In "Biologically Inspired Intelligent Robots", Y. Bar-Cohen and C. Breazeal, eds., SPIE Press, Vol. PM122, May 2003

Chapter 8 - Biomimetic Robot Control

Dan Paluska, MIT, 200 Technology Sq., room 006, Cambridge MA 02139,
617-253-2475 ext 109 leinad@ai.mit.edu
Maja Mataric, University of Southern California, 941 West 37th Place, SAL 228, Mail Code 0781, Los Angeles, CA 90089-0781, (213) 740-4520 mataric@cs.usc.edu

Robert Ambrose, NASA Johnson Space Center, Metrica Automation and Robotics, Mail Code ER4, Houston, Texas 77058, (281) 244-5561 rambrose@ems.jsc.nasa.gov

and

Jerry Pratt, Institute for Human and Machine Cognition, 40 South Alcaniz Street, Pensacola, FL 32501 (850) 202-4481, jpratt@yobotics.com

8.1 Architectures for Robot Control

Robot control architecture provides structure and principles for designing a variety of specific robot controllers. In general, while there are infinitely many possible controllers, all fit into one of the following basic types of control architectures:

Reactive
Deliberative
Hybrid

Behavior-BasedInterestingly, all of the above architecture types are inspired by biological systems, ranging from insect locomotion to human cognition. We introduce and describe each in turn, then focus on behavior-based control and its biomimetic properties.

8.1.1 Reactive Control

An appropriate slogan for reactive control is: "Don't think, react!" Reactive control is directly inspired by the tight coupling between sensory inputs and motor outputs in biological systems, which enable animals to respond to a variety of changing conditions, threats, and opportunities. Similar capabilities are crucial for robots, even for mundane tasks such as navigating around corridors in a populated building. In general, the ability to respond effectively to very quickly to changing and unstructured environments is termed "reactivity", and it typically implies an underlying reactive controller. In order to react quickly, however, a robot cannot take the time to do complex processing. Consequently, reactive systems are defined by a lack of internal models and representations, and subsequently a lack of ability to learn or change over time. They are capable of maintaining short-term state, which can allow them to avoid repetitive actions and cycles (e.g., getting stuck in corners), but such "ephemeral" state is purposefully lightweight in terms of processing, so as to not slow down the response time of the system.

The tradeoff is made in favor of fast reaction time and against complexity of reasoning. Formal analysis has shown that for environments and tasks that can be characterized, reactive controllers can be shown to be highly powerful, and if properly structured, capable of optimal performance in particular classes of problems (Schoppers 1987, Agre & Chapman 1990). Reactive control is often described as its biological equivalent: "stimulus-response". This is a powerful control method: many animals, after all, are largely reactive.

Some of the most prominent architectures for reactive control include the Subsumption Architecture (Brooks 1986, Brooks 1991) and situated automata [Rosenschein & Kaelbling 1986]. In the current state of the art, all robots situated in the physical world contain a reactive controller as a part of the control system, thus ensuring the robot's ability to survive in the environment, as a necessary foundation for performing any of its higher-level tasks.

Because reactive control draws its fundamental inspiration from biological systems, in particular fast reaction times in simple animals like insects, it is fundamentally biomimetic. Most reactive controllers do not mimic specific natural systems, or their structure, but instead their functionality, using a similar philosophy about reactivity and adaptivity.

8.1.2 Deliberative Control

An appropriate slogan for deliberative control is: "Think hard, then act." While reactive control grew out of feedback control, and is focused on action, deliberative control grew out of work in

Artificial Intelligence, and is focused on cognition. In this type of control, the emphasis is on knowledge, not reaction. Thus, deliberative systems employ rich internal representations and complex reasoning engines to derive optimal plans for action. Because deliberative systems were developed without concern for real-time response, they are computationally heavy, and thus typically not capable of reasoning about real-world situations requiring immediate answers. When there is sufficient time and sufficient information to maintain an accurate world model, deliberative control allows the robot to act strategically, selecting the best course of action for a given situation. However, being situated in a noisy, dynamic world usually makes this impossible. Thus, real-world robots are almost never purely deliberative. Interestingly, early AI was based on models of human cognition, so deliberative systems grew out of those models. The abilities to plan and reason that are fundamental to deliberative control are also considered by some to be the distinguishing properties of human cognition. Consequently, deliberative control is, in a sense, biomimetic.

8.1.3 Hybrid Control

An appropriate slogan for hybrid control is: "Think and act independently, in parallel." As it inevitably happens, once robot control matured, it combined the best aspects of the existing methodologies in pursuit of improved functionality. Thus, hybrid control combines the best aspects of reactive and deliberative control: it attempts to combine the real-time response of reactivity with the rationality and optimality of deliberation. As a consequence, hybrid control system contains at least two fundamentally different components (thus the name), the reactive and the deliberative ones, which must interact in order to produce a coherent system-level output. This is challenging, because the reactive component deals with the robot's immediate needs and responses, such as avoiding obstacles, and thus operates on a very fast time-scale and uses direct external sensory data and signals. In contrast, the deliberative component uses highly abstracted, symbolic internal representations of the world, and operates on them on a longer time-scale.

As long as the outputs of the reactive and deliberative components of a hybrid system are not in conflict, the system requires no further coordination. However, the two parts of the system must interact if they are to benefit from each other. The reactive system must override the deliberative one if the world presents an unexpected and immediate challenge. Analogously, the deliberative component must inform the reactive one in order to guide the robot toward more efficient and optimal plans of action. The interaction of the two such different parts of the system requires an intermediate component, whose construction is typically the greatest challenge of hybrid system design. As a result, hybrid systems are often called "three layer systems", consisting of the reactive, intermediate, and deliberative layers. A great deal of research has been performed on the approaches to designing these components and their interactions (Giralt et al. 1983, Firby 1987, Arkin 1989, Malcolm & Smithers 1990, Connell 1991, and Gat 1992).

