Multi-level Control for Animated Autonomous Agents:

Do the Right Thing... Oh, Not That...

Bruce Blumberg, Tinsley Galyean
MIT Media Lab
20 Ames St., E15-305
Cambridge, MA. 02139
bruce|tag@media.mit.edu

1. INTRODUCTION

Imagine making an interactive virtual "Lassie" experience for children. Suppose the autonomous animated character playing Lassie did a fine job as a autonomous dog, but for whatever reason was ignoring the child. Or suppose, you wanted the child to focus on some aspect of the environment which was important to the story, but Lassie was distracting her. In both cases, you would want to be able to provide external control, in realtime, to the autonomous Lassie. For example, by increasing its "motivation to play", it would be more likely to engage in play. By telling it to "go over to that tree and lie down" it might be less distracting.

This paper focuses on how to provide multiple levels of control for autonomous animated characters for use in interactive story systems. Much has been written on the problems of action-selection for autonomous agents and robots [Maes90, Blumberg94, Brooks86]. Some work has focused on the specific problems of building autonomous animated characters [Reynolds87, Tu94]. Less work, however, has focused on the problem of integrating external control into autonomous agent architectures, particularly as it relates to the types of control needed for characters in interactive story systems. It is this problem which is the topic of this paper and much of our research in general.

This paper is organized as follows. We begin by identifying 4 different levels of control and discuss the forms the control may take. We then describe how a particular autonomous agent architecture, developed by Blumberg and Galyean, supports these different levels of control. Lastly, we describe the use of these ideas in 2 systems. Details of the underlying architecture may be found in [Blumberg95]

2. LEVELS OF CONTROL

Maes suggests that there are two dimensions along which one can classify degrees of control. The first is the level of control, from concrete or direct control to more abstract or indirect control. The second dimension is whether the control is prescriptive (i.e. do something), proscriptive (i.e. avoid doing something) and some measure of the importance placed upon the control.

It is proposed that at least 4 levels of control are desirable:

1- At the motor skill level (e.g. "Go-Forward", "Look-At"). Here the director wants the character to engage in specific low level actions. These actions may be "in-character" (e.g. "Walking") or "out-of-character" (e.g. "SetPosition"). Typically, no planning or elaborate sequencing is required by the character, nor is any sensory input used.

2- At the behavioral level (e.g. "Find that juicy steak in the user's hand"). Here the director wants the character to engage in a specific, albeit complicated behavior, and trusts that the character will do the right thing which may require some sort of planning or sequencing of actions by the character. Typically, the character must use sensory input to accomplish the task.

3- At the motivational level (e.g. "You are very hungry"). Here the director is not necessarily directing the character to engage in a specific behavior, but rather is predisposing it to act in that manner, if other conditions are right. For example, if no food is present the character may not necessarily search for food, but the moment it catches a whiff of a steak, it heads directly for it.

4- At the environmental level (e.g. "The Jones town dam just broke", or "There is a juicy steak on the table and your owner is in the kitchen"). Here the director manipulates the behavior of the character by manipulating its environment, thus inducing a potential response from the creature. Imaginary sensory input is a special case of this type of control. Here the director manipulates the sensory input of the character so that the character effectively "imagines" the existence of a significant object or event and acts accordingly. For example, if the director wants the character to approach the user, the director could attach an imaginary "dog biscuit" to the user's hand that is only visible to the dog.

Rather than viewing external control as "directorial dictates", we view it more as a mechanism for expressing "weighted preferences or suggestions" for or against certain behaviors or actions. The higher the weight, the greater the preference's influence on the creature's subsequent behavior. Seen in this light, a preference is just one more factor in

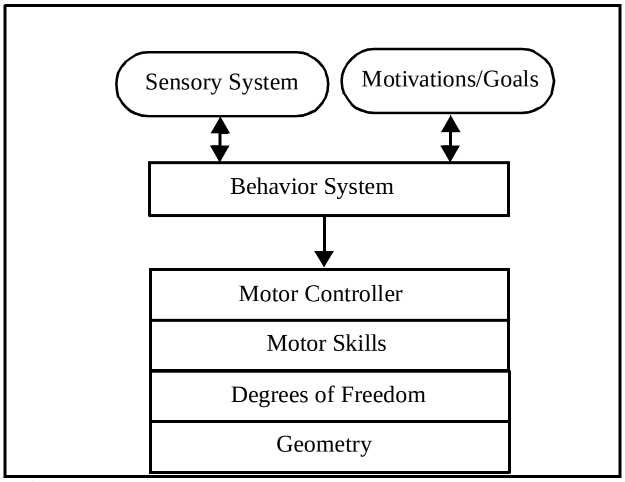

Figure 1: A 5-layered architecture for Animated Autonomous Agents. The lowest level is the geometry (e.g. the shapes, materials and transforms that make up the creature). The Degrees of Freedom (i.e. neck joint) represent the "knobs" which are available to modify the geometry so as to produce motion. Motor Skills use one or more DOFs to produce coordinated motion (e.g. Walk or Wag-Tail). The Controller maps abstract commands (e.g. "forward") into calls to the Creature's Motor Skills (e.g. "walk"). The purpose of the Behavior System is to weigh the potentially competing motivations of the creature, assess the state of its environment, and issue to the Controller, the set of actions which makes the "most sense" at any given instant in time.

the mix of internal motivations and external stimulus which ultimately determine in which behaviors the creature will engage.

