Resolving Conflicts Among Actions in Concurrent Behaviors: Learning to Coordinate

Henry Hexmoor
University of North Dakota
Grand Forks, North Dakota, USA
hexmoor@cs.und.edu

Abstract

A robotic agent must coordinate its coupled concurrent behaviors to produce a coherent response to stimuli. Reinforcement learning has been used extensively in coordinating sensing to acting of a single behavior and it has been shown useful in loosely coupled concurrent behaviors. We present a technique for applying Q values developed in learning individual behaviors for coordination among coupled concurrent behaviors.

1 Introduction

In psychology, it is argued that task execution can be divided into three successive stages of (a) perceiving the stimulus, (b) choosing the response, (c) and producing the response (Pashler, 1997). The strongest theory is that the second stage is the bottleneck for concurrent tasks. This means that the agent's sensory "attention" can operates in parallel. It is division of attention to concurrent behaviors that is difficult.

Consider a robot with multiple, concurrently active behaviors. Each behavior generates a desired response (i.e., an action) corresponding to the robot's sensory input. The responses from concurrent behaviors might contradict or conflict. For example, a robot with *obstacle avoidance* and *wall following* behaviors upon sensing a wall might generate a response to turn right and another response to turn left. If it executed both actions at the same time, it would end up with an inappropriate action of going straight. However, we might be able to blend the two actions in a differential manner. In contrast, consider a robot with a single camera that is used to *follow a target object* and to *avoid obstacles*, it may have conflicting gaze control responses to look ahead and look at the target. This example is usually considered to be a resource-sharing problem and since usually the robot can switch its gaze back and forth, the solution is one of frequency of switching and time-sharing. We have addressed this resource-sharing problem with reinforcement learning (Hexmoor and Shapiro, 1997). The first example involves behaviors that are more closely coupled than the second and is the focus of this paper. For a survey article on behavior coordination see (Pirjanian, 1999).

Furthermore, consider that the robot is learning to improve its performance in each concurrent behavior using the standard reinforcement based learning known as Q learning. There is minimal domain knowledge in the concurrent behaviors and no direct information sharing among behaviors. We have developed a technique that uses Q values to resolve the contradiction between behaviors. As a behavior is learned, the robot responds with a very decisive choice for some sensory input making these inputs critical (or deterministic) and with a less decisive choice for some other sensory input making these inputs non-critical (or nondeterministic). When a set of inputs is critical for a behavior, we consider the behavior to be more important to the robot. This is as if that behavior deserves more attention from the robot. We can consider this situation as an influence that this behavior in its critical region possesses over other behaviors. Another way to think of this situation is a degree of subsumption from the behavior in its critical region over other concurrent behaviors. This subsumption degree can be used as an explicit measure to resolve contradictions in behavior output either as a cancellation of other behaviors or a differential activation level given to contradictory behaviors.

Our scenario is similar to considerations described in (Sen and Sekaran, 1998). They addressed multiagent coordination strategies using Q learning as well as genetic approaches. The competing actions belong to different agents instead of a single agent's concurrent behaviors. Issues of coordination among different agents as Sen and Sekaran addressed, and behaviors within a single agent as we have addressed, are similar if we consider cooperative attitude among agents. Their work showed that prior or explicit information among agents was not needed to learn coordination. However, they reveal that when agent actions are strongly coupled, an individual learning is unable to produce effective coordination. In our approach behaviors are simultaneously learned using the Q learning algorithm. Each behavior is assigned its own reward scheme. Our introduction of suppression

rates among behaviors solves the coordination problem even in highly coupled behaviors.

Reinforcement based learning is used in the concurrent activities of robotic soccer (Asada, Uchibe, and Hosoda, 1999). This work uses a coordination discount factor that is multiplied to the Q values of separately learned behaviors when the behaviors are conflicting, which they call re-learning area. For an example, consider a robot that must simultaneously (a) shoot a ball into a goal area and (b) avoid collision with a goal keeper robot. Such a robot may learn either behavior separately, but when it can do both, the coordination factor will adjust the learned Q values to prevent interference between behaviors. In this case, shooting straight to the goal will not select if the ball can collide with the goalkeeper. A sub-optimal shooting will be produced that might still do the job. Instead of adjustments to individual Q values by coordination parameter, our approach implements a direct suppression between behaviors beyond the instantaneous Q value. Whereas their coordination factor is an interpolation of Q values within one behavior, our inter-behavior suppression rates implement direct impact of one behavior's Q standard deviation over another behavior.

There are neutral network approaches for sensory-motor coordination such as the work described in (Kuperstein, 1989) for coordinating hands and eyes in reaching elongated objects. This notion of coordination is association of senses with motor actions. The coordination is not between competing behaviors. As such this work compares to application of reinforcement based learning to individual behaviors.

A learning algorithm that accounts for reliability and relevance of behaviors is presented in (Maes and Brooks, 1991). This algorithm coordinates leg movement of a six-legged robot called Genghis. Relevance is the difference between reward correlation and punishment correlation. Correlation is the Pearson product-moment correlation coefficient of frequencies of receiving reward and punishment. This algorithm has worked wonderfully with Genghis but the behaviors were treated as on-off signals of separate leg movements. Our behaviors allow a choice of actions and for each behavior we compute influence over other behaviors.