Hybrid systems are considered by their proponents to be the architecture of human intelligence, combining the evolutionarily older and lower-level reactive capabilities with the newer, higher-level reasoning and linguistic ones. The proponents of the architecture thus consider it not only efficient for robots but also biomimetic.

8.1.4 Behavior-Based Control

Behavior-based control emerged out of reactive control, in response to and as an alternative to hybrid control, with a specific goal of being more directly biomimetic. The methodology is based on a biologically-inspired philosophy that favors parallel, decentralized processing modules, since natural systems are believed to be similarly organized, starting with spinal reflex movements (Bizzi et al. 1991), up to more complex behaviors such as flocking and foraging (Mataric, 1995). The approach is general and fits well within other biomimetic control frameworks such as schema theory (Arbib 1981).

In behavior-based systems, the robot controller consists of a collection of ``behaviors'', each of which achieves and/or maintains a specific goal. For example, the ``avoid-obstacles'' behavior maintains the goal of preventing collisions with objects in the environment, and the ``go-home'' behavior achieves the goal of reaching some home region. Each behavior is a processing element or a procedure or a control law, which can be implemented either in software or hardware; each can take inputs from the robot's sensors (e.g., camera, ultrasound, infra-red, tactile) and/or from other behaviors, and send outputs to the robot's effectors (e.g., wheels, grippers, arms, speech) and/or to other behaviors in the system. Consequently, a behavior-based robot is controlled by a structured network of interacting behaviors.

The organizational methodology of behavior-based systems differs from the other control methods in its approach to modularity, i.e., the way in which the system is organized and subdivided. The behavior-based philosophy mandates that the behaviors be relatively simple, incrementally added to the system, and not executed in a serial fashion. These systems are meant to be constructed in a bottom-up fashion resembling evolution in its incremental refinement as well as its utilitarian exploitation of existing modules.

Behaviors are activated in response to external and/or internal conditions, i.e., sensory inputs and internal state. The system as a whole activates entire subsets of behaviors so that parallelism can be exploited, both in speed of computation and in the resulting dynamics. The latter is a critical aspect of behavior-based control: as multiple behaviors or modules are active, dynamics of interaction arise both within the system itself (from the interaction among the behaviors) and within the environment (from the interaction of the behaviors with the external world). Inspired by biological organisms, designers of behavior-based systems exploit these dynamics to create (by hand or automatically through the use of learning) repeatable, stable, and, ultimately, intelligent behavior without relying on top-down, centralized, and often even hierarchical control (Agre & Chapman 1987, Brooks 1991).

8.1.5 Design and Coordination of Behaviors

A methodological constraint of behavior-based systems is their use of state and representation: information is not centralized or centrally manipulated. Instead, various forms of distributed representations are used, ranging from static table structures and networks, to active, procedural processes, providing a rich medium for innovative interpretations.

Behaviors can be designed at a variety of levels of abstraction. In general, they are made to be higher than the robot's atomic actions (i.e., typically above “go-forward-by-a-small-increment”, “turn-by-a-small-angle”), and they extend in time and space. This effectively elevates the representational level of the system, which has been shown to facilitate higher-level cognition and learning (Mataric, 1990, and Mataric 1996). Some commonly implemented behaviors include: ``go-home", ``find-object", ``get-recharged", ``avoid-collisions", ``pick-up-object". More specialized behaviors include: ``avoid-the-light", ``aggregate-with-group", ``find-mate", ``follow-edge", etc.

The internal behavior structure of a system need not necessarily mirror its externally manifested behavior. For example, a robot that flocks with other robots may not have a specific internal ``flocking" behavior; instead, its interaction with the environment and other robots may produce flocking. Behavior-based systems are typically designed so the effects of the behaviors interact in the environment rather than internally through the system, so as to take advantage of the richness of the interaction dynamics. These dynamics are sometimes called ``emergent" because they result from the interactions and are not internally specified by the robot's program (Steels 1994).

A key issue in behavior-based systems concerns the coordination of the multiple behaviors, thus making "arbitration", i.e., deciding what behavior to execute at each point in time, one of the central challenges. For the sake of simplicity, most implemented systems use a built-in, fixed priority ordering of behaviors. More flexible solutions, which can be less computationally efficient and harder to analyze, have been suggested, commonly based on selecting a behavior by computing some function of the behavior activation levels, such as voting or activation spreading (Maes 1989, and Payton et al. 1992).

8.1.6 A Historical Overview

Behavior-based systems were founded on the work in reactive robotics and in particular on the Subsumption Architecture (Brooks 1986), which achieves rapid real-time responses by embedding the robot's controller into a collection of preprogrammed parallel condition-action rules, or reflexes, with minimal internal state (e.g., ``if bumped, stop") (Brooks & Connell 1986, and Agre & Chapman 1987). In contrast to these so-called bottom-up systems, traditional AI deliberative planner-based systems are top-down, and require the robot to perform a serial sequence of processing sense-plan-act steps (e.g., ``combine the sensory data into a model of the world, then use the planner to find a path in the model, then send each of the steps of the plan to the robot's wheels") (Giralt et al. 1983, Moravec & Elfes 1985, Laird & Rosenbloom 1990). Hybrid systems attempt a compromise between the ``thinking" and ``acting" extremes by employing a reactive system for low-level control and a planner for higher-level decision-making (Firby 1987, Georgeff & Lansky 1987, Arkin 1989, Payton 1990, Malcolm & Smithers 1990, and Connell 1991). Hybrid systems tend to separate the control system into two or more communicating but largely independent parts. Behavior-based systems are an alternative to hybrid systems; they enable fast real-time responses through simple reactive behaviors that directly link sensors and effectors, but also provide for higher-level deliberation, by distributing the representation and computation over more sophisticated concurrent behavior processes. The power, elegance, and complexity of behavior-based systems all stem from the ways in which their constituent behaviors are defined and used. Some proponents of behavior-based systems claim that they better model cognition, while others employ them purely from pragmatic motivations, including their ease of system development and the robustness of the results.