Preferences may be proscriptive, for example, "I don't care what you do as long as you don't eat". This is different than (2) or (3) in that it provides a different type of control. For example, it would be difficult using just (2) or (3) to implement a "I don't care what you do as long as you don't...", since even reducing the motivational level associated with a behavior may not assure that the behavior will not be executed if the behavior has a strong sensory component. Moreover, it would also allow the director to achieve the same effect as that of temporarily changing a motivational variable without, perhaps, the attendant problems (e.g. restoring the value when done, or insuring that changing the value does not have an unintended side-effect).

Weighted preferences may also be used at the direct motor level. For example, it is important that the director be able to make recommendations for or against certain motor actions so as to insure that the character satisfies certain spatial or behavioral constraints. These recommendations can used by the low-level behaviors to arrive at compromise solutions which help satisfy the director's constraints as well as those of the behavior. This is different than (1) above in that a preference is used in the context of a running behavior, whereas (1) is an explicit command to execute an action. For example, if the director does not want the character to turn left because this would move the character toward the center of the scene, it can make a recommendation against turning left. If the character is currently searching for food, it may be indifferent to turning left or going straight in the absence of a recommendation. However, in the presence of a recommendation against turning left, the behavior will avoid turning left.

Thus, an architecture for autonomous animated creatures needs to support these two dimensions of control. That is, it should provide support for control at varying degrees of abstraction, and it should also include a mechanism whereby the control can be specified as weighted preferences for and against actions and behaviors.

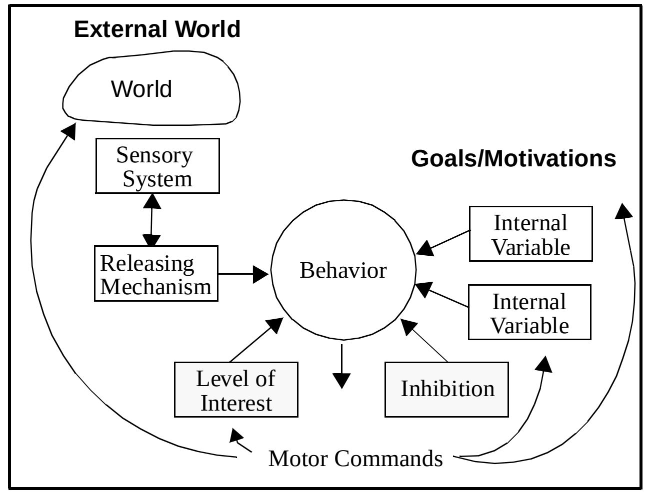

Figure 2:The purpose of a Behavior is to evaluate the appropriateness of the behavior, given external stimulus and internal motivations, and if appropriate issue motor commands. Releasing Mechanisms act as filters or detectors which identify significant objects or events from sensory input, and which output a value which corresponds to the strength of the sensory input. Internal Motivations or goals are represented via Internal Variables which output values which represents the strength of the motivation. A behavior combines the values of the Releasing Mechanisms and Internal Variables on which it depends and that represents the value of the Behavior before Level of Interest and Inhibition from other Behaviors. Level of Interest is used to model boredom or behavior-specific fatigue. Behaviors must compete with other behaviors for control of the creature, and do so using Inhibition.There are a variety of explicit and implicit feedback mechanisms.

Blumberg and Galyean have proposed, and implemented just such an architecture for animated autonomous creatures. This architecture is summarized in Figure 1. The remainder of this paper highlights how external control has been integrated into this architecture.

3. ARCHITECTURE

The two most relevant parts of the architecture for the purposes of this discussion are the Behavior System and the Motor Controller. These are briefly introduced below.

The purpose of the Behavior System (the top level in Figure 1) is to send the "right" set of control signals to the motor system at every time-step. It must weigh the potentially competing goals or needs of the creature, assess the state of its environment, and choose the set of actions which make the "most sense" at that instant in time. More generally, it provides the agent with a range of high-level behaviors of which it is capable of performing competently and autonomously in a potentially unpredictable environment. Indeed, it is this ability to perform competently in the absence of external control which makes high level control (motivational or behavioral) possible. Otherwise, an external entity would have to provide time-step to time-step motor control in order to assure that a given goal was achieved, or task accomplished.

