power, control and complexity. Even so, the capabilities added by the additional degrees of freedom outweigh the problems, if the system is carefully designed.

One of the limiting factors associated with snake-like multi-segment robots is the difficulty of controlling even straight-line motion over uneven terrain. While snake-like robots can perform acrobatic feats that are well beyond any single-segment system, the control issue looms large in any design that is to be operated in an unpredictable environment [10, 14].

One simple method for improving performance of a snake-like robot is to combine a well-known locomotive capability with the multi-segment concept. By adding wheels or tracks to the segments of the robot, the system achieves the best benefits of both tank-like robots and snake-like robots, as is seen in [15-17] (although these robots have varying degrees of controlled articulation). Understanding that tracks are generally a better solution than wheels for uneven terrain, we consider the implications of adding independent tracks to the links in a serial robot.

The addition of tracks to a mobile robot system has benefits and drawbacks. On the one hand, the track applies force all along its surface, in essence transforming the side of a robot into one large wheel. Though tracks generally give robots an advantage in irregular terrain, the added complexity of a track system and the need for specialized parts often make them prohibitive. In the case of an urban search and rescue robot, the benefits of a track system have been seen to outweigh the specialization and complexity problems (as almost every referenced robot system possesses tracks).

It is the fundamental tenet of the design of the proposed system that the irregular and unpredictable nature of the environment in urban search and rescue calls for creative ways of dealing with terrain challenges, and that a robot with a variety of methods of movement has a greater chance of succeeding in such a difficult setting. Thus, the combined snake and track approach was deemed to be the most promising.

Unfortunately, a tracked snake-like robot still experiences three fundamental problems when working in typical urban search and rescue environments: counterproductive force, terrain compliance, and the operator interface. In the sequel, it is shown that the proposed system overcomes two of these problems using carefully designed robotic mechanisms, and that the third issue is of fundamental concern for advancing designs of this type.

II.A The Problem of Counterproductive Force

A common difficulty with existing robots for urban search and rescue is caused by the wraparound track that is typical of most of these devices. Wraparound tracks actually prohibit motion when encountering top-and-bottom obstacles, e.g., when tunneling through a narrow void or in loose, non-cohesive terrain (see Figure 2). When moving through large pieces of loose rubble, the counterproductive force produced by contact with the environment on the top surface of the vehicle results in decreased efficiency and increased difficulty in control. Obviously, vehicles that make uncontrolled contact with the environment on top (due to computer housings, sensors or chassis extensions) also experience reduced mobility due to drag and potential hang-ups.

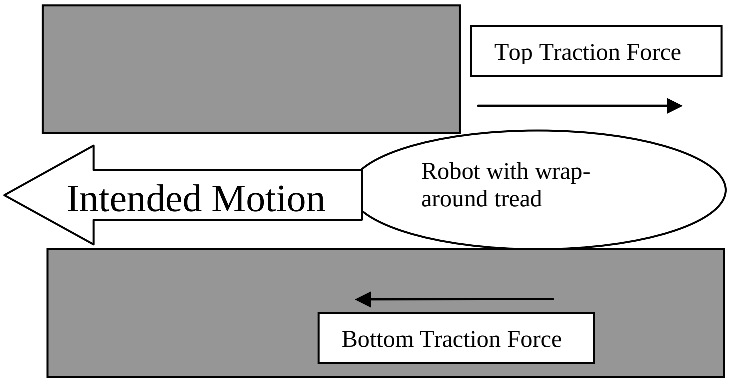


Figure 2: Effect of "wraparound" track

The mechanism chosen to overcome this difficulty is to place tracks on both top and bottom of the vehicle. These tracks can be independent or coupled in such a way as to always turn in opposite directions. Under this architecture, the problem of counterproductive force is negated, and the robot is able to push through tight voids and loose rubble with ease and efficiency. While the added tracks do increase complexity of the system, the improved performance in highly cluttered environments outweighs the drawbacks. Figure 3 shows the benefits of using tracks on both the top and bottom of the vehicle. A similar, independently developed design for a single-cart vehicle, intended for reconnaissance inside an ancient pyramid, can be seen in [18] (although this single-unit device also possesses a mechanism by which the vehicle expands vertically, as it was intended to traverse a small tunnel of known dimensions and not move through rubble).

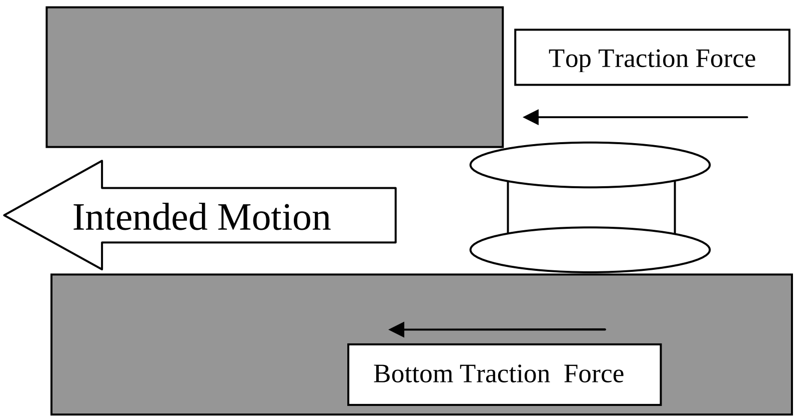


Figure 3: Effect of independent top and bottom tracks

Based on this discussion, each link of the robot was designed to have four independent tracks, situated at top left and top right, and bottom left and bottom right. As seen, mounting tracks on both the top and bottom of each link in the robot enhances mobility in confined areas by allowing the robot to use the roof of the environment along with the floor in order to propel itself forward. The independent left-right tracks provide differential steering on each cart, which aids in mobility by not relying on strictly serpentine motion (which may not be computationally simple nor very efficient over even smooth terrain). Placing additional tracks on each of the 'side' faces was considered, but the complexity precluded their inclusion in the initial prototype phase. These extra tracks could provide needed traction in case the robot tipped over, although it is anticipated that the device could right itself using only the actuation between the carts as seen in [19].