8.1.7 Adaptation and Learning

In robotics, adaptation usually refers to the ability of the system to function within and adapt to a range of conditions. Learning, on the other hand, is considered a more general capability, enabling the system to improve its behavior over time, adapting to potentially larger changes in conditions.

Adaptation and learning are challenging in robotics, because sensing and acting in the physical world involves a great deal of uncertainty due to incomplete and noisy information and dynamically changing environment conditions. It is often difficult for robots to correctly perceive (due to limited sensory technology) and act on (due to limited effectors) the variety of situations that arise in the physical world. Nonetheless, robot learning is an active branch of robotics, and is one of the variations and adaptations of standard machine learning techniques, and in particular reinforcement learning, that have been effectively applied to robots. In particular, behavior-based robots have been demonstrated to learn to walk (Maes & Brooks 1990), navigate (Mataric 1992, Millan 1994, and Pfeifer & Verschure 1992), communicate (Yanco & Stein 1993), divide tasks (1993), behave socially (Mataric 1994), and even identify opponents and score goals in robot soccer (Asada et al. 1994).

Reinforcement learning is the most popular method for learning in mobile robotics (Mahadevan & Kaelbling 1996, and Kaelbling et al. 1996). It refers to a set of problems (rather than methods), in which the robot must improve its behavior based on rewards or punishment from the environment. The reinforcement learning model is based on early conditioning work in psychology, and recently an increasing number of robot learning systems have employed related concepts from biological reinforcement learning, most notably shaping (Mataric 1994, Mataric 1996, and Dorigo & Colombetti 1997) and operant conditioning (Gaudiano et al. 1996, and Touretzky & Saksida 1997). Supervised learning methods using neural networks have also been used extensively (Baloch & Waxman 1991, Pfeifer & Verschure 1992, and Gaudiano et al. 1997). Some of the most effective demonstrations of learning in mobile robots have been inspired by biological learning systems.

8.1.8 Demonstrations and Applications

Interestingly, as the behavior-based approach is being explored for modeling natural systems, the resulting research is demonstrating methods with immediate practical applications. Behavior-based robots have demonstrated various standard robotic capabilities, including obstacle avoidance, navigation, terrain mapping, following, chasing/pursuit, object manipulation, task division and cooperation, and learning maps, and walking. Application domains have included mobile robots, underwater vehicles, space robotics (most recently Sojourner, the robot that autonomously explored the surface of Mars), as well as robots capable of manipulation, grasping, walking, and running. Models of large-scale group behavior developed with behavior-based systems appear to be the methodology of choice for deploying robots in hazardous or inaccessible environments, including under water, in mine fields, under ground, and in space. These domains require a combination of individual independence and group cohesion capable of adapting to varying group sizes and organizations, as it does in natural societies. Consequently, behavior-based approaches have presented popular options for addressing analysis of natural behavior and synthesis of practical artificial behavior.

Faithful simulation and robotic models have been developed of a variety of natural behaviors, ranging from reflexive behavior selection strategies (Connell 1990), cricket phonotaxis for flight and mating behaviors (Webb 1994), lobster odor location (Grassoetal 1996), fly (Franceschini 1975) and hover-fly (Cliff 1990) vision, to insect navigation, trail formation, and path-finding (Deneubourg et al. 1986, Deneubourg et al. 1987), the application of the schema theory to modeling navigation [Arbib 1997] and frog behavior (Arbib 1989), the use of evolutionary computation methods, modeled after natural selection, to develop individual robotic behaviors (Nolfi et al. 1994) as well as group behaviors (Agah & Bekey 1997) and many others. A bi-annual international conference, ``Simulation of Adaptive Behavior'' is devoted to the subject, and has been convening since 1990; its proceedings provide a representative review of the various activities in this field, as does the associated journal, "Adaptive Behavior."

8.1.9 Example of Biologically-Inspired Navigation

Our own work has used behavior-based systems to model navigation, map learning, and path finding mechanisms loosely modeled on the rat's hippocampal place cells (Mataric 1990). In this system, the behaviors served not only for general movement and obstacle avoidance, but also for landmark detection and representation. As new landmarks were discovered, behaviors became associated with them, and subsequently became activated whenever the robot returned to the same location, much like hippocampal place cells in rats (Okeefe & Nadel 1978, Eichenbaum & Cohen 1988, Foster et al. 1989, and Okeefe 1989). Unlike its neural counterpart, whose network topology has no obvious mapping to the physical space it represents, our synthetic navigation system maintained a clear isomorphic mapping between the two. Consequently, the resulting robot-generated maps were easily readable by humans interacting with the robot. The landmark behaviors also served as predictors that allowed the robot to localize more precisely in its environment; an active landmark used context to activate its network neighbor in the robot's direction of travel, thus ``priming'' it, and generating expectation. A lack of expectation indicated novel locations or incorrect localization.

