Behavior Hierarchy for Autonomous Mobile Robots: Fuzzy-behavior modulation and evolution

Edward Tunstel,* Tanya Lippincott and Mo Jamshidi
NASA Center for Autonomous Control Engineering
Department of Electrical and Computer Engineering
University of New Mexico
Albuquerque, NM 87131 USA

In *Intl. J. of Intelligent Automation and Soft Computing*, Vol. 3, No. 1, Special Issue on *Autonomous Control Engineering*, pp. 37–50, 1997

Abstract

Realization of autonomous behavior in mobile robots, using fuzzy logic control, requires formulation of rules which are collectively responsible for necessary levels of intelligence. Such a collection of rules can be conveniently decomposed and efficiently implemented as a hierarchy of fuzzy-behaviors. This article describes how this can be done using a behavior-based architecture. A behavior hierarchy and mechanisms of control decision-making are described. In addition, an approach to behavior coordination is described with emphasis on evolution of fuzzy coordination rules using the genetic programming (GP) paradigm. Both conventional GP and steady-state GP are applied to evolve a fuzzy-behavior for sensor-based goal-seeking. The usefulness of the behavior hierarchy, and partial design by GP, is evident in performance results of simulated autonomous navigation.

KEYWORDS: fuzzy-behavior, mobile robots, genetic programming, behavior hierarchy, behavior modulation, autonomous navigation.

1 Introduction

Real-time intelligent robot controllers are required to achieve the level of autonomy necessary in unstructured or "non-engineered" operating domains. For mobile robots the operating domain, whether an indoor industrial workplace, outdoor paved road, or planetary surface, dictates acceptable design and control approaches. One common requirement, independent of the operating domain, is a capability of intelligent navigation. Traditional approaches based on sequential task decomposition have met with difficulty in achieving real-time response. Controllers are now being developed that are modeled after natural processes and that advocate a behavioral decomposition of tasks with quasi-parallel execution. These behavior-based systems facilitate real-time intelligence by decomposing motion control capabilities into a set of special-purpose routines that operate concurrently.

Robust behavior in autonomous robots requires that uncertainty be accommodated by the robot control system. Fuzzy logic is particularly well suited for implementing such controllers due to its capabilities of inference and approximate reasoning under uncertainty. As such, we have focused our research efforts on developing an architecture that incorporates fuzzy control theory into the framework

*NASA/Jet Propulsion Laboratory, Pasadena, CA.

of behavior-based control for mobile robots. A mobile robot's behavior is encoded as a fuzzy rule-base that maps relevant sensor inputs into control outputs according to a desired control policy. Many fuzzy controllers proposed in the literature utilize a monolithic rule-base structure. That is, the precepts that govern desired system behavior are encapsulated as a single collection of *if-then* rules and, in most instances, the rule-base is designed to carry out a single control policy or goal. In order to achieve autonomy, mobile robots must be capable of achieving multiple goals whose priorities may change with time. Thus, controllers should be designed to realize a number of task-achieving behaviors that can be integrated to achieve different control objectives. This requires formulation of a large and complex set of fuzzy rules. In this situation a potential limitation to the utility of the monolithic fuzzy controller becomes apparent. Since the size of complete monolithic rule-bases increases exponentially with the number of input variables [2], multi-input systems can potentially suffer degradations in real-time response. This is a critical issue for mobile robots operating in dynamic surroundings. *Hierarchical* rule structures can be employed to overcome this limitation such that the size of the rule-base increases linearly with the number of input variables [2, 3].

This article describes a behavior-based robot navigation control system. Autonomy is achieved within a hierarchical structure of fuzzy-behaviors in which low-level navigation behaviors are realized as fuzzy logic controllers, while higher-level coordination behaviors are implemented as fuzzy decision systems. This hierarchy of fuzzy rule-bases enables distribution of intelligence amongst special-purpose *fuzzy-behaviors*. Its structure is motivated by the hierarchical nature of behavior as hypothesized in ethological models of animal behavior. A fuzzy coordination scheme is described that employs weighted decision-making based on contextual behavior activation. We also address the problem of automatic discovery/learning of coordination rules using genetic programming (GP) — an approach to computational intelligence. Performance results are presented on coordination behaviors evolved for the goal-seeking portion of a mobile robot behavior hierarchy. Simulated navigation experiments highlight interesting aspects of the decision-making process arising from behavior interaction. The focus here is on indoor navigation, however the architecture and approach can be extended for implementation on autonomous rovers or other vehicles designed for outdoor traversal in natural terrain.

2 Hierarchical Fuzzy-Behavior Control

The behavior control paradigm has grown out of an amalgamation of ideas from ethology, control theory and artificial intelligence [4, 5]. High-level motion control is decomposed into a set of special-purpose behaviors that achieve distinct tasks when subject to particular stimuli. Clever coordination of individual behaviors results in emergence of more intelligent behavior suitable for dealing with complex situations. The paradigm was initially proposed by Brooks [5] and realized as the 'subsumption architecture' wherein a behavior system is implemented as distributed finite state automata. Until recently [6]–[8], most behavior controllers have been based on crisp (non-fuzzy) data processing and binary logic-based reasoning. In contrast to their crisp counterparts, fuzzy-behaviors are synthesized as fuzzy rule-bases, i.e. collections of a finite set of fuzzy if-then rules. Each behavior is encoded with a distinct control policy governed by fuzzy inference. Thus, each fuzzy-behavior is similar to the conventional fuzzy controller in that it performs an inference mapping from some input space into some output space. If X and Y are input and output universes of discourse of a behavior with a rule-base of size n, the usual fuzzy if-then rule takes the following form

$$IF\ x\ is\ \tilde{A}_i\ THEN\ y\ is\ \tilde{B}_i \tag{1}$$

where x and y represent input and output fuzzy linguistic variables, respectively, and $\tilde{A}_i$ and $\tilde{B}_i$ $(i = 1...n)$ are fuzzy subsets representing linguistic values of x and y. In the mobile robot controller, the input x refers to sensory data or goal information; y refers to motor control signals or behavior activation levels.