While the Behavior System is shown in Figure 1 as a monolithic entity, it is in fact composed of a loosely hierarchical network of "self-interested, goal-directed entities" called Behaviors each of which is fighting for control of the creature. Figure 2 summarizes the structure of an individual

Behavior. The granularity of a Behavior's goal may vary from very general (e.g. "reduce hunger") to very specific (e.g. "chew food"). General Behaviors (e.g "reduce hunger") typically depend on several more specific behaviors to accomplish their objective (e.g. "find food", "sniff", "chew"), which in turn may depend on even more specific Behaviors. Thus, it is often useful to view the Behaviors as being organized in a loose hierarchy as is shown in Figure 3. Branches of the tree often correspond to sub-systems which perform a specific task-level function. Details of the actual implementation may be found in [Blumberg 95], and the ethological justification for the model may be found in [Blumberg 94].

A Behavior does not itself perform any actions. Rather it issues commands to the Motor Controller, and the Motor Controller dispatches a Motor Skill which performs the appropriate action. For example, the "move to" Behavior issues the "forward" command to the Motor Controller, which in turn invokes the "walk" Motor Skill. In fact, the Motor Controller is implemented in such a way that the commands may take one of three imperative forms:

- As a primary command. Primary commands are executed immediately, as long as the required resources are available for it. That is, the degrees of freedom or "knobs" (see Figure 1) needed by the underlying Motor Skill must be available in order for the command to succeed.
- As a secondary command. Secondary commands are intended to be used to specify desirable but non-essential actions. In other words "do it if you have nothing better to do." Secondary commands are queued by the Motor Controller and executed after primary commands. A priority for a secondary command is also specified when it is issued. This priority dictates the order in which they are executed, thereby giving high priority commands the opportunity to grab the resources (DOFs) before lower priority ones.
- As a meta-command. Meta commands are recommendations as to how a particular action should be performed. For example, a meta command might suggest that "if you are going forward, use this gait." Like secondary commands, meta-commands are queued by the Motor Controller. Behaviors may query the controller to see if a given meta-command has been issued, and if so, it may retrieve the command and use it as it sees fit. Unlike secondary commands, meta-commands only have an effect if a Behavior subsequently makes use of the recommendation to affect what commands it issues.

These multiple imperative forms for motor commands are used extensively by the Behavior System. The action-selection algorithm used by the Behavior System implements a winner-take-all arbitration scheme in which there is a single Behavior which is in control at any given instant. This winning Behavior issues primary commands, since it is the most appropriate Behavior to run at that instant. However, before it runs (i.e. issue motor commands), losing Behaviors may issue secondary or meta-commands. For example, the Dog may have a low-priority behavior whose sole function is to alter the dog's characteristics to demonstrate to the user how the dog is feeling. In this case, the behavior may issue secondary commands for ear, mouth, and tail position as well as body posture, and meta-commands for gait. The reason for using secondary commands is that these are desirable actions, but not essential ones. Similarly, this behavior may not know whether the dog should go forward or not, but it may be in a position to offer a suggestion as to how the dog should move forward should some other behavior decide that the dog should move.

4. INTEGRATION OF EXTERNAL CONTROL

Motivational and behavioral control (control levels 2 and 3 above) is accomplished by adjusting the factors which influence the relative strength of a Behavior, or group of Behaviors. This may be done in a number of ways.

First, motivational control is provided via named access to the Internal Variables which represent the motivations or goals of the Behavior System. By adjusting the value of a given motivational variable at runtime, one can make the creature more or less likely to engage in Behaviors which depends on that variable.

Second, the constituent parts of a Behavior are also accessible, and this provides another mechanism for exerting behavioral control. Releasing Mechanisms (see Figure 2) act as "object-of-interest" detectors for a Behavior. By modifying the kind of "object-of-interest" to which a given Releasing Mechanism is sensitive, one is in effect modifying the conditions under which a given Behavior will become active. For example, the "marking" Behavior of a dog may rely on a Releasing Mechanism which triggers on "fire hydrants". By modifying it at runtime so that it triggers on "user's pants leg", the resulting behavior of the dog will be very different. One can also adjust the "strength" of a Releasing Mechanism so as to raise or lower its effect on the value of the Behavior (and thus the likelihood of the Behavior becoming active when the Releasing Mechanism is triggered).

Third, the Behavior System is structured so that action-selection (i.e. the process of choosing which Behavior should be active at any given instant), can be initiated at any node in the system. This allows an external entity to force execution of a particular part or branch of the Behavior System, regardless of motivational and sensory factors which might otherwise favor execution of other parts of it. Since branches often correspond to task-level collections of Behaviors, this provides a way of achieving task-level control.