8.1.10 An Example of Ethologically-Inspired Group Behavior

In our own work, we have focused on applying the principles of behavior organization to high-dimensional behavior-based systems such as those involved in the control of groups of interacting robots and of humanoid robots (see below). In both problem domains, we have used basis behaviors, or primitives, in order to structure and simplify the control problem, as well as to enable adaptation and learning. ***Basis behaviors*** are a small set of necessary and sufficient behaviors that could be composed (by sequencing or superposition), as a means of handling controller complexity and simplifying robot programming. Basis behaviors are the {\it primitives} that serve as a substrate for control, representation, and learning in our behavior-based systems. The notion of basis behaviors comes from neuroscience of articulated limb control, as discussed in the next section. We initially adapted the notion to mobile robot and robot team control, and have subsequently applied it to humanoids, as described below.

Inspired by ethologically common natural behaviors, we used a basis set consisting of avoidance, following, homing, aggregation, and dispersion to demonstrate higher-level group behaviors including flocking, foraging/collection, and herding (Mataric 1995). We also demonstrated how, given such a basis behavior set, learning algorithms could be applied to improve behavior selection over time (Mataric 2001, 1997) as well as to learn of models of interaction(Goldberg &

Mataric 2001, 2000, 1999), resulting in task division, specialization, and dominance hierarchy formation. More recently, we have begun work on more tightly-coupled cooperation tasks such as construction (Jones & Mataric 2002). Finally, we have used a metaphor from economics, based on auctions, to more principally describe multi-robot coordination and communication (Gerkey & Mataric 2000, Gerkey & Mataric 2002).

These robotic implementations resemble the equivalent behaviors found in species ranging from ants, crabs, and chickens to chimps and humans (Chase & Rohwer 1987, Beer 1990, Mataric 1993), but are not designed to be careful mechanistic models. Instead, they serve as demonstrations of possible mechanisms that push the state of the art in robotics, as well as allow us to postulate theories about their inspirations from nature.

8.1.11 Summary

All of the existing robot control architectures are, to varying degrees, inspired by biological systems. Proponents of each consider their methodology of choice to be biomimetic. Here we have focused on work in behavior-based systems, which use functional modeling, i.e., the synthesis of life-like and/or biologically-inspired behavior that is robust, repeatable, and adaptive. Inspiration from cognitive science, neuroscience, and biology drives the development of new methods and models in behavior-based robotics, and the results tie together several related fields, including Artificial Life, evolutionary computation, and multi-agent systems. Ideas from Artificial Intelligence and engineering continue to be actively explored and applied to behavior-based robots as their role in animal modeling and practical applications is being developed.

8.2 Humanoid Upper Body Control

8.2.1 Motivation

Humanoid robotics is a rapidly growing branch in robotics, spurred by the recent availability of commercial humanoid robots (e.g., Honda, Sony, etc.) as well as growing number of research test-beds and innovative applications (e.g., NASA's Robonaut). The challenge of humanoid control presents a leap in complexity, as it combines both locomotion and articulation, and involves an unprecedented number of degrees of freedom (DOF) to be simultaneously controlled. The more complex the system to be controlled, the more necessary it is to modularize the controller in order to make it robust and efficient. Not surprisingly, most research on humanoid control draws inspiration and models from biology, in particular from neuroscience of motor control and coordination (e.g., [Cog,Vanderbilt,WASEDA,Stefan]). In this section, we describe a biomimetic approach to humanoid upper body control, employing some of the same principles as described in the previous section, specifically behavior-based control and the notion of basis behaviors or primitives, and its neuroscience origins.

8.2.2 Approach

Inspired by neuroscience theories of motor control (e.g., Bizzi et al. (1991), and Mussaivaldi & Giszter (1992)), which demonstrate evidence of a finite set of additive force fields controlling the movement repertoire of frogs and possibly mammals, we are developing behaviors for the control of articulated humanoid movement. As with our work on coordination of group behavior, we are using ``basis behavior'' primitives as a substrate for a broader repertoire of higher-level humanoid upper body behaviors, obtained by sequencing and combining the basis set (Mataric 1995). Our basis set includes behaviors for discrete movement-to-point, posture maintenance, and oscillatory movements, all based on theories of human motor control (Flash & Hogan 1985, Morasso 1981, Arbib et al. 1995, and Brooks et al. 1995). We have also developed a technique for automatically deriving primitive behaviors from human movement data, by applying dimensionality reduction techniques (Jenkins & Mataric 2002).

Besides the principle of behavior primitives, a central notion of our approach to humanoid control and learning is the use of imitation. This capability allows robots to be programmed and interacted with through human demonstration, a natural human-humanoid interface. Human ability to imitate, i.e., to observe behaviors performed by a teacher and then repeat them, is a powerful but still poorly understood form of learning. The fundamental open problems in imitation are: 1) interpreting and understanding of the observed behavior, and 2) integrating the visual perception and movement control systems to reconstruct the demonstration. Consequently, the goal of our biomimetic research is similarly two-fold; we are developing methods for segmenting and classifying visual input for recognition of human behavior and, at the same, methods for structuring the motor control system in a way that makes it capable of general movement and imitation learning. Our approach brings these two pursuits together much in the same was as the evolutionary process brought them together in biological systems (Mataric 2000, Mataric 2001, and Rizzolatti et al. 1996). Our general approach is to structure the motor system into a collection of movement primitives (i.e., basis behaviors), which are then used to both generate the humanoid's movement repertoire and to provide prediction and classification capabilities for visual perception and interpretation observed of movement. In this way, what the humanoid is capable of *doing* helps it understand what it is *seeing*, and vice versa. The more it sees, the more it learns to do, and thus the better it understands what it sees for further learning; this is the imitation process.