The antecedent consisting of the proposition "x is $\tilde{A}_i$" could be replaced by a conjunction of similar propositions; the same holds for the consequent "y is $\tilde{B}_i$".

2.1 Behavior Hierarchy

The proposed architecture is a conceptual model of an intelligent behavior system and its behavioral relationships. Overall robot behavior is decomposed into a bottom-up hierarchy of increased behavioral complexity in which activity at a given level is a function of behaviors at the level(s) below. A collection of *primitive behaviors* resides at the lowest level which we refer to as the primitive level. These are simple, self-contained behaviors that serve a single purpose by operating in a reactive (non-deliberative) or reflexive (memoryless) fashion. They perform nonlinear mappings from different subsets of the robot's sensor suite to common actuators. Each primitive exists in a state of solipsism[1], and alone, would be insufficient for autonomous navigation tasks. Primitive behaviors are building blocks for more intelligent *composite behaviors*. Their capabilities can be modulated through synergistic coordination to produce composite behavior(s) suitable for goal-directed operations.

Mobile robots operating in non-engineered domains must be capable of reliable navigation in the presence of static and dynamic obstacles (e.g. humans and/or other moving robots). It is preferred that such robots be designed to autonomously navigate, in an equally effective manner, in both sparsely populated environments (e.g. an overnight security robot patrolling an office building) and in cluttered environments (e.g. a robot transporting material on a busy factory floor). Our approach to providing the necessary autonomy for such robots is to construct behavior hierarchies suitable for the task(s) and operating environment. A behavior hierarchy for indoor navigation might be organized as in Figure 1. It implies that goal-directed navigation can be decomposed as a behavioral function of `goal-seek` (collision-free navigation to some location) and `route-follow` (assuming some direction is given in the form of waypoints or a path plan). These behaviors can be further decomposed into the primitive behaviors shown, with dependencies indicated by the adjoining lines. `Avoid-collision` and `wall-follow` are self-explanatory. The `doorway` behavior guides a robot through narrow passageways in walls; `go-to-xy` directs motion along a straight line trajectory to a particular location. The circles represent weights and activation thresholds of associated primitive behaviors. As described below, fluctuations in these weights are at the root of the intelligent coordination of primitive behaviors which leads to adaptive system behavior. The hierarchy facilitates decomposition of complex problems as well as run-time efficiency by avoiding the need to evaluate rules from behaviors that do not apply.

As an additional example, consider the outdoor navigation problem encountered by natural terrain vehicles such as planetary rovers. Autonomous rovers must be capable of point-to-point navigation in the

[1] A primitive behavior is only cognizant of its own purpose and knows nothing about other behaviors.

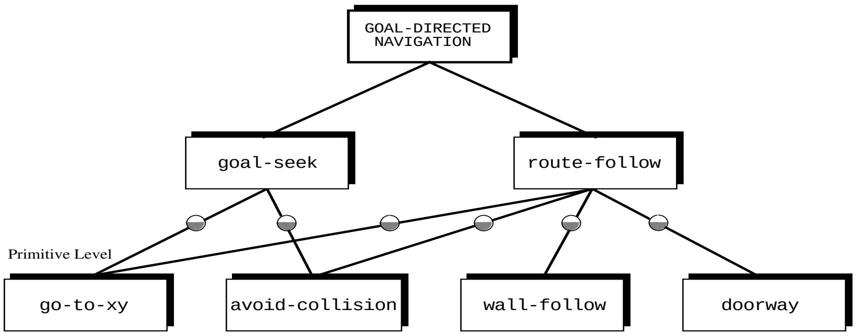

Figure 1: Hierarchical decomposition of mobile robot behavior.

presence of varying obstacle (rocks, boulders, dense vegetation, etc.) distributions, surface characteristics, and hazards. Often the task is facilitated by knowledge of a series of waypoints, furnished by humans, which lead to designated goals. In some cases, such as exploration of the surface of Mars [9, 10], this supervised autonomous control must be achieved without the luxury of continuous remote communication between the mission base station and the rover.[2] Considering these and other constraints associated with rover navigation, suitable behavior hierarchies similar to the hypothetical one shown in Figure 2 could be constructed. In this figure the behavioral functions of `goal-seek`, `route-follow`, and an additional composite behavior, `localize` are decomposed into a slightly different suite of primitive behaviors. The design of behaviors at the primitive level would be tailored to the navigation task and an environment with characteristics of natural terrain.

Note that decomposition of behavior for a given mobile robot system is not unique. Consequently, suitable behavior repertoires and associated hierarchical arrangements are arrived at following a subjective analysis of the system and the task environment. The total number, and individual purpose, of fuzzy-behaviors in a given behavior hierarchy is indicative of the problem complexity and can be conveniently modified as required.

3 Coordination by Behavior Modulation

During any sufficiently complex navigation task, interactions in the form of behavioral cooperation or competition occur when more than one primitive behavior is active. These forms of behavior are not perfectly distinct; they are extremes along a continuum [11]. Instances of behavior throughout the continuum can be realized using *behavior modulation*, which we define as the autonomous act of regulating, adjusting or adapting the activation level of a behavior to the proper degree in response to a context, situation, or state perceived by an autonomous agent. In a given implementation the "proper degree" is governed by the desired behavioral response of the agent.

In our hierarchical approach, coordination is achieved by weighted decision-making and behavior modulation embodied in a concept called the *degree of applicability* (DOA), which is a measure of the instantaneous level of activation of a behavior. It can be thought of in ethological terms as a motivational tendency of the behavior. Fuzzy rules for coordination are formulated to include weighting consequents which modulate the DOAs of lower-level primitives. The DOA, $\alpha_j \in [0,1]$, of primitive behavior j is specified in the consequent of *applicability rules* of the form

$$IF\ x\ is\ \tilde{A}_i\ THEN\ \alpha_j\ is\ \tilde{D}_i \tag{2}$$

[2]Time delays between Earth and Mars can be anywhere between 6 and 41 minutes.