2 The algorithm

Our robots run the following steps (shown in the box) in an infinite loop.

1. Determine the current set of concurrent behaviors
2. Determine the current state of the world
3. Match the current state with behavior preconditions and select an action for each concurrent behavior for execution
   3a. α percent of time pick the action with highest Q value
   3b. (1-α) percent of time pick a random action
   3c. Fully execute the action of the behavior with the highest influence (the standard deviation among actions of this behavior is larger than 60%-- This is a parameter) but suppress (i.e., do not execute) actions of other behaviors
4. Decay the impact of actions from the previous cycle
5. Compute rewards for Q learning
6. Q learn each behavior individually
7. Determine the instantaneous inter-behavior influences for coordination
8. If there is state action deviation, increase the percentage α, else reduce α toward its initial level
9. Update values for next cycle
10. Compute system performance: ignoring navigation in no conflict zones, record the standard deviation of conflicting simultaneous actions issued by concurrent behaviors

The system operates very much like a forward branching production system, matching current state (which reflects the content of working memory) with the behaviors, which are implemented as sets of productions. Each production in a behavior has preconditions, which are states of the world, and a disjunction of actions that can be selected for execution. Only one action is selected per behavior at each cycle. I.e., Each behavior can produce one of several actions and it will be the comparison of the Q values developed by Q learning (our machine learning technique) to pick the best action. Each action of a behavior has a Q value, which is its suitability. Q values are set at an initial level and are updated using Q learning. I will not explain Q learning. There are many good sources for this, such as (Whitehead, 1991). We set our parameter λ to 0.25 and γ to 0.75 for Q learning.

In the tradition of other reinforcement-based learning systems, conflict resolution among potential actions of a behavior, the action with the highest Q value is preferred most of the time (α in the algorithm). The rest of the time (1 - α), we pick an action randomly so we experiment with actions that may turn out to be successful and earn high Q values.

State Action Deviation (SAD) is when the robot performs an action but the next state is the same as the

current state. This can be due to conflict in concurrent actions or due to inefficacy of actions. For example, the robot can try to turn left and right by the same amount simultaneously, canceling the effect. We initially set our experimentation level at 10% (1 - α in the algorithm) and when SAD is higher than one, increment of this value. When SAD is reduced, we also reduce the experimentation level.

In order to produce smooth actions, our system remembers actions from the previous cycle and decays their effect (to about 10%), which is added to the current actions.

The performance of coordination is measured as the standard deviation of degrees of turn among actions that are issued simultaneously by concurrent behaviors. We only measure this value when the behaviors are both active and produce an action. For instance, if only one concurrent behavior produces an action, there is no possibility of conflict, and it is not counted in our performance measurement.

3 Experiments and results

We developed a Nomad mobile robot experiment with two behaviors. The first behavior implements *wandering and obstacle avoidance*. Our Nomad robot has 16 sonars, arranged on the robot circumference and 22.5 degrees apart. The robot uses its front 7 sonar sensors, spanning 135 degrees in front of the robot to detect obstacles. If it detects obstacles it is allowed to choose one of seven actions: to go straight, or to turn 10, 25, or 60 degrees in either left or right direction. If there are no obstacles, it may go straight or pick one of its six turn actions in random. These 7 actions are the only actions available to our robot.

To be specific, for the first behavior we have 7 productions, each corresponding to detecting an obstacle by a sonar in the near range. There are three other productions corresponding to very close obstacles seen from the front 3 sonars. One last production corresponds to the situation that all sonar values report obstacles far away. We have a total of 11 productions. The possible choice of actions in each production is the 7 actions available. There are an additional 4 productions for the first behavior corresponding to the 4 sonars furthest of 11 front sonars from front, but the behavior produces a no-op, so they are ineffective.

The second behavior implements *wall following*. The robot uses its front 11 sonars spanning 247.5 degrees in front of the robot to detect a wall, align, and move along a wall. It detects a wall from its front 5 sonars, it randomly chooses 10 degrees turn in either direction or goes straight. If it senses the wall from its 6 lateral sonars (3 on either side), it chooses randomly from its 7 possible actions. We have 11 productions for this behavior.

The Q value of all actions in all productions is set to zero at the start. As it is evident, the initial knowledge of these behaviors is negligible, i.e., these behaviors are not optimized for the tasks. The first behavior will receive a reward of 1.0 if the robot does not see an obstacle and -1.0 when it senses an obstacle very close from the front three sonars. The second behavior will receive a 1.0 when it senses a wall from either of its two sonars at 90 degrees from the front. Similar to the first behavior, it receives -1.0 when it senses an obstacle very close in the front three sonars.

We ran two versions of these two robot concurrent behaviors in a simulated room while each behavior is improved with Q learning. Figure 1 shows a trace of a run of this robot in a room.