Acquiring new skills by imitation is a well-known robotics problem. It is usually classified as learning by demonstration, where the robot uses vision sensors to interpret the behavior of a human user and thus acquire a new task. Historically, assembly tasks have been learned by demonstration, without an effort to precisely model the observed behavior but focusing on achieving the demonstrated goals (Ikeuchi et al. 1990, Kuniyoshi et al. 1994, Lee & Xu 1996, and Hovland et al. 1996). More recently, imitation learning between two mobile robots has been demonstrated (Demiris & Hayes 1996), as has skill learning between a human demonstrator and an articulated robot arm with human kinematics, such as learning to balance a pole (Schaal 1997) and play Kendama (Miyamoto et al. 1996). The latter was an instantiation of the bi-directional theory of motor control (Kawato 1996), another example of a robotic implementation of a neuroscience model. Modeling human skill acquisition, while a tremendously challenging task, is gaining popularity in robotics. It has recently been approached from a Piagetian perspective, using developmental stages in order to simplify the complex learning problem (Brooks & Stein 1994, Marjanovic et al. 1996, and Williamson 1996). The recent wave of interest in humanoid control and imitation has been largely biomimetic, in contrast to early work. Next we describe the neuroscience foundations for our approach.

8.2.3 Neuroscience Inspiration

Evidence from neuroscience studies in animals points to two neural structures we find of key relevance: spinal fields and mirror neurons. Spinal fields, found in frogs and rats so far, code for complete primitive movements (or behaviors), such as reaching and wiping (Bizzi et al. 1991). More interestingly, they are additive; when multiple fields are stimulated, the resulting movement is a meaningful combination. Since the spine codes a finite number of such fields, they represent a basis set of primitives, and were precisely the inspiration for our work on basis behaviors, described above.

Neurons with so-called ``mirror'' properties were recently found in monkeys and humans. They appear to directly connect the visual and motor control systems by mapping observed behaviors, such as reaching and grasping, to motor structures that produce them (Rizzolatti et al. 1996). It is not yet known how many such movements are directly mapped by the mirror system, but the basic idea serves as rich inspiration for structuring a robotic imitation system.

We combine these two lines of evidence, spinal basis fields and mirror neurons, into a more sophisticated notion of behaviors, or ***perceptual-motor primitives***. These allow a complex system, such as a humanoid, to recognize, reproduce, and learn motor skills. As mentioned above, the primitives are used as the basis set for generating movements, but also as a ``vocabulary'' for classifying observed movements into executable ones. Thus, primitives can classify, predict, and act.

Our approach is an extension of our earlier hippocampus-inspired navigation work, described above, in which landmark behaviors primed and anticipated other landmarks based on their local topology. Similarly, in the current work partial movement matches recognize and prime complete behaviors. Inspired by developmental psychology work providing evidence for infant prediction of goals implied in observed incomplete and incorrect actions (Meltzoff & Moore 1995), our primitives infer movement goals by internally matching, predicting, and completing the observed movements.

In our approach to imitation, the vision system continually matches any observed human movements onto its own set of motor primitives. The primitive, or combination of primitives, that best approximates the observed input also provides the best predictor of what is expected to be observed next. This expectation facilitates visual segmentation and interpretation of the observed movement. Imitation, then, is a process of matching, classification, and prediction. Learning by imitation, in turn, is the process of creating new skills as novel sequences and superpositions of the matched and classified primitives. Our imitation approach is hierarchical in structure; it allows the robot to initially observe and imitate a skill, then perfect it through repetition, so that the skill becomes a routine and thus a primitive itself. As a result, the set of primitives can be adapted over time, to allow for learning arbitrary new skills.

Figure 1 shows the overall structure of our imitation architecture, including the visual perception and attention module, the classification module, and the motor primitives [Mataric 2000]. The learning component is also shown, allowing adaptation both at the level of classification, for finding a closer match to the observed behavior, and repetition for optimizing and smoothing the performance.

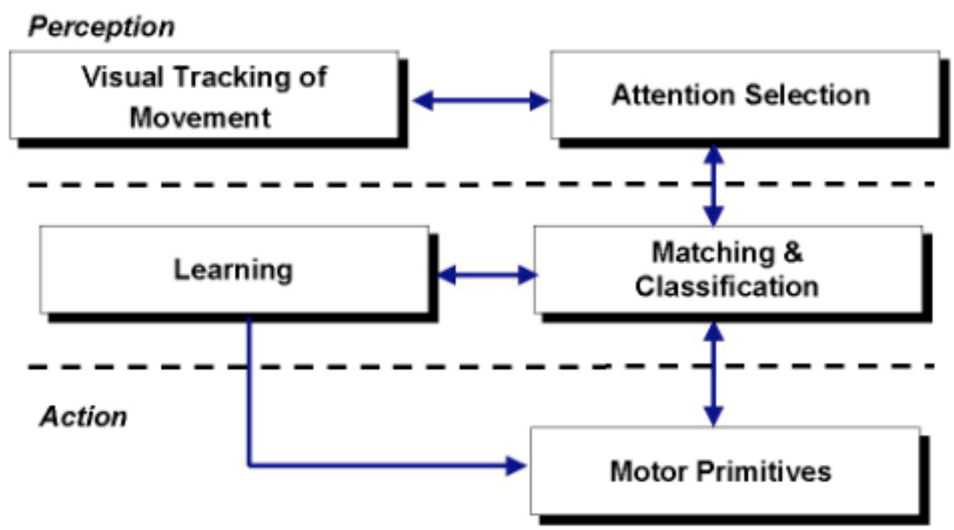

FIGURE 1: Our imitation architecture, structured around perceptual-motor primitives used for movement segmentation, classification, and generation.