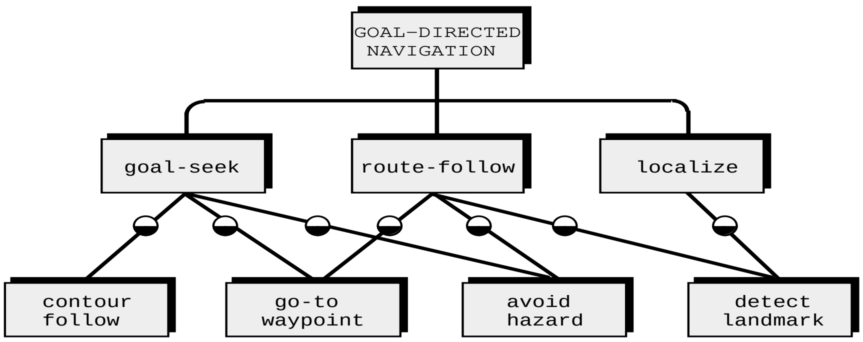

Figure 2: Hypothetical behavior hierarchy for rover navigation.

where as in (1), x is an input linguistic variable and $\tilde{A}_i$ is a fuzzy subset representing its linguistic value. $\tilde{D}_i$ is a fuzzy subset specifying the linguistic value (e.g. "*high*") of α_j for the situation prevailing during the current control cycle. This feature allows certain robot behaviors to influence the overall behavior to a greater or lesser degree depending on the current situation. It serves as a form of motivational adaptation since it causes the control policy to dynamically change in response to goals, sensory input, and internal state. Thus, composite behaviors are meta-rule-bases that provide a form of the ethological concepts of inhibition and dominance. Behaviors with maximal applicability ($\alpha_{max} \leq 1$) can be said to dominate, while behaviors with partial applicability ($0 < \alpha < \alpha_{max}$) can be said to be inhibited. These mechanisms allow exhibition of behavioral responses throughout the continuum. This is in contrast to crisp behavior selection which typically employs fixed priorities that allow only one activity to influence the robot's behavior during a given control cycle. The coordination scheme includes behavior selection as a special case when the DOA of a primitive behavior is nonzero and above its activation threshold while others are zero or below threshold. When this occurs, the total number of rules to be consulted on a given control cycle is reduced. The reduction in rule evaluations is not as dramatic or static as in the strict rule hierarchies proposed in [2, 3] since we are dealing with a *behavior* hierarchy that achieves interacting goals. As such, the number of rules consulted during each control cycle varies dynamically as governed by the DOAs and thresholds of the behaviors involved.

Coordination and conflict resolution are achieved within the framework of fuzzy logic theory — via operations on fuzzy sets [12]. Fuzzy rules of each applicable primitive behavior are processed yielding respective output fuzzy sets. These fuzzy sets are equivalent to the result produced by rule-base evaluation in conventional fuzzy controllers *before* applying the defuzzification operator. Following consultation of applicable behaviors, each fuzzy-behavior output is weighted (multiplied) by its corresponding DOA, thus effecting its activation to the level prescribed by the composite behavior. The resulting fuzzy sets are then aggregated using an appropriate t-conorm operator (i.e. a fuzzy union/disjunction operator such as arithmetic maximum), and defuzzified to yield a crisp output that is representative of the intended coordination. Since control recommendations from each applicable behavior are considered in the final decision, the resultant control action can be thought of as a consensus of recommendations offered by multiple experts.

3.1 Related Work

This strategy for multiple behavior coordination was developed to enable *robust* autonomous performance. It represents an approach that is particularly suitable in the context of fuzzy-behavior hierarchies. Several instances of independent research have converged to similar ways of approaching autonomous mobile robot navigation [7][13]–[15].

Experience thus far has revealed that in many practical instances fuzzy control alone is insufficient for addressing complex intelligent control problems of robotics. Furthermore, humans often find it difficult to design knowledge-based control systems with interacting rule-bases, particularly in the absence of experts or sufficient knowledge of the problem. In what follows, we address the problem of automatic discovery/learning of coordination rules for use in the proposed coordination strategy. This problem has been previously approached in the contexts of other coordination schemes by using reinforcement learning [16] and hybrids of reinforcement and neural networks [17, 18]. In [19], we demonstrated the potential of the GP paradigm for learning fuzzy rule-bases for low-level regulation and tracking types of problems. The reader interested in details of its implementation may consult the reference. Using this earlier work as a foundation, we apply our GP approach here to higher-level behavior coordination.

4 Genetic Programming of Fuzzy-behaviors

The GP paradigm [20] computationally simulates the Darwinian evolution process by applying fitness-based selection and genetic operators to a population of parse trees representing computer programs of a given programming language. It departs from the conventional genetic algorithm primarily with regard to its representation scheme. Structures undergoing adaptation are executable hierarchical programs of dynamically varying size and structure, rather than numerical strings. In our GP system a population comprised of fuzzy rule-bases, that are candidate solutions to the problem, evolves in response to selective pressure induced by their relative fitnesses for implementing the desired behavior.

4.1 Rule Discovery

In the process of learning fuzzy rules, GP manipulates the linguistic variables directly associated with the fuzzy-behaviors. The dynamic variability of the representation allows for rule-bases of various sizes, thus enhancing population diversity which is important for the success of the GP system. It also increases the potential for discovering rule-bases of smaller sizes than necessary for completeness[3] but sufficient for realizing desired behaviors. In this work, lower and upper bounds of 8 and 27 were imposed on rule-base size. The search space is contained in the set of all possible rule-bases that can be composed recursively from a set of *functions* and a set of *terminals*. The function set consists of components of the generic fuzzy *if-then* rule and common fuzzy logic connectives, i.e. functions for antecedents, consequents, fuzzy intersection, rule inference, and fuzzy union. Each rule-base is an executable program that evaluates to an output fuzzy set resulting from fuzzy inference. The terminal set is made up of the input and output linguistic variables and pre-specified membership functions associated with the desired behavior.