II.B The Problem of Terrain Compliance

One of the primary difficulties with snake-like robots is the control of contact forces in uneven terrain. In order for the robot to achieve robust and efficient locomotion, contact must be made between the links and the environment along as much of the length of the vehicle as possible (in most cases). The difficulty here is that of *compliance*. That is, how much of the contact with the irregular surface is achieved through passive means and how much through active control. Most traditional robot manipulators without force feedback are devices with no compliance. The relative position of every link is always controlled, but the robot has no ability to conform to the environment. In fact, the robot typically resists change induced by the environment through joint-level feedback control. At the other extreme, a chain has a high level of compliance. A chain conforms to an irregular environment, but it is impossible to control the position of each link in relation to the others.

While it is true that linked units in a snake-like robot have far more versatility and flexibility than a monolithic structure, and that powered joints also provide the ability to perform shape forming with the links, it is also true that when the joints of a snake robot are powered, the robot becomes inflexible and unable to adapt its overall shape to the environment without purposive action from the controller. A robot that automatically conforms to the terrain may be much more effective, as it maximizes the amount of contact between the robot and the surface without requiring complex (and challenging) control. On the other hand, a passively compliant robot could not achieve shape forming, as the shape would be dictated by the environment. Further, high levels of compliance could easily result in the robot falling into a bad configuration due to environmental forces. In order to accomplish both shape forming and compliance, novel pitch-yaw robot joints were designed.

The *adjustable compliance* joint was based on a simple principle. Adjustable compliance allows the robot to adjust the level of compliance based on the expected situation and environment. The goal of adjustable compliance during normal operation is to keep as many cars in contact with the floor surface as possible, especially over irregular terrain, without the need to modify the control or use force feedback. Achieving this goal involved careful design of the portion of the mechanism that controls pitch. Instead of a rigid connection from the motor controlling pitch to the next car, the two entities are connected by a rod, which has a rigid connection to the distal link and a passive, one dimensional pin joint on the proximal link. Springs are connected on the top and bottom of a disk attached to the motor shaft on the proximal link, with the other ends of each spring connected to the top and bottom of the distal link (see Figure 4). The motor is a position-controlled servomotor, and the springs are both pre-tensioned in the nominal zero position (as will be discussed in Section III). Compliance was only implemented in pitch, not yaw.

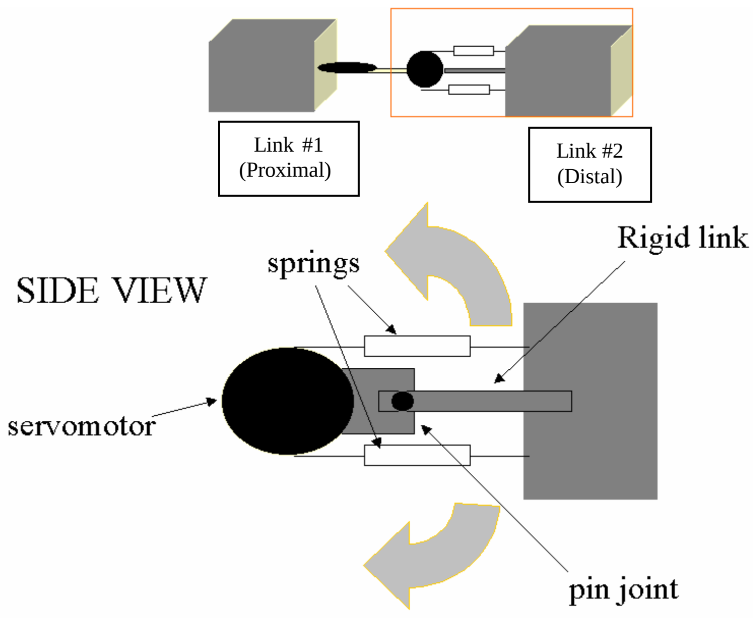
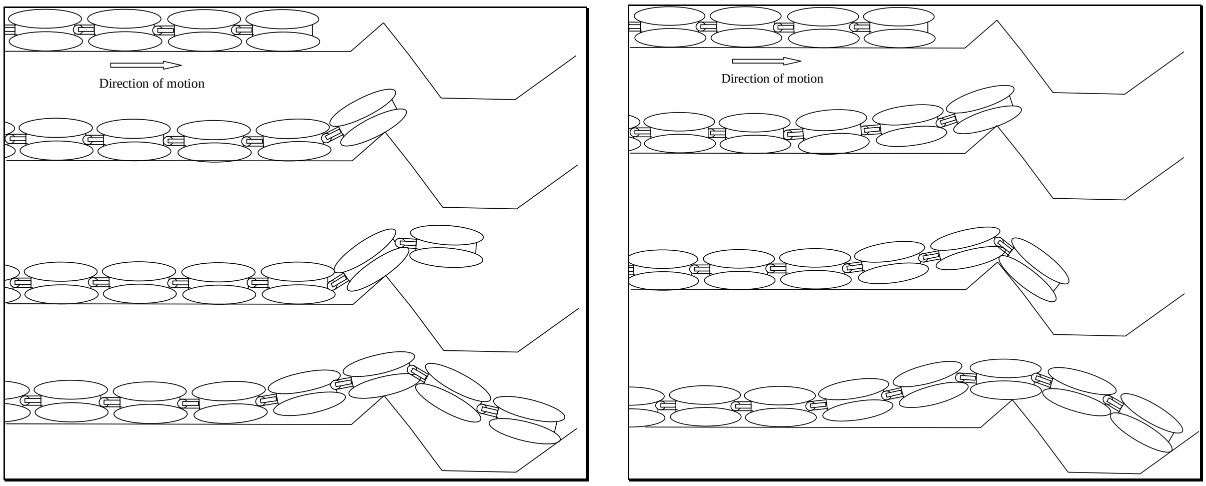


Figure 4: Adjustable compliance mechanism

Given the mechanism shown in Figure 4, it is seen that rotating the motor shaft changes the tension on the springs and the torque between the two links. Let us consider the end of a many-link robot. We let the distal link in Figure 4 be the last (end) link in the robot. If the motor shaft in Figure 4 is rotated counter-clockwise, the upper spring extends while the lower spring compresses. This action results in a net tension force on the proximal link that tends to pull the link upward. If the motor is moved far enough in the counterclockwise position, the upper spring will be fully extended and the distal link will lift off the ground. Thus, the system achieves control of the link. More important is the effect on the compliance of the system.