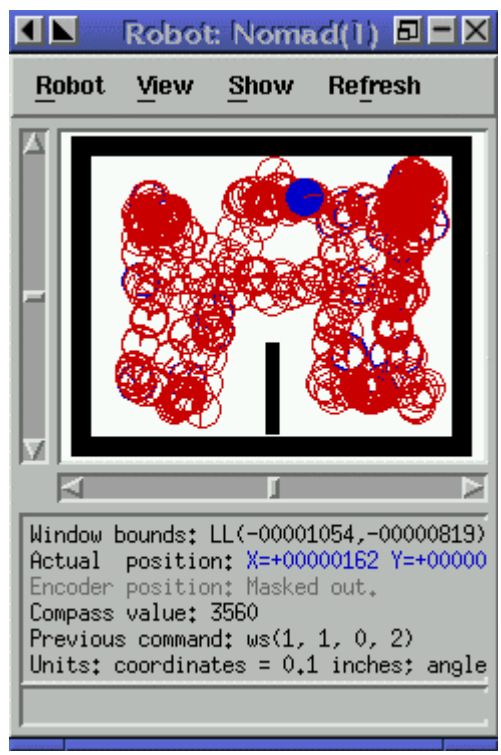

Figure 1. Sample run of concurrent behaviors for 500 cycles. For about 120 cycles, the robot is close to the wall and there is serious conflict between obstacle avoidance and wall following.

First we allowed the behaviors to compete without any explicit suppression between behaviors. Second, we computed the standard deviation in Q

values of each production in each behavior to determine a degree of suppression on the action suggested by the competing behavior. The result is that in the second version, conflict between behaviors was reduced more quickly. The darker traced areas in Figure 1 show that the robot struggled with conflicting behaviors and dwelled in the area. The lighter traced areas in Figure 1 show that the robot has resolved some conflicts (after learning for some time) and dwells less in those areas.

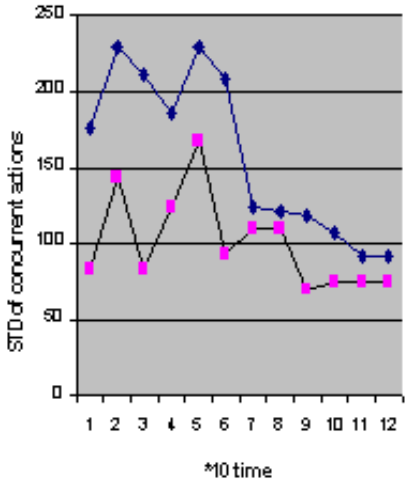

Figure 2. Conflict resolution over time. The higher curve (series 1) belongs to reinforcement learning only. The lower curve (series 2) shows the effect of explicit suppression in addition to reinforcement learning.

Figure 2 shows the standard deviation of turning actions being produced over time. What is shown is 120 cycles of the run where there the robot was near the wall and there was a direct conflict between behaviors. The points on the graph are the sums of values over the last 10 steps. We see that reinforcement learning of behaviors had an overall improvement on conflict resolution. We also see that explicit suppressions enhanced the reduction of conflict resolution. Neither method completely eliminated conflict.

4 Conclusion

This paper presents a direct method of conflict reduction among closely coupled concurrent behaviors. The method uses the Q values developed during Q learning of each concurrent individual behavior. The effect is complimentary to reinforcement-based learning of individual behaviors.

We plan to explore the limits of our technique for conflict resolution among other examples of closely coupled concurrent behaviors.

Acknowledgement

We thank comments made by anonymous reviewers.

References

S. Sen, M. Sekaran, 1998. Individual learning of coordination knowledge, **Journal of Experimental & Theoretical Artificial Intelligence**, 10, pages 333-356, (special issue on Learning in Distributed Artificial Intelligence Systems).

M. Asada, E. Uchibe and K. Hosoda, 1999. Cooperative behavior acquisition for mobile robots in dynamically changing real worlds via vision-based reinforcement learning and development**, Artificial Intelligence**, 110, pp.275-292.

H. Hexmoor, S. Shapiro, 1997. Architecture of a Communicating, Visually Driven Robot Assistant, SUNY Buffalo TR.

M. Kuperstein, 1989. Implementation of a an Adaptive Neural Controller for Sensory-Motor Coronation, **Connectionism Perspectives**, R. Pfeifer, Z. Schreter, F. Fogelman-Soulie, and L. Steels (Eds), pp. 49- 61, Elsevier Science pub.

P. Maes and R. Brooks, 1990. Learning to Coordinate Behaviors, Proceedings of **AAAI-90**: The American Conference on Artificial Intelligence, pp. 796-802 AAAI Press/MIT Press, Boston, MA.

P. Pirjanian, 1999. Behavior Coordination Mechanisms - State-of-the-art, Tech-report IRIS-99-375, Institute or Robotics and Intelligent Systems, School of Engineering, University of Southern California.

H. Pashler, 1997. Doing two things at the same time, **American Scientist**, Vol. 81, No 1, pp. 48-55, Sigma Xi, The Scientific Research Society.

S. Whitehead, 1991. Complexity and Coordination in Q-learning. In Proceedings of 8th International Worksop on **Machine Learning**, pp. 363-367, Evanston, IL., Morgan Kaufman publishers.