Given suitable function and terminal sets GP proceeds by randomly generating an initial population of behaviors. This is followed by evaluating each behavior in the current population and applying, primarily, reproduction and crossover to behaviors selected with a probability based on fitness. Crossover is done by swapping portions of parse trees from two parent behaviors such that proper syntax is preserved. Upon satisfaction of termination criteria, the GP result is the best-fit behavior that appeared in any generation. In addition to this generational process, a steady-state evolution can be applied as in the Steady State Genetic Algorithm (SSGA) [21] which has recently been applied in GP for behavior evolution [22, 23]. In the SSGP approach the concept of "generations" does not exist. Instead, on each iteration following creation of the initial population only two new offspring are produced; the offspring replace two of the worst individuals in the population. In our approach, parent behaviors are selected to produce offspring by tournament[4], while behaviors to be removed are chosen randomly from the set of below-average behaviors in the current population.

5 Evolution of Coordination

When behaviors compete for control of the robot by recommending different control actions for common actuators, the problem becomes one of coordinating multiple task-achieving behaviors and resolving conflicts amongst them. In order to formulate suitable coordination rules, one must first decide what the DOAs of low-level behaviors should be in all practical situations perceived from sensory input. Formulation of such rules for coordination is not entirely intuitive, and expert knowledge about how to concurrently coordinate primitive behaviors is not readily available. We apply GP for this purpose. A fuzzy coordination behavior for goal seeking, comprised of rules in the form of Equation (2), is what we wish to evolve.

[3] A rule-base is *complete* if there exists a rule for any valid combination of inputs

[4] Three behaviors are chosen at random from the population; the behavior with the best fitness is selected.

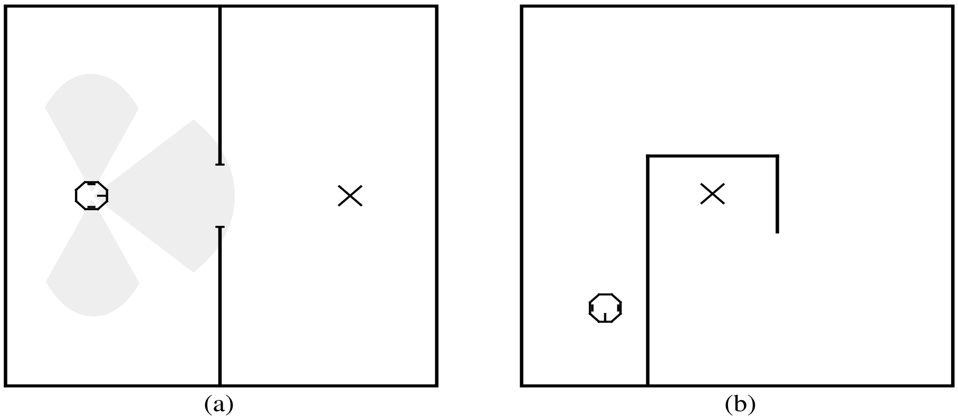

Figure 3: Example fitness cases.

In the current implementation, applicability rules used by `goal-seek` to modulate the underlying primitive behaviors consider three instantaneous input states — the range to the nearest obstacle, the distance from the goal, and the angular heading to the goal. The consequents of these rules prescribe a DOA for each primitive behavior.

5.1 Behavior Fitness Evaluation

During evolution each behavior in the current population is evaluated via simulation in a number of indoor *fitness cases* subject to an upper time limit of 200 seconds. In this work, $n_f = 5$ fitness cases are used; the simplest and most difficult of these are illustrated in Figure 3(a) and 3(b), respectively. Goal locations in the figure are indicated by an X, the robot is depicted as an octagonal icon with a radial line designating its initial heading, and its range sensor horizon is indicated by the shaded regions of Figure 3(a). In each case, the dimension of the indoor space is $10m \times 10m$. Each fitness case was chosen to represent situations likely to be encountered in indoor environments.

For a given behavior, the score of a trial run through fitness case i is given by

$$S_i = \begin{cases} 100 & ; \ \textit{goal reached} \\ \frac{100}{\gamma(1+10e_N)} & ; \ \textit{otherwise} \end{cases} \tag{3}$$

where e_N is the normalized residual distance to the goal in the case of a time-out or collision. The parameter $\gamma = 2$ if a collision occurs; otherwise $\gamma = 1$. That is, the score for an unsafe trial is half of that for a collision-free trial with all else being equal (see Figure 4). The overall fitness of the behavior is the average score over all n_f fitness cases:

$$F = \frac{1}{n_f} \sum_{i=1}^{n_f} S_i \tag{4}$$

Thus, the highest possible score, and hence fitness, is 100. In evolutionary algorithms, such as GP, it is important that the fitness function map observable parameters of the problem into a spectrum of values that differentiate the performance of individuals in the population. If the spectrum of fitness values is not sufficiently rich, the fitness function may not provide enough information to guide GP toward regions of the search space where improved solutions might be found. The fitness function and score were formulated with this in mind, and also to reward behaviors responsible for consistently reaching, or coming within close proximity to, the goals.