The springs in Figure 4 provide compliance, based on their tensioning. Small deviations in surface height on the ground will result in the distal link moving upward or downward, depending on the nature of the irregularity. The amount of deflection will depend on the tension on the springs attached to the distal link as well as the tension on the other springs in the body. Two distinct operating examples are shown in Figure 5, where the distal links are tensioned up or down a small but meaningful amount and no additional pitch control is implemented.



(a) (b)

Figure 5: Example motion with joints tensioned 'up' (a) and 'down' (b). Note that the configuration 'prefers' upward relative pitch in (a) and downward relative pitch in (b).

This simple example demonstrates the effective modification of compliance in a desired direction. While the overall configuration difference is not extreme in the cases shown, the ability to control the pitch while simultaneously allowing compliance to a desired degree is a capability that significantly enhances the overall performance of the system.

An important example of the benefit of adjustable compliance involves traction forces. By tensioning a link downward against a surface, the tracks on that link can provide more locomotive force due to the increased normal force. In a noncompliant system, the link could easily lose contact along the surface by touching the surface only at the tip, unless some form of force sensing was incorporated. The use of adjustable compliance allows increases in normal force but provides some leeway in control by maintaining surface conformability over a range of tensions. While a similar result might be attained by a well-designed gain-scheduled position (or pose) controller, the proposed mechanism is more simple, more robust, and easier to implement (as will be seen in Section III) than such a technique. This mechanism represents a major step in increasing mobility in search and rescue robots.

II.C The Problem of User Interface

As mentioned, the robot is designed to be teleoperated. Therefore, it does not require a fully autonomous controller. On the other hand, given the number and variety of different degrees of freedom that must be coordinated, it is not possible for a human operator to control each motor individually. Even performing a 90^o turn involves operating each track at a different velocity while coordinating with the joints between the links. As such, the best manner for operating the robot is semi-autonomously. The user interface in robotic urban search and rescue is a well-known problem [4], and development of simple user controls for a complex system such as that discussed in this paper forms a fundamental basis for effective human-robot interaction.

We envision a library of complex actions, any one of which the operator can initiate asynchronously. For example, when faced with crossing a small canyon, the operator can order the robot to rear up, move forward to the edge, and unfurl across to the other side as a single operation or a set of simple instructions.

Each action in the proposed library, aside from being task compatible, should be:

- Blind. In order to simplify the development of the controllers, the individual actions should operate without any environmental sensor feedback. The task of developing the controllers is difficult, and limiting them to operating without environmental sensing should make the task more manageable. Even so, it is recognized that internal sensor feedback using, e.g., encoders, will be necessary.
- Atomic. The action should be easily trigger-able by the operator with a single command. It should require minimal further input from the operator (though it should be able to accept some additional input). The operator should be able to execute a "cross canyon" without specifying any of the motor commands.
- Robust. Each action should be successfully completed without the use of environmental sensors in a variety of physical settings. A turn or a climb should work nearly as well with the presence of surface irregularities as it does on open terrain. The adjustable compliance

built into the joints can provide a degree of the necessary robustness when traveling across irregular surfaces, but the actions themselves should be designed to operate in the widest range of situations possible.

- Serially blendable. Sequences of actions given to the robot should blend together seamlessly. For example, if the robot performs a left turn, at some point in the turn, the front cars have completed the turn, cars in the middle are making the turn, and the rear cars have yet to begin the turn. At this stage, if the operator issues a “right turn” command, the front cars should be able to begin the right turn while the remaining cars complete the left turn and then the right turn.

Unfortunately, the many degrees of freedom in the robot that prohibit direct operator control also impede the kinematic development of the action library. The vastness of the kinematic space combined with the variety of types of solutions (e.g. using the tracks vs. wriggling like a snake to move forward) make finding inverse kinematic solutions extremely challenging if not impossible. Thus a mechanism such as a Genetic Algorithm that automatically searches the kinematic space for solutions is preferable, as seen in [14].

The use of Genetic Algorithms (GAs) to develop a library of actions is made up of two components: the parameters of the environment and the algorithm (including the representation of the genome, the fitness function, and the actual search techniques). We opted for a simulated environment rather than evolving directly on the hardware primarily because it is recognized that evolutionary controller development takes many, many iterations of many individual controllers. Performing the requisite number of experiments on the physical prototype for even a simple set of actions would prove to be time-prohibitive. As such, we chose to attack the problem using a high-fidelity simulation to develop initial controllers. Additionally, we wished to be able to develop the control programs simultaneously with the development of the physical model, rather than wait until the robot was completed. Finally, we were unwilling to risk damage to the prototype that might occur due to a violent or otherwise incorrect control program during evolution.

II.C.1 Environment

Because the desired controllers are targeted at the kinematic level, it is important for the simulated robot and environment to be as accurate as possible. This precluded the use of any preexisting robot simulators and instead required the construction of a specific simulator for the robot. Constructing such a simulation entailed selecting a capable physics simulation package that included features to be included in the prototype design. Open Dynamics Engine (ODE) was selected as the tool for modeling and simulation. ODE is "a free, industrial quality library for simulating articulated rigid body dynamics - for example ground vehicles, legged creatures, and moving objects in VR environments. It is fast, flexible, robust and platform independent, with advanced joints, contact with friction, and built-in collision detection" [21].