8.2.4 Choosing the Primitives

Behavior primitives are the unifying mechanism between visual perception and motor control in our approach, and choosing the right ones is a research challenge, driven by several constraints. On the one hand, the motor control system imposes physical bottom-up limitations, based on its kinematic and dynamic properties. It also provides top-down constraints from the type of movements the system is expected to perform, since the primitives must be sufficient for the robot's entire movement repertoire. On the other, the choice of primitives is also influenced by the structure and inputs into the visual system, in order to map the various observed movements into its own executable repertoire.

In order to serve as a general and parsimonious basis set, the primitives encode groups or classes of stereotypical movements, invariant to exact position, rate of motion, size, and perspective. Thus, they represent the ``generic" building blocks of motion that can be implemented as parametric motor controllers. Consider for example a primitive for reaching. Its most important parameter is the goal position of the end-point, i.e., hand or held object. It may be further parametrized by a default posture for the entire arm. Such a primitive enables a robot to reach toward various goals within a multitude of tasks, from grasping objects and tools, to dancing, to writing and drawing. We used just such a reaching primitive in our experiments, in order to, for example, reconstruct the popular dance Macarena (Mataric et al. 1999).

This approach to motor control stands in sharp contrast to the explicit planning approach to controlling robot manipulators, which computes trajectories at run-time, whenever they are needed. While fully general, on-demand trajectory generation is computationally expensive and potentially slow. In our approach, instead of computing trajectories ***de novo*****,** stereotypical trajectories are built-in as well as learned, then merely looked up and parameterized for the specific task at hand. The notion of primitives takes advantage of the fact that it is simpler to learn and reuse an approximation of the inverse kinematics for specific areas in the workspace or a specific trajectory, than it is to compute them anew each time.

What constitutes a good set of primitives? We have experimented with two types: innate and learned. Innate primitives are user-selected and pre-programmed. We have demonstrated the effectiveness of a basis set consisting of three types: 1) discrete straight-line movements of subsets of degrees of freedom (DOFs), accounting for reaching-type motions; 2) continuous oscillatory movements of subsets of DOFs, accounting for repetitive motions; and 3) postures, accounting for large subsets of the body's DOFs (Mataric et al. 1999).

Learned primitives are computed directly from human movement data. We have gathered different types of such data, using various motion tracking methods (vision-based and magnetic). We first segment the data (manually or automatically), then apply a dimensionality reduction technique followed by clustering in order to extract patterns of ``similar'' movements in the data. Importantly, this purely data-driven approach is effective in discovering underlying structure in movement streams without human priming (Jenkins & Mataric 2002). The resulting clusters form the basis for the primitives; the movements in the clusters are generalized and parameterized, in order to be turned into general primitives for producing a variety of similar movements.

8.2.5 Visual Classification Into Primitives

Visual perception is also an important constraint on the primitives, and a key component of the imitation process. Since the human (and humanoid) visual attention is resource-limited, it must select the visual features that are most relevant to the given imitation task. Determining what those features are for a given demonstration is a challenging problem.

Our previous work showed that people watching videos of arm movements show no difference in attention whether they are just watching, or intending to subsequently imitate. In both cases, they fixate at the end-point, i.e., the hand or a held object (Mataric & Pomplun 1998). However, it is impossible to classify all possible end-point trajectories into a useful set of task-independent categories or primitives. Fortunately, this is not necessary, since the imitation system is targeted for mapping observed movement of bodies similar to the observer's, to the observer's own motor repertoire. Accordingly, the mirror system is sensitive to biological motion of similar bodies (for example, in monkeys it was shown to respond to monkey and human movements). Furthermore, although human(oid) bodies are potentially capable of vast movement repertoires, the typical, everyday movement spectrum is not nearly as large.

Consequently, the visual perception mechanism can be effectively biased toward recognizing movements it is capable of executing, and especially those it performs most frequently. The structure of the motor control system, and its underlying set of movement primitives, provides key constraints for visual movement recognition and classification. Our primitive classifier uses the descriptions of the primitives to segment a given motion based on the movement data. We have used different types of classifiers. In one, end-point data for both arms were used as input for the vector quantization-based classifier (Arya & Mount 1993, Weber et al. 2000). As discussed, a key issue in classification is representing the primitives such that they account for significant invariances, such as position, rotation, and scaling. This allows for a small set of high-level primitive representations instead of a potentially prohibitively large set of detailed ones. Instead, the details of the observed movement can be used for parameterizing the selected primitive(s) at the level of movement reconstruction and execution.

To validate our approach, we have implemented various movement tasks, including reaching, ball throwing, aerobics, and dance, on humanoid test-beds. These are learned from human movement demonstrations, described next.

8.2.6 Experimental Validation

We have experimented with different types of such primitives on Adonis (Mataric et al. 1999). Specifically, we implemented two versions of the spinal fields found in animals, described earlier. One closely modeled the frog data, and used a joint-space representation, i.e., it controlled individual joints of Adonis' arms. The other used another biologically-inspired approach, impedance control (Hogan 1985), which operates in the external coordinate frame, in our case each of Adonis' hands. Our impedance motor controller applied forces at the hands and ``dragged'' the rest of the arm along. We also used a default posture for the arm, which provided natural-appearing whole-arm movements that reached the desired hand destination.