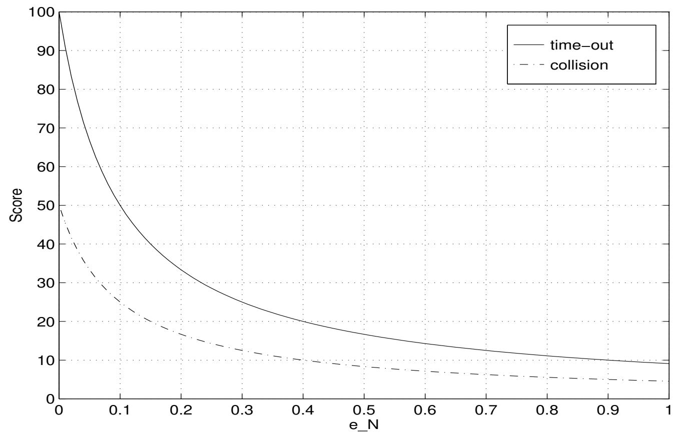

Figure 4: Behavior fitness case scoring function.

6 Autonomous Navigation Results

In order to demonstrate the operational aspects of the controller in the simplest manner possible we consider only the composite behavior — `goal-seek`. As illustrated in Figure 1, its effect arises from synergistic interaction between primitive behaviors, `go-to-xy` and `avoid-collision`. When more behaviors are involved the approach proceeds in a straightforward manner by appending additional DOAs and any necessary antecedents to applicability rules accordingly.

The simulated mobile robot is modeled after LOBOt, a custom-built base with a 2-wheel differential-drive and two stabilizing casters. It is octagonal in shape, 75 cm tall and 60 cm in width. The sensor suite includes optical encoders on each driven wheel and 16 ultrasonic transducers arranged primarily on the front, sides, and forward-facing obliques. Ideal pose information $(x\ y\ \theta)^T$ is assumed and is computed using a kinematic model of the differential-drive mechanism. Its maximum speed was limited to $0.3m/s$ which is sufficiently slow to justify the use of a kinematic model only. The sensor model generated range readings with errors as large as $\approx 100mm$ and lower resolution than the actual sonar. The simulated operating domain is a hypothetical indoor layout of walls and furniture not unlike an industrial workplace.

6.1 Hand-derived behavior:

As a result of an arduous trial-and-error procedure, the authors were able to formulate a suitable set of 11 coordination rules for the `goal-seek` behavior. The navigation task of Figure 5 illustrates a fragmented view of the primitive capabilities and the result of their modulation. The initial state of the simulation is shown in Figure 5a with LOBOt located at a docking station with pose $(11.7\ 12.3\ \frac{\pi}{2})^T$. Its task is to navigate to the goal located at (1.5 1), marked by the X. The `avoid-collision` and `go-to-xy` behaviors are each shown acting alone in Figure 5b and 5c, respectively. Recall that these behaviors are only capable of exhibiting their particular primitive roles. Thus, `avoid-collision` merely displays cyclic collision-free motion in the immediate vicinity of the robot's initial location, while `go-to-xy` displays a taxic reaction that propels the robot toward the goal irregardless of obstacles in its path. Successful completion of the task, resulting from modulation of the primitive behaviors, is shown in Figure 5d.

In Figure 6, the behavioral interaction during the run is shown as a time history of the DOAs of each primitive behavior. The interaction dynamics show evidence of brief bouts of competition (overlapping oscillations) and cooperation with varying levels of dominance. Initially, `avoid-collision` has the dominant influence over the robot due to the close proximity of walls at the docking station. It virtually

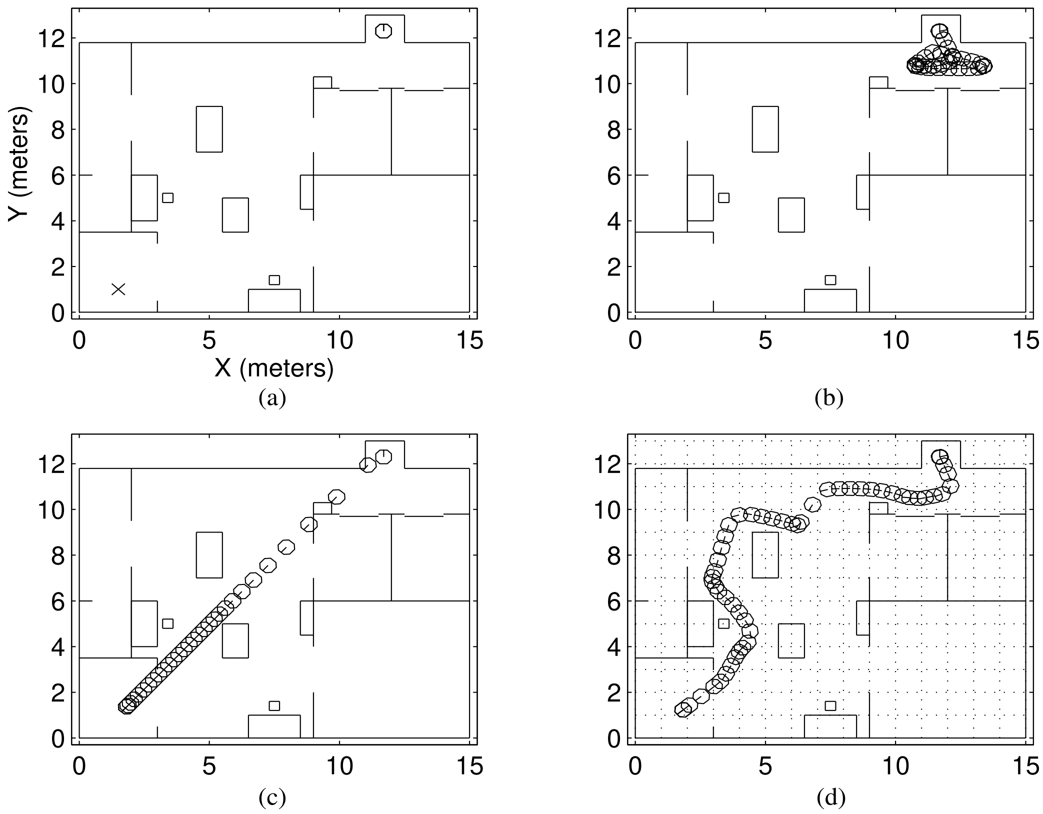

Figure 5: Ingredients of goal-seeking behavior.