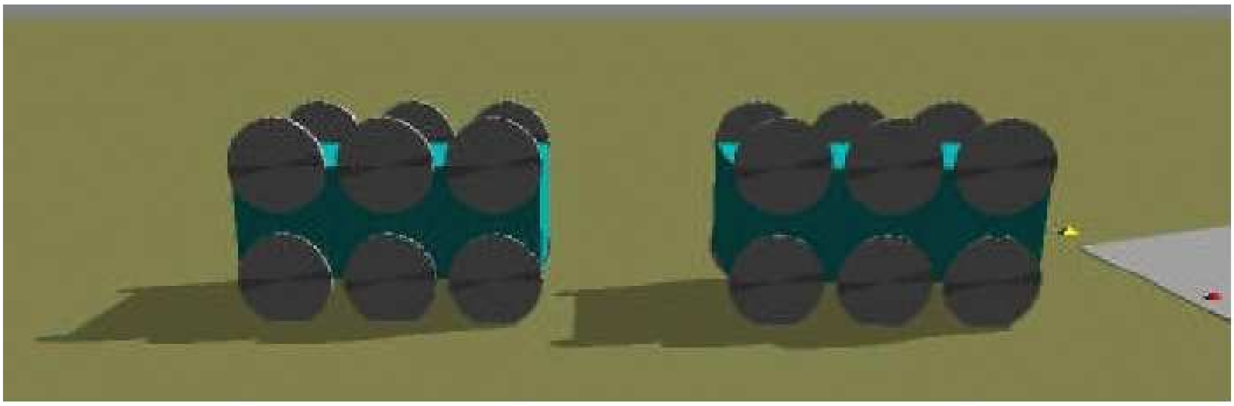


Figure 6: The simulation model of the robot. Only two of the six cars are shown. The joints are not displayed in the model.

The ODE simulation was written in C++ on A Linux Workstation. The final simulated robot (shown in Figure 6) was similar to the actual physical prototype (discussed in Section III below). The primary difficulty encountered was the simulation of the tracks. Tank track modeling is a notoriously difficult problem in physical simulation (as far as we know, no one has successfully created a full physical simulation of tank tracks). Instead the tracks were approximated using multiple wheels, as seen in the figure. All the wheels were powered to model the tracks as closely as possible.

Another difficulty was the simulation of the adjustable compliance in the joints between the cars. ODE allows setting of global parameters that enable joints to behave with some elasticity, but setting these parameters causes all joints and collisions to exhibit that characteristic. Instead, the joints are currently modeled without compliance.

II.C.2 Genome and Search Techniques

Representation of the genome was simplified due to the lack of sensing requirements for the individual actions. This meant that each action should be a sequence of motor movements regardless of the external environment. The chromosome was a 60 element sequence of "motor vectors," with each vector representing a set of motor accelerations across one second. Within each motor vector were six four-dimensional "car vectors" and five two-dimensional "joint vectors," representing the motors of the individual cars and joints, as seen in Figure 7. During a test of a particular genome, the motor vectors were sampled, one per second, and the accelerations were applied to each individual motor. The hierarchical structure of the representation, with links and joints as units within the motor vectors, was selected to encourage the emergence repetition of motion across the multiple links. It was our belief that most correct controllers will repeat motor movements at different links or joints at different times. We hoped to encourage the emergence of such properties.

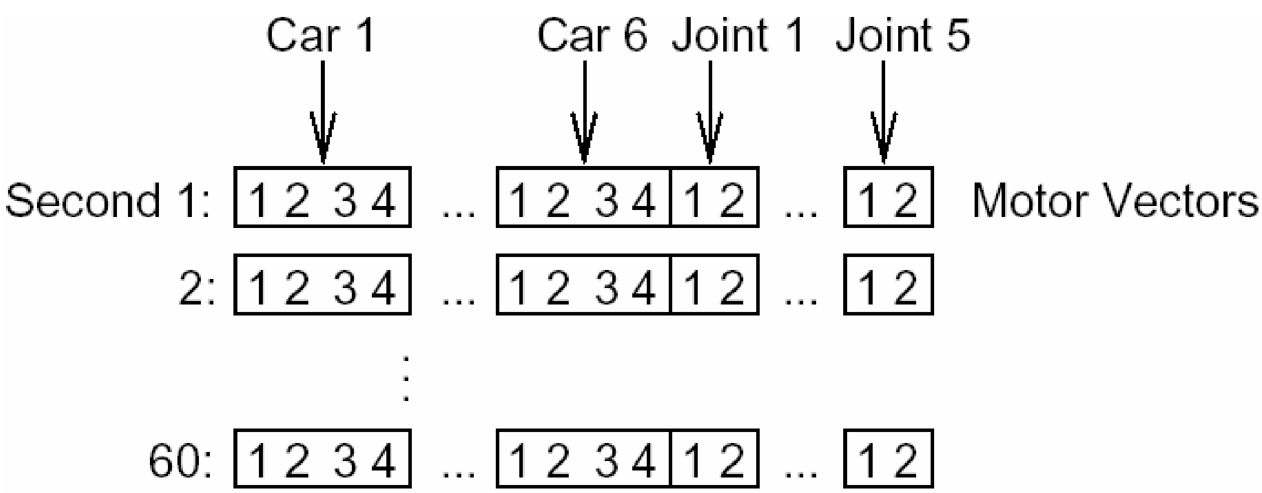


Figure 7: The genome is made up of 120 motor vectors. Each motor vector is made up of six link vectors and five joint vectors. Each link vector is made up of four individual motor acceleration values, and each joint vector is made up of two accelerations.

The GA used both mutation and two-parent crossover during the evolutionary process [22]. Mutation could occur at any single acceleration value, changing it to a new value. Crossover points were randomly selected at locations between the motor vectors. Finally, because we wanted to encourage repetition of motion across cars, individual link or joint vectors could be copied over joint or link vectors in other motor vectors. For example, link vector 4 in motor vector 7 could be duplicated in link vector 6 in motor vector 30. The vector duplication operation was performed within an offspring after crossover and before mutation had been applied.

The fitness functions used were straightforward. For turning actions, fitness was measured as the deviation from the desired heading at the end of the time period. The heading of the robot was the average heading of all the cars. For the remaining actions, fitness the distance traveled forward, measured by the center of mass of the front car.