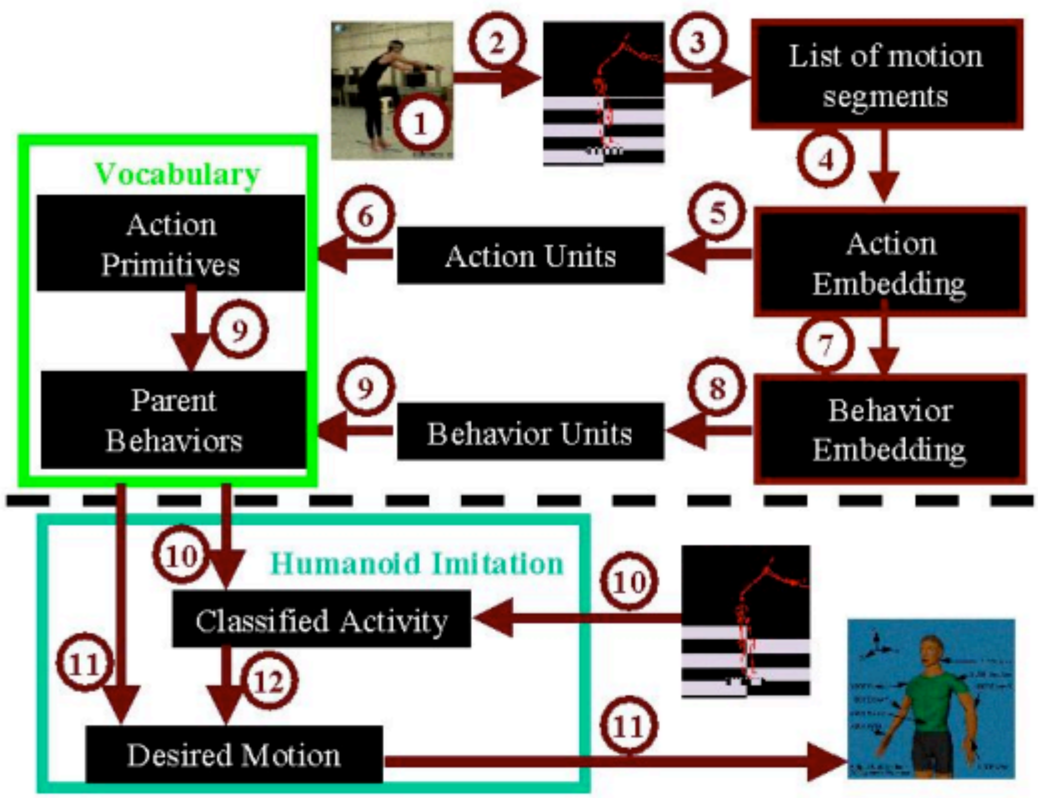

FIGURE 2: Snapshots of Adonis dancing the Macarena

We tested both types of primitives on a complicated sequential motor task, dancing the Macarena. Both were shown to be effective, but each had limitations for particular types of movements. This has led us to propose and explore a combination approach, where multiple types of primitives can be sequenced and combined. For example, we constructed a basis behavior set consisting of three types of primitives: 1) discrete straight-line movements using impedance control; 2) continuous oscillatory movements using coupled oscillators (or a collection of piece-wise linear segments using impedance control); and 3) postures, using PD-servos to directly control the joints. We also added a forth type of primitive, for avoidance, implemented as a repulsive vector field. The fourth primitive was continuously active, and combined with whatever other primitive was being executed, in order to prevent any collisions between body parts. In the Macarena, for example, this is necessary for arm movements around and behind the head (see Figure 2).

These results from using primitives for movement control were used as a basis for the imitation architecture. Primitives provide higher-level descriptions of a movement and visually observed metric information is used to parameterize those descriptions for generating executable movement. This allows for a small number of general primitives to represent a large class of different movements, such as reaches to various destinations on and around the body.

FIGURE 3: A selection of imitation results showing, from the top down, a throw, twists, and arm raises. On the left, plots of the human demonstrated movements and reconstructed motions; on the right, the classification results for each.

To validate our approach, we selected imitation tasks from athletics, dancing, and aerobics; 6 tasks were performed in front of the vision system and presented to our imitation system. Figure 3 shows an example of the imitation system performance on three human-demonstrated movements: a throw, a twist, and a raising of the arms. Videos demonstrating imitation of various types of movements, including details of the input, tracking system, associated mapped primitives, and Adonis imitating can be found at http://www-robotics.usc.edu/$\sim$~agents/imitation.html .

Note that since the movement primitives represent whole-arm movements, our visual system can use either only the end-point (hand) location for imitation, or can gather higher fidelity data that includes the position of the elbow and other markers. In the experiments described here, the position of the hand over time was sufficient for effective imitation. The hand trajectory over time is segmented using the classifier, which, at each point in time, matches the expected output of each of the primitives with the observed input, and selects the best match. Consecutive matches of the same primitive indicate a higher confidence in the match. The output of the classification is a sequence of primitives and their associated parameters. These are then sent to the motor control system, and activate the primitives in turn (and avoidance in parallel), in order to reconstruct the observed behavior.

8.2.6 Summary

Control of and interaction with humanoid robots is a complex problem. Neuroscience evidence points toward some promising avenues for structuring and modularizing the problem in order to make it tractable for real-time control.