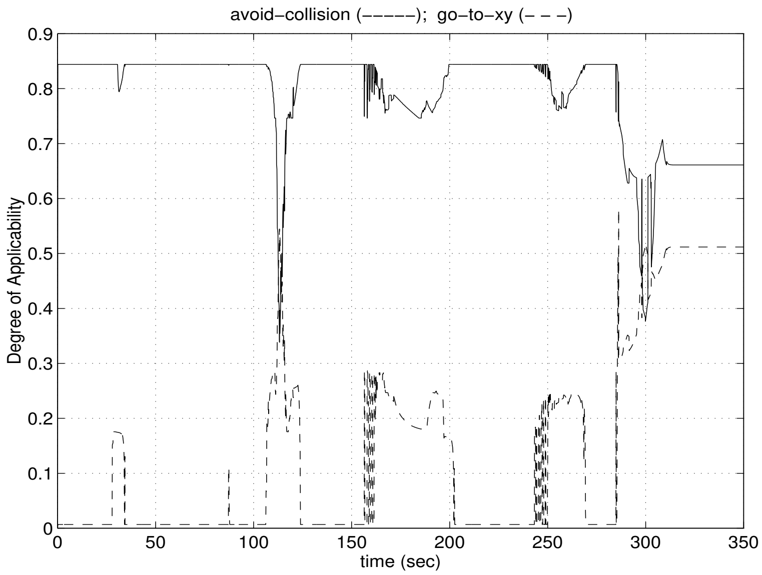

Figure 6: Behavior modulation during goal-seeking.

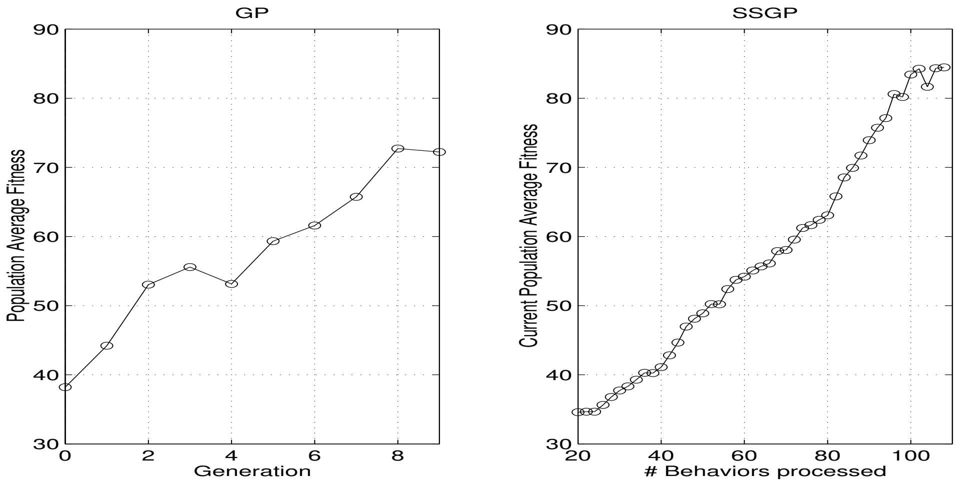

Figure 7: Mean performance of GP and SSGP evolution.

maintains dominance throughout the task due to the relatively uniform clutter in the environment. The first bout of competition corresponds to the robot's approach toward the obstacle located at $(5, 8)$; a second bout occurs as it enters the goal room. Elsewhere, applicabilities vary continuously reflecting levels of activation recommended by the behavior control system.

6.2 Evolved behavior:

The operating domain is considerably more general than any one of the fitness cases used during the evolution process and, thus, provides a suitable environment to test the generalization capability of the evolved behaviors. The GP system was run using population sizes of 10–20 rule-bases for a number of generations ranging from 10–15. In GP, genetic diversity remains high even for very small populations due to the tree structure of individuals [20]. Steady-state GP was also applied using a population size of 20. Results of runs using both approaches are summarized graphically in Figure 7. The mean performance of GP over five runs is shown, in the left half of Figure 7, as the progression of the population average fitness during the first ten generations. The right half of Figure 7 shows the progression of the average fitness of the current population at each iteration. Twenty behaviors were processed in the initial population; thereafter, two new behaviors evolved at each iteration. A trend towards higher fitness is evident for both GP and SSGP. Table 1 lists some quantitative details about the best behavior evolved by each approach. The success rate was determined from navigation runs in three different simulated domains not included in the set of fitness cases. For this problem good regions of the search space were discovered with less processing by SSGP.

Table 1: Best evolved composite `goal-seek` behaviors.

	Population size	#Generations	#Rules	Best fitness	Success rate
GP	10	15	11	86.5	70%
SSGP	20	N/A	9	87.8	83%

Having pointed out some operational details of the behavior hierarchy, let us compare the performance of the hand-derived `goal-seek` behavior to a behavior evolved for the same purpose. We will consider an arbitrary point-to-point navigation task from initial state $(1\ 11\ -\frac{\pi}{2})^T$ to a goal located at