II.C.3 Experiments

The difficulties found in the construction of the simulation have prevented, to this point, full execution of the GA. The lack of true tracks plus the lack of adjustable compliance resulted in many situations where the simulated robot would become stuck as it traversed uneven terrain. It was common for a sharp edge of an obstacle to wedge in-between the wheels, preventing any rolling movement without lifting that car off the obstacle. Similarly, without adjustable compliance, rolling over just a slightly rough surface required complete coordination with the joint motors. Both these difficulties combine to make suitable solutions much more difficult to find in the search space. They also make what solutions are found less applicable to the physical robot, as it does not have these properties.

As a proof of concept, we performed experiments using the simulated robot on smooth surfaces for a forward motion task (using a combination of serpentine motion and track actuation). While motion over smooth terrain with a tracked snake is well understood, this sample case served to inform the use of GA methods in development of controllers for this complex system.

We used a mutation rate of 0.1. Link or joint vectors were duplicated within an individual at the same 0.1 rate. As a baseline, we also evolved a population with just mutation and crossover but without duplications of link or joint vectors. The population size was 100 individuals, and they were run for 15 generations. On an Intel 2.0GHz Pentium 4 Processor, this took roughly two days per trial.

While the number of generations was not enough to gather clear results, some properties can be seen. For the forward motion task, the GA without duplication produced some forward movement (18 cm), but the best performing individual in the trial with duplication moved 82 cm over one sample period. Similar trials executed for the turning task did not result in such clear improvements over the 15 generations. Further trials were halted until the simulation itself could be refined. Those improvements will be discussed in Section VI, but it was seen from the simulations performed that the basic concept of GA-based teleoperation controller development was valid. Further, these experiments provided valuable insight into the need for advances in dynamics simulation capabilities.

III. PHYSICAL PROTOTYPE

Having developed an operational concept and general robot morphology, a physical prototype was designed. Many fundamental problems were addressed, but the primary focus of the prototype development was implementation of the two major concepts of the design (multiple tracks and adjustable compliance) in a robust, lightweight package that was simple to maintain and operate. The basic morphological design of the robot is of no use if it is not feasible to implement it in a compact and powerful package. In this section, we describe how the basic morphology was developed into a physical prototype, piece-by-piece. The vehicle was powered through a tether that also carried the control signals from the operator. We do not consider sensor packages in this work.

The final prototype robot (Fig. 6) consisted of three links and two joints with 2 DOF each (pitch and yaw). This was the minimum size system needed to accomplish basic locomotive tests. Each car was constructed out of ¼” thick plywood and was reinforced by ¼” steel threaded rod. The total weight of the robot was approximately thirteen pounds (with no onboard power supply). The car links had a maximum pitch angle of approximately +/- 60^{o} (depending on the pre-tensioning in the adjustable compliance springs) and a yaw (heading) range of motion +/-45^{o}. Each link (including the joint) is 10”x4”x4”.

The major components of the physical design included the tracks, motors, chassis, joints and control hardware, each of which will be addressed separately.

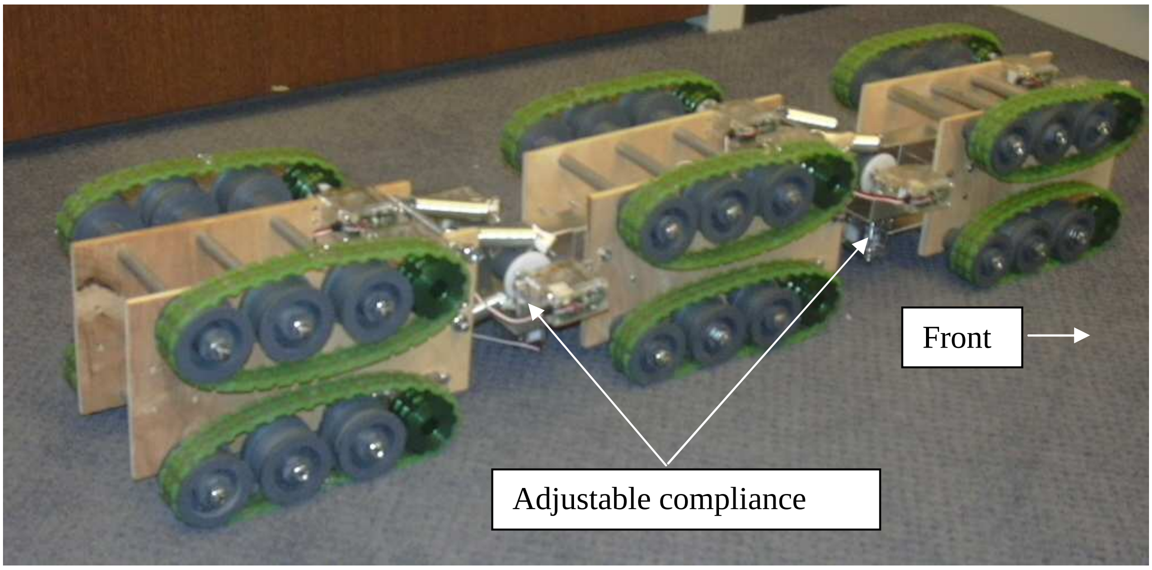


Figure 8: Robot prototype

III.A Track Design

The tracks selected needed to be strong, durable, and conducive to modification according to the dimensional requirements of the robot. Further, the tracks needed to be made of material that could be easily repaired and/or replaced by a field team in order to get the system back up and running after damage was incurred. Finding tracks that met these requirements proved a much greater challenge than expected. Timing belts manufactured by W.M. Berg proved to be an acceptable solution. A basic overview of various approaches to designing track systems for mobile robots can be found in [8].

The timing belts selected have two very nice properties. First, they are "no walk," or grooved in more than one direction, such that they cannot slip off the side of the pulley (which is attached to the drive motor). The second characteristic of the belts that made them attractive was that they consisted of steel cords surrounded by polyurethane, making them very strong and durable without too much weight.