8.3 Biped Walking Control

8.3.1 Introduction

Animals need to move around in order to survive. Some animals, most notably birds and humans, do so by walking bipedally. Bipedalism in robots is advantageous for the same reasons it is in bipedal animals. For a given body size, it helps maximize the height of the head, and all its sensory organs, above the ground in order to have a high viewing platform upon the rest of the world. If the number of limbs are limited, then using only two of them (as apposed to 4 or 6) for locomotion frees the others up for manipulation. The footprint of a biped can be made small so that moving through tight passageways and doors is easy. Many environments are engineered for bipedal humans. Bipedal robots should be best suited to manuver in these environments. Another factor for creating bipedal robots is the quest to make something similar to ourselves, which brings up many moral and psychological issues.

Bipedal robots can be bimimetic in shape, control, actuation or all of the above. Robots that may appear biological on the outside may be far from biological on the inside. We will discuss control systems for bipedal robots, which are biomimetic as a function of their shape, but the control systems are not necessarily biomimetic.

In this chapter we will examine some of the technical approaches researchers have used in developing bipedal robots. These include modulated playback of prerecorded joint trajectories, mathematical synthesis of a controller, passive dynamic robots, and physics based heuristics. For each approach, we will look at a case study of a functional robot that was built and controlled using that approach.

8.6.2 The Bipedal Walking Problem

Walking is easy. Or is it? Many do not appreciate the skills required in order to walk until they try to reproduce it in a machine, simulate it in a virtual world, or augment it with assistive devices. But what does it mean to walk? One can buy a simple wind-up toy that walks across a flat surface. If such a toy constitutes walking, then how can we measure progress in the field of bipedal walking robots? We are unaware of anyone precisely defining bipedal walking, but there seems to be a number of subjective criterions:

- Smoothness: One measure of this would be minimizing acceleration or possibly jerk.
- Speed: How fast the robot walks. To be fair, the speed should be scaled to the body height of the robot, or to the square root of the height of the robot to preserve dynamic similarity between robots of different heights.

- Maneuverability: How quickly the robot can turn, walk in different directions.
- Robustness to rough terrain: How rough of terrain a robot is able to traverse without falling. This needs to be in terms of robot size as well.
- Robustness to disturbances: How large of a force distirbance of ground impedance change a robot call handle without falling down. This should be normalized in terms of robot weight and size.
- Biological Similarity: How similar in form and internal function is this to a biological creature.

The control of bipedal wakling is a challenging problem, not currently solveable by "text-book" control theory. Some of the characteristics that make it difficult include:

- Nonlinear: Linear systems are of the form $\dot{X} = AX + Bu$ where X is the state of the system (joint angles and velocities), u is the input (from actuators), A is a constant matrix, and b is a constant vector. A bipedal walking robot has non-linear dynamics, and thus the tools that work for linear systems typically cannot be applied to bipeds, except in special cases.
- Multi-variable: A bipedal robot has many degrees of freedom. Typical bipedal robots have 12 or more joints in their lower body. Many humanoid robots have many more degrees of freedom in their upper body.
- Time Variant: During walking, the dynamics of a bipedal robot abruptly change as it transitions from being supported by one leg to being supported by both legs, to being supported by the other leg. Thus, bipedal robots have characteristics of both continuous and discrete systems, which makes both control design and analysis more difficult.
- Naturally unstable dynamics: Without control, most bipedal robots will fall down.
- Under-actuated: Because a bipedal robot's foot can only push on the ground, and not pull on the ground, control is limited. For example, if the robot's center of mass is outside the "support polygon" of the foot, there is little that can be done to prevent falling, except to be caught by the next support foot.
- Subjective Performance Evaluation: A successful walk results in getting from point A to point B safely, efficiently, and quickly. Exact joint trajectories and muscle firing patterns are not strictly important. Therefore it is difficult to determine a cost function that automatic learning techniques and many control synthesis techniques require.

8.6.3 Biped Robot Control Approaches

Many approaches have been taken in developing bipedal walking robots and their corresponding algorithms.

8.6.3.1 Modulated Playback

Bipedal walking at constant speed on flat ground results in a repetitive, periodic motion. In 1972 researchers at Waseda University built a robot, WL-5, which was controlled with a bang-bang hydraulic servo controller. The desired inputs to this controller where in a tabular format. Much like a wind-up toy, the robot moved its joints in a repetitive pattern and when placed on the ground, it walked, albeit slowly. This robot was a static walker, always keeping its center of pressure over its support foot.

In 1981, the Waseda engineers extended this approach to Quasi-dynamic walking with WL-9RD. In further years they achieved fully dynamic walking. However, the walking was not robust to disturbances. In particular, the robot would fall down if it encountered even a 1cm variation in the floor height. To compensate for floor variations, the trajectories were computed offline in a dynamical simulation and modified until they resulted in a stable motion in simulation. They were then transferred to the robot and played back, often resulting in stable walking.

A compliant foot was added on WL-12RVI and WL-12RVII [Yamaguchi et al. (1994, 1995, 1996)]. This compliant foot could detect forces and moments. The previous control method was used for flat ground walking. However, when the robot encountered a small step 12 mm high, the foot detected the change in terrain and the preset joint trajectories were modified to account for the height change. After adapting to the terrain, the robot returned to its preset trajectories.

This approach has been taken one step further with the advancement of computational power. The trajectory modulation can now be done online, while the robot is walking, in quick response to a disturbance or changing variation in terrain. The Honda Asimo robot is perhaps the currently most advanced of the robots which take this approach.

A typical methodology employed in modulated playback control consists of the following:

1. Determine trajectories, either by
    a. capturing motion on a human,
    b. using parameterized spline curves,
    c. or through optimizing some metric.
2. Determine through dynamical simulation, whether those trajectories will result in stable motion.
3. Iterate steps