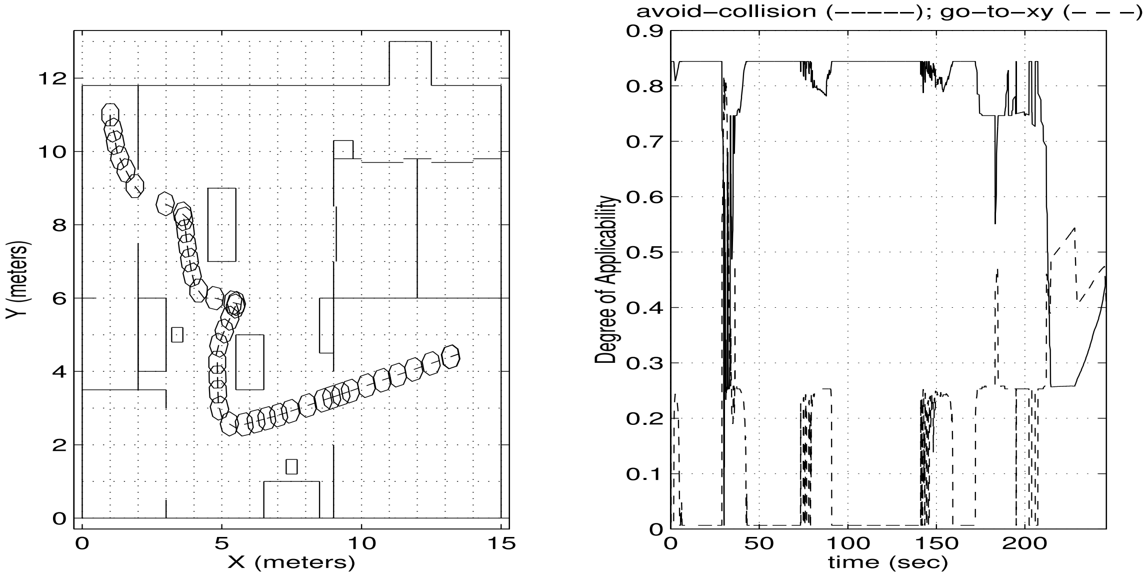

Figure 8: Hand-derived coordination and behavior modulation.

(13.5 4.5). The successful path executed by the hand-derived behavior is shown in Figure 8 along with the corresponding behavior modulation history. We compare this with the same task executed by the best SSGP-evolved behavior shown in Figure 9. The evolved behavior coordination results in a more direct path to the goal due to higher motivation applied to `go-to-xy`. The resulting path in this example is executed about 20% faster than the path taken via hand-derived coordination. We also note that the behavior modulation demonstrated by the evolved behavior is more complex. Near uniform bouts of competition and cooperation throughout the task are evident in the decision-making, thus leading to similar amounts of behavioral influence for each primitive behavior. As listed in Table 1, this was achieved using less applicability rules than both the hand-derived behavior and the best GP-evolved behavior. For an identical navigation task, the relative levels of activation induced by the SSGP-evolved `goal-seek` behavior more closely resembles a consensus.

7 Conclusion

The hierarchy of fuzzy-behaviors provides an efficient approach to controlling mobile robots. Its practical utility lies in the decomposition of overall behavior into sub-behaviors that are activated only when applicable. When conditions for activation of a single behavior (or several) are satisfied, there is no need to process rules from behaviors that do not apply. This would result in unnecessary consumption of computational resources and possible introduction of "noise" into the decision-making process. Behavior modulation allows realization of a continuum of behavioral responses governed by respective behavioral degrees of applicability. This multiple-behavior coordination scheme includes behavior selection as a special case. The modularity and flexibility of the approach, coupled with its mechanisms for weighted decision-making, makes it a suitable framework for modeling and controlling situated adaptation in autonomous robots.

Genetic programming proved useful for learning fuzzy-behaviors at the coordination level of the hierarchy. In particular, rules were evolved for coordinating low-level fuzzy-behaviors which reside at the primitive level. Here, we have focused on the goal-seeking portion of a behavior hierarchy, however the GP approach is generally applicable throughout. Conventional GP and steady-state GP were applied,

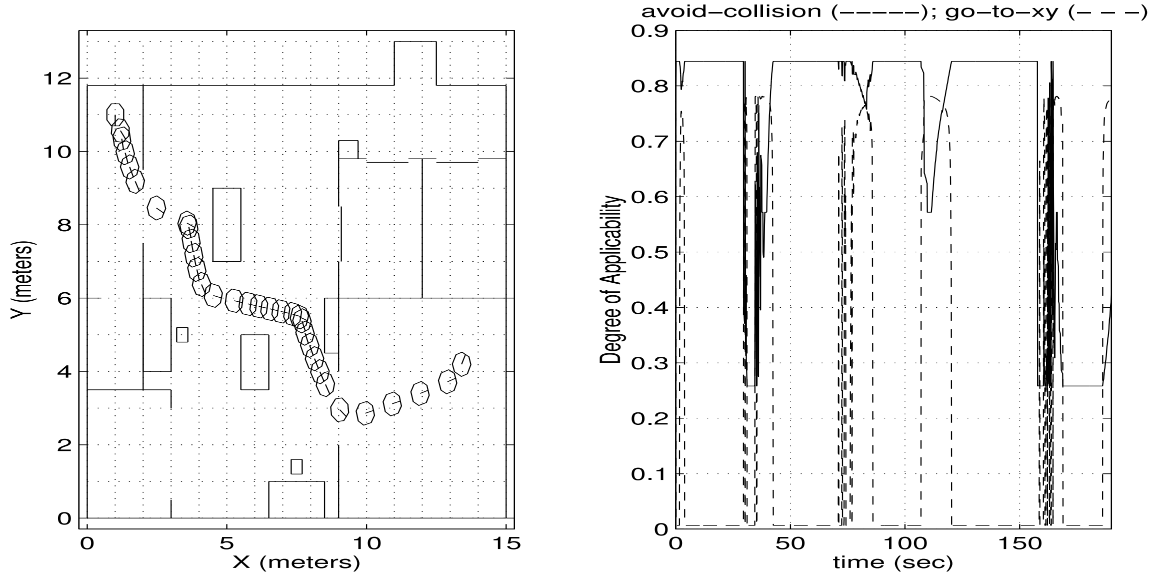

Figure 9: SSGP-evolved coordination and behavior modulation.

each yielding good results for small populations. Overall, SSGP yielded slightly better results for goal-seeking coordination. Using only five fixed fitness cases during behavior evolution modest generalization capabilities were exhibited by the highest fit behaviors. An SSGP-evolved behavior showed a better behavior modulation capability than both the hand-derived behavior and the best GP-evolved behavior. It is expected that additional improvement can be achieved using a richer set of fitness cases.