The first downside to the timing belt emerged when it came time to splice pieces together into loops to be used as tracks. W.M. Berg offered a splicing kit for such purposes. Unfortunately, the manufacturer's splicing system left a large gap in the track that would have caused unacceptable snags on the contact surface. The splicing kit itself was also very expensive and could prove to be difficult to use in a disaster environment. Consequently, the decision was made to explore other splicing options. Boring small holes at each end of a piece of timing belt and using steel thread to sew the two ends together proved to be a solution that was both effective and time-efficient.

Surprisingly, the strength of the timing belt proved to be another drawback when it was used as track. When looped around small objects such as the wheels of the robot, the timing belt's steel cord center acted as a spring that gave the belt a natural tendency to straighten out. This tendency of the timing belt to push outwards further enhanced the normal force contact with an object. However, this characteristic also lessened the contact area between pulley and belt, making it more likely a track would be thrown, rendering one side of the car useless (although this never occurred during testing due to the 'no-walk' timing belts). A less rigid material under tension would sag and ensure greater consistency of motion.

The track system included passive idler wheels that were mounted alongside the wheels actually driving the tracks. By maintaining contact points and normal forces along the entire length of the bottom of each track, the idler wheels were the parts that gave the track its high capability in terms of mobility. Thus the idler wheels needed to be lightweight, but also of a material that could be grooved to provide a continuous path for the track to follow. PVC plastic was selected for its lightweight yet durable nature, as well as the ease with which it could be lathed and shaped.

III.B Motor Selection

The robot design required two different types of motors; one type of motor to power the tracks and one type to power the joints between cars. Both types needed to be lightweight DC motors capable of producing high torque. The one significant difference between the two motor types was that the joint motors required position control, while the track motors needed only speed control. A heavy-duty R/C servomotor was able to accomplish both tasks.

R/C servomotors are small DC motors with internal gearing, shaft position sensing and built-in control based on pulse-coded inputs. R/C servomotors were perfectly suited to powering the links because the built-in shaft encoders provided accurate positional control. Unfortunately, the one drawback to using a servomotor to power the tracks was that servomotors have mechanical stops that only allow a range of motion of approximately 180°. While this is more than sufficient for the joint motors, the drive motors require continuous rotation. Additionally, the feedback loop of the servomotors is controlled by a single-turn potentiometer connected to the output shaft. The stops had to be removed and the potentiometer cut out of the circuit for continuous rotation. In order to achieve speed regulation, a resistor bridge was put in place of the potentiometer. The internal PD controller of the R/C servomotor always reads a position corresponding to the setting of the resistor bridge, and so position commands result in speeds that vary according to the distance between the nominal sensor reading and the desired 'position' [6].

The R/C servomotor selected was a CS-80MG Pro, made by Cirrus. Weighing only 2 ounces, it is able to deliver 129.8 oz-in of torque at 6 volts. Constructed with metal gears and ball bearings, the servomotor is much more durable than typical servomotors that use plastic gearing. R/C servos are reliable and inexpensive, and can be acquired in large quantities and easily coupled to the vehicle. This, along with their high power/weight ratio, internal gear train and simple interface, make them excellent choices for a low cost, high capability search and rescue robot, for both joints and tracks. That the position and speed regulation controller is built into these motors makes them even more attractive, and also highlights an extra benefit of the adjustable compliance joint, which is separate and distinct from the position control hardware.

III.C Chassis Design

In order to optimize the robot's effectiveness, the chassis of each car had to fulfill a few important requirements. First, the chassis needed to be lightweight. This was important not only to increase the ease of transportation, but also to assist in the implementation of a two-degree of freedom link. Assuming a distance of about four inches from servo shaft to vehicle center of mass, a single servomotor as detailed in Section III.B above is capable of lifting a link that weighs approximately 30 ounces, so each link was designed to be no more than twice that value in total weight (excluding the joint).

The chassis was required to support three items: the servomotors driving the tracks, the idler wheels, and the joint structure (plus any sensor packages that might be included). One of the benefits of using servomotors to drive the wheels of the vehicle was that the servos could be mounted in a simple manner with little extra bracing, as the servomotor casing had fabricated holes that allowed it to be bolted in place. This is beneficial from the standpoint of maintenance and repair.

The chassis design eventually selected consisted of two flat sidepieces identical in appearance. Threaded steel rod was run through holes in the two sides. The rods supported the idler wheels while allowing them to rotate freely. Aluminum spacers were placed on the rods to reinforce the sidepieces while maintaining the car's exact dimensions. Aluminum and wood were used to construct the various car prototypes, as they were both adequately strong while costing little in terms of weight.

An important consideration in the design of a fielded version of the system is the design of the end links. The prototype, shown in Figure 8 above, was designed to demonstrate the capabilities of the morphology. In implementation, the end links (both 'front' and 'rear') should be tapered, as shown in Figure 9 below, to allow the vehicle to push its way into loose rubble and utilize its designed locomotive capabilities.

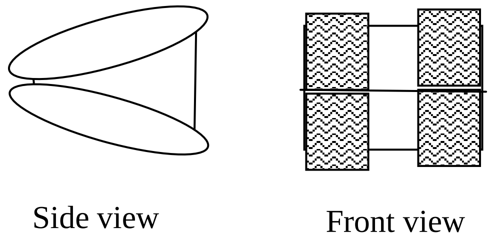


Figure 9: End-link design

The chassis designed is lightweight, durable and functional. The open structure allows easy access to any parts that may need to be replaced. Covers can be added to protect the components from potential damage due to environmental factors common in disaster relief situations.

III.D Joint Design

The joint forms the most important subsystem of the design. The yaw (or heading) component of the joint formed the first stage between the proximal and distal links (attached to the proximal). A single, unmodified R/C servomotor connected directly to the second stage of the joint and the proximal link. While this setup does not provide a significant amount of torque, the yaw motor is assisted by the differentially-driven links themselves in controlling heading.

The second stage of the joint, for the adjustable compliance, proved to be more challenging to implement. PVC spindles were machined to couple to the tensioning springs, which are fixed to the spindle in one of a variety of positions (to allow for a wide range of pre-tensioning states, depending on the operations required). Two servomotors are used for the pitch component, each with its own set of tensioning springs to maintain balance. The schematic layout can be seen in Figure 10, while the finished joint is shown in Figure 11. The springs used were standard door-closing springs from a hardware supply store.