Since each creature has its own sensory system, it is very straightforward to provide "imaginary" sensory input. For example, objects may be added to the world which are only visible only to a specific creature. The presence of these objects, however, may trigger certain behaviors on the part of the creature. It should be noted that this may be done more directly by manipulating the creature's Releasing Mechanisms. However, the advantage of this technique is that it does not require the external entity to know anything about the internal structure of the Behavior System.

The mechanisms described above for controlling the Behavior System naturally support Maes' second dimension of control. For example, by adjusting the level of a motivational variable which drives a given branch of the Behavior System, the director is expressing a weighted preference for or against the execution of that behavior or group of behaviors.

The multiple imperative forms supported by the Motor Controller allows the director to express weighted preferences directly at the motor level. For example, at one extreme, the external entity may at any time "shut off" the Behavior system and issue motor commands directly to the creature. Alternatively, the Behavior system can be running, and the external entity may issue persistent secondary or meta-commands which have the effect of modifying or augmenting the output of the Behavior system. For example, the external entity might issue a persistent secondary command to "wag tail". Unless this was explicitly over-ruled by a Behavior in the Behavior system, this would result in the Dog wagging its tail. Or, for example, the external entity might suggest, via a meta-command, that if the dog chooses to move, it should move with a "trot", or "use any gait but a trot". Meta-commands may also take the form of spatial potential field maps which can be combined with potential field maps generated from sensory data to effectively attract or repel the creature from parts of its environment.

5. EXAMPLES

These ideas have been implemented as part of a project to develop a general architecture for building and controlling artificial autonomous characters. We have developed several creatures using a tool kit which implements this approach and these creatures have been used in several different applications. For the Alive project, we have developed "Silas T. Dog" an autonomous animated dog which interacts with a user in a 3D virtual world in a believable manner. Silas responds to a dozen or so gestures and postures of the user, and responds appropriately (e.g. if the user bends over and holds out her hand, Silas moves toward the outstretched hand and eventually sits and shakes his paw). A good deal of attention has been paid to conveying the dog's internal state to the user. The dog always looks at its current object of interest (head, hand, etc.), and when it is sad or happy, its tail, ears and head move appropriately. As mentioned earlier, the behavior system makes extensive use of the different imperative forms supported by the motor system.

We have also developed a number of creatures which are used in the context of an interactive story system developed by Tinsley Galyean of the Media Lab. Galyean's system features a computational director which provides "direction" to the creatures so as to meet the requirements of the story. For example, at the beginning of the story, a dog hops out of a car and wanders around. If the user, who is wearing a head-mounted display, does not pay attention to the dog, the director will send the dog over to the user. If the user still does not pay attention, the director effectively tells the dog: "the user's leg is a fine replacement for a hydrant, and you really have to...". The resulting behavior on the part of the dog usually captures the user's attention.

6. CONCLUSION

Autonomous animated characters can play an important role in Interactive Story Systems by freeing the director from the details of time-step to time-step control. However, to be truly useful, the director must be able to control the character at a number of different levels of abstraction, and levels of influence. We have presented a system which shows how these ideas may be integrated into an architecture for autonomous animated characters.

REFERENCES

Blumberg, B., 1994, Action-Selection in Hamsterdam: Lessons from Ethology. in Third International Conference on the Simulation of Adaptive Behavior. 1994. Brighton.

Blumberg, B. & Galyean, T., 1995. Multi-Level Direction of Autonomous Creatures for Real-Time Virtual Environments, submitted for publication, Siggraph '95.

Brooks R., 1986, A Robust Layered Control System for a Mobile Robot, IEEE Journal of Robotics and Automation, RA-2, April 1986.

Maes P., 1990, Situated Agents Can Have Goals, In: Designing Autonomous Agents: Theory and Practice from Biology to Engineering and Back, Edited by P. Maes, MIT Press, Cambridge.

Maes, P., 1993, Modeling Adaptive Autonomous Agents, to appear in The Artificial Life Journal, Vol. 1, Numbers 1&2, Summer 1994.

Maes, P., 1994, Artificial Life meets Entertainment: Interacting with Lifelike Autonomous Agents, submitted to Communications of the ACM Special Issue on Novel Applications of A.I., March 95

Reynolds, C.W. 1987, Flocks, Herds, and Schools: A Distributed Behavioral Model. in ACM Siggraph. 1987. 25-34

Tu, X. and Terzopoulos, D. 1994,Artificial Fishes: Physics, Locomotion, Perception, Behavior. in ACM Siggraph94. 1994. Orlando, FL