The results presented here apply only to autonomous navigation in indoor environments. The architecture can be applied for outdoor navigation as well. In future research, implementations for outdoor and rugged terrain vehicles such as planetary rovers will be considered.

Acknowledgments:
This work was supported in part by NASA under contract # NCCW-0087. The authors would also like to acknowledge the research assistance and contribution of Harrison Danny to this work.

References

[1] **Jamshidi, M., N. Vadiee and T. Ross** (Eds.) *Fuzzy Logic and Control: Software and hardware applications.* Prentice-Hall, Englewood Cliffs, NJ, 1993.

[2] **Raju, G.V.S., J. Zhou and R.A. Kisner** "Hierarchical Fuzzy Control." *Intl. J. of Control*, Vol. 54, No. 5, pp. 1201–1216, 1991.

[3] **Bruinzeel, J., M. Jamshidi and A. Titli** "A Sensory Fusion-Hierarchical Real-Time Fuzzy Control Approach for Complex Systems." Technical Report No. 95347. LAAS-CNRS, Toulouse, France, August 1995.

[4] **McFarland, D.J.** *Feedback Mechanisms in Animal Behavior.* Academic Press, New York, 1971.

[5] **Brooks, R.A.** "A Robust Layered Control System for a Mobile Robot." *IEEE Journal of Robotics & Automation*, Vol. RA-2, No. 1, pp. 14–23, 1986.

[6] **Pin, F.G. and Y. Watanabe** "Navigation of Mobile Robots Using a Fuzzy Behaviorist Approach and Custom-Designed Fuzzy Inferencing Boards." *Robotica*, Vol. 12, No. 6, pp. 491–504, 1994.

[7] **Saffiotti, A., K. Konolige and E.H. Ruspini** "A Multivalued Logic Approach to Integrating Planning and Control." *Artificial Intelligence*, Vol. 12, pp. 481–526, 1995.

[8] **Tunstel, E. and M. Jamshidi** "Fuzzy Logic and Behavior Control Strategy for Autonomous Mobile Robot Mapping." *IEEE Intl. Conf. on Fuzzy Systems*, pp. 514–517, June 1994.

[9] **Matthies, L., E. Gat, R. Harrison, B. Wilcox, R. Volpe and T. Litwin** "Mars Microrover Navigation: Performance evaluation and enhancement." *Autonomous Robots*, Vol. 2, No. 4, pp. 291–311, 1995.

[10] **Matijevic, J. and D. Shirley** "The Mission and Operation of the Mars Pathfinder Microrover." *IFAC 13th Triennial World Congress*, San Francisco, 1996.

[11] **Staddon, J.E.R.** *Adaptive Behavior and Learning.* Cambridge University Press, New York, 1983.

[12] **Tunstel, E.** "Coordination of Distributed Fuzzy Behaviors in Mobile Robot Control." *IEEE Intl. Conf. on Systems, Man and Cybernetics*, pp. 4009–4014, October 1995.

[13] **Moreno, L., E. Moraleda, M.A. Salichs, J.R. Pimentel and A. de la Escalera** "Fuzzy Supervisor for Behavioral Control of Autonomous Systems." *Intl. Conf. on Industrial Electronics, Control, and Instrumentation*, pp. 258–261, 1993.

[14] **Michaud, F., G. Lachiver and C.T. Le Dinh** "Fuzzy Selection and Blending of Behaviors for Situated Autonomous Agent." *IEEE Intl. Conf. on Fuzzy Systems*, pp. 258–264, September 1996.

[15] **Correia, L. and A. Steiger-Garção** "A Useful Autonomous Vehicle with a Hierarchical Behavior Control." *Advances in Artificial Life, 3rd European Conf. on Artificial Life.* F. Móran, A. Moreno, J.J. Merelo and P. Chácon (Eds.), Springer-Verlag, Granada, Spain, pp. 625–639, June 1995.

[16] **Bonarini, A.** "Learning to Coordinate Fuzzy Behaviors for Autonomous Agents." *2nd European Congress on Intelligent Techniques and Soft Computing EUFIT'94*, pp. 475–479, 1994.

[17] **Beom, H.R., K.C. Koh and H.S. Cho** "Behavioral Control in Mobile Robot Navigation Using Fuzzy Decision Making Approach." *IEEE Intl. Conf. on Robots and Systems*, pp. 1938–1945, 1994.

[18] **Gachet, D., M.A. Salichs, L. Moreno and J.R. Pimentel** "Learning Emergent Tasks for an Autonomous Mobile Robot." *IEEE Intl. Conf. on Robots and Systems*, pp. 290–297, 1994.

[19] **Tunstel, E. and M. Jamshidi** "On Genetic Programming of Fuzzy Rule-Based Systems for Intelligent Control." *Intl. Journal of Intelligent Automation and Soft Computing*, Vol. 2, No. 3, pp. 271–284, 1996.

[20] **Koza, J.R.** *Genetic Programming: On the Programming of Computers by Means of Natural Selection.* MIT Press, Cambridge, MA, 1992.

[21] **Syswerda, G.** "A Study of Reproduction in Generational and Steady-State Genetic Algorithms." *Foundations of Genetic Algorithms.* G. Rawlins (Ed.), Morgan Kaufmann, San Mateo, CA, 1991.

[22] **Reynolds, C.W.** "Evolution of Corridor Following Behavior in a Noisy World." *3rd Intl. Conf. on Simulation of Adaptive Behavior*, pp. 402–410, August 1994.

[23] **Nordin, P. and W. Banzhaf** "Genetic Programming Controlling a Miniature Robot in Real Time." Technical Report 4/95, CS Dept. University of Dortmund, Germany, 1995.