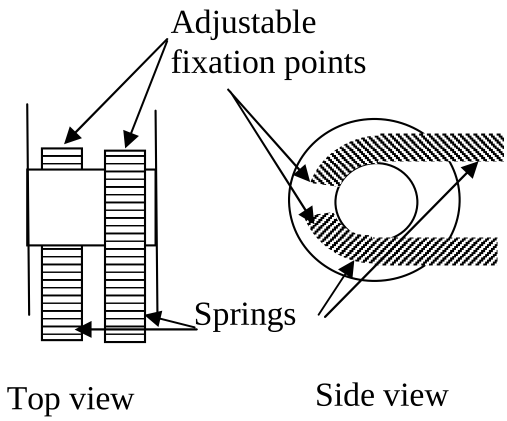


Figure 10: Spindle design for adjustable compliance

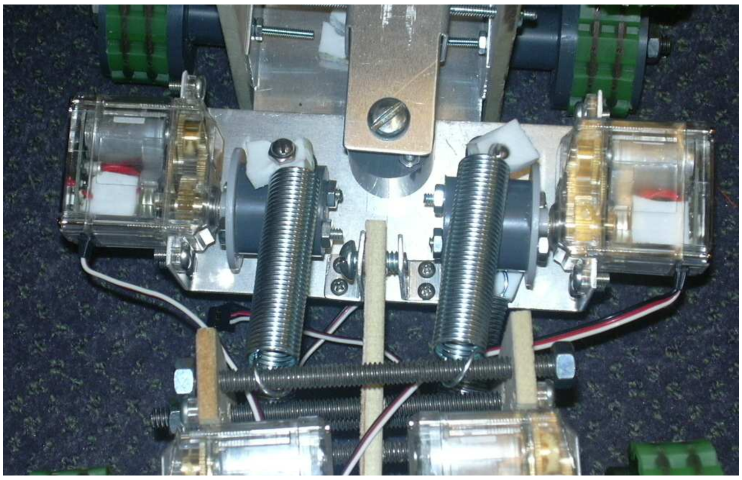


Figure 11: Passive compliance joint (top down)

III.E Electronics

To properly evaluate the prototype robot, a controller and simple user interface was designed. A Rabbit microprocessor forms the core of the control system, and is interfaced to a set of simple custom user controls (*forward/backward*, *tension up/tension down*, *curve left/curve right*). The burden of controlling eighteen motors is handled by a Pontech SV203 board on each link, interfaced to the Rabbit through standard RS232 communications. The SV203 is capable of controlling up to eight standard R/C servos simultaneously.

IV EVALUATION AND TESTING

The physical prototype was tested on small samples of the terrain types shown in Figure 1. Overall, the physical prototype performed as expected. The mechanisms of adjustable compliance and heading control worked as designed. The vehicle performed well on uneven terrain, even with little user effort, due to the adjustable compliance. It was seen that the vehicle needed to be covered with some form of smooth outer coating to prevent snags on the environment, which is to be implemented in version two of the vehicle.

The top-and-bottom tracks performed as expected, but the lack of a front-end unit of the nature of that shown in Figure 9 limited the unit's ability to enter a loose rubble pile. Even with the specialized end-link, this type of tunneling relies on the strength of the vehicle and the weight and composition of the rubble. Nevertheless, this result is very promising.

Because full controllers have not yet been fully developed, the testing of the canyon and ladder was conceptual in nature. The robot's design is seen to be effective against a canyon obstacle. Using algorithms such as the one developed by Nilsson [20], the robot could be manipulated to achieve an upright position. From this position, it could then lunge across the canyon and use the terrain on the other side to pull itself over the gap. The developed system provides even greater flexibility than that of the system discussed in [20] due to the additional capability of being able to roll after having lifted many segments off the ground.

The morphology is also seen to be well suited to surmounting a ladder obstacle, in which the robot's only method of traversing would be to wind its way through the rungs (especially for rounded rungs or steep ladders). The robot's ability to use tracks on both its top and bottom should allow it to maintain forward motion by using the rungs above and below it for contact. Pitching the car upwards with the powered links should provide the force necessary to maintain contact with a rung located above a car, and the adjustable compliance mechanism should provide adequate flexibility to allow climbing without a great deal of sophisticated control, even for varying rung spacing and thickness. The robot's unique structural features should give it a large advantage when trying to defeat this difficult obstacle.

V. CONCLUSIONS AND FUTURE WORK

In this paper, we have described a working prototype urban search and rescue robot employing structural features designed to enhance the robot's movement capabilities. The adjustable compliance joint and cooperative top-and-bottom tracks are unique features that have never been applied to urban search and rescue robot designs. The robot's ability to control the relative angles between the cars, while at the same time allowing them a degree of compliance, was critical in creating a design that had a significant degree of flexibility. Dual top and bottom tracks significantly enhance mobility in confined spaces and loose rubble. A simple prototype was developed to demonstrate the ability to implement these mechanisms in a real, low-cost and readily repairable unit. A physical simulation was developed to assist with the design of user interfaces for simple control.

Future work on the robot morphology involves generation of the second generation of the vehicle. In the next prototype, the spring tensioners will be re-designed for more efficient control, the overall system will be tightened to reduce weight while still providing ease of maintenance, and new front/end modules will be built according to the design shown in Figure 9. The next prototype will include six segments plus the end links, and will be tested using the methods from [20] and those discussed in Section IV above for the canyon and ladder obstacles.

There are three main improvements in development for the simulation and the genetic algorithm: improved model of the tracks, a model of the adjustable compliance on the joints, and the serial blendability of the actions. For the tank tracks, we are currently expanding the capabilities of ODE to include a specialized object that can approximate track behavior. This process involves creating a new object, the elliptical cylinder, in the set of geometries. This will enable the simulated robot to represent an approximation of the physical space that would be taken up by a complex track system. Once the elliptical cylinder is completed, a specialized motor will be added to apply forces at the contact points of the cylinder as if it were a track, without rotating the cylinder itself. Together, these new features should more adequately model the tracks of the physical robot, as seen in Figure 12. We are further modifying ODE to introduce springing parameters for joints that can be applied to specific degrees of freedom. This will enable more a closer model of the adjustable compliance of the robots joints.

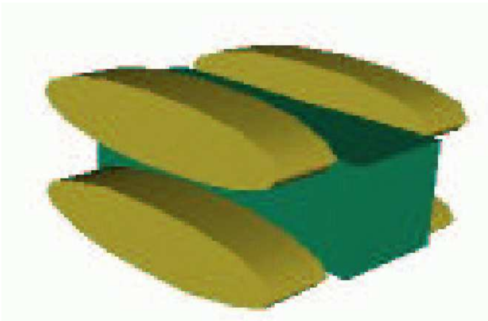


Figure 12: Simulated tracks on a single link.

As discussed in section III, one of the challenging aspects of controller development is blendability, the ability of the actions to blend into one another as one completes and other begins. This is difficult because the evolution of the individual behaviors is done in isolation, with no guarantee that the beginning of one action would not interfere with the completion of the previous. One possible solution for blending is to gradually shift cars from one behavior to the next, from the front car to the rear. This would most likely work for simple turns, but may fail in cases where the action of the front cars are required to complete the action for the rear cars, such as pulling them through the maneuver. In these situations, the action is unblendable with a successive action. Instead, we intend to try to evolve blendable actions. We will do this by first evolving the actions as described above. Once base controllers are obtained, the fitness function will be modified so that the controllers are evaluated when blended with other controllers. This should result controllers that generate blendable actions, but will continue to be an area of active research.

VI. REFERENCES

[1] Choset, Howard, "Search and Rescue," http://www.ri.cmu.edu/projects/project_407.html; INTERNET. June 16, 2004.

[2] "Center for Robot-Assisted Search and Rescue," http://crasar.csee.usf.edu/MainFiles/index.asp : INTERNET. June 16, 2004.

[3] Tadokoro, S.; Erkmen, A.; Erkmen, eds., *IEEE Robot. Automat. Mag, Special Issue on Rescue Robotics*, vol. 9, no. 3, 2002.

[4] Murphy, Robin. "Human–Robot Interaction in Rescue Robotics," *IEEE Transactions on Systems, Man and Cybernetics – Part C: Applications and Reviews*, Vol. 34, No. 2, May 2004, pp. 138-153.

[5] Osuka, K.; Murphy, R.; Schultz, A.C., "USAR competitions for physically situated robots," *IEEE Robot. Automat. Mag., Special Issue on Rescue Robotics*, vol. 9, no. 3, 2002. pp. 26 -33.

[6] Takahashi, T.; Tadokoro, S., "Working with robots in disasters," *IEEE Robot. Automat. Mag., Special Issue on Rescue Robotics*, vol. 9, no. 3, 2002. pp. 34-39.

[7] Bogatchev, et al. *Wheel Propulsive Devices for Mobile Robots: Design and Characteristics*. Russian Mobile Vehicle Engineering Institute

[8] Sandin, P. E., *Robot Mechanisms and Mechanical Devices*, McGraw-Hill Corp: 2003.

[9] Packbot Homepage, http://www.packbot.com/: INTERNET. June 16, 2004.

[10] Erkmen, I.; Erkmen, A.M.; Matsuno, F.; Chatterjee, R.; Kamegawa, T., "Snake robots to the rescue," *IEEE Robot. Automat. Mag., Special Issue on Rescue Robotics*, vol. 9, no. 3, 2002, pp. 17-25.

[11] R. Murphy, "Marsupial and shape-shifting robots for urban search and rescue," *IEEE Trans. Intell. Syst.*, vol. 15, no. 3, May/June, 2000, pp. 14–19.

[12] R. Dollarhide and A. Agah, "Simulation and control of distributed robot search teams," *Comput. Electr. Eng.*, vol. 29, no. 5, pp. 625–642, 2003.

[13] H. Jones and P. Hinds, "Extreme work groups: Using swat teams as a model for coordinating distributed robots," in *ACM 2002 Conf. Computer Supported Cooperative Work*, 2002.

[14] Kamegawa, T., Matsuno, F. and Chatterjee, R., "Proposition of Twisting Mode of Locmotion and GA based Motion Planning for Transition of Locmotion Modes of a 3-Dimensional Snake-like robot," *Proceedings of the 2002 IEEE International Conference on Robotics and Automation*, 2002, pp. 1507 – 1512.

[15] *Klaassen, B.; Paap, K.L,* "GMD-SNAKE2: a snake-like robot driven by wheels and a method for motion control," *Proceedings of the 1999 IEEE International Conference on Robotics and Automation*,10-15 May 1999, pp. 3014 - 3019 vol.4.

[16] S. Hirose and E. Fukushima, "Development of mobile robots for rescue operations," *Adv. Robotics*, vol. 16, no. 6, pp. 509–512, 2002.

[17] Kamegawa, T.; Yarnasaki, T.; Igarashi, H.; Matsuno, F., "Development of the snake-like rescue robot "kohga"," *Proceedings of the 2004 IEEE International Conference on Robotics and Automation*, April 26-May 1, 2004, pp. 5081 – 5086

[18] "IRobot Pyramid Rover," http://www.irobot.com/industrial/prover.asp: INTERNET. June 16, 2004.

[19] "Connected Crawler Vehicle for Inspection "Souryu I, II"," http://www-robot.mes.titech.ac.jp/robot/snake/soryu/soryu_e.html: INTERNET, June 16, 2004.

[20] Nilsson, Martin. *Snake Robot, Free Climbing.* IEEE Control Systems Magazine. Vol. 18, No 1. February 1998. p. 21-26.

[21] Smith, Russell. "Open Dynamics Engine," http://www.q12.org/ode/ode.html : INTERNET. April 14, 2003.

[22] S. Nolfi, D. Floreano. Evolutionary Robotics: The Biology, Intelligence, and Technology of Self-Organizing Machines (Intelligent Robotics and Autonomous Agents), MIT Press, Cambridge, MA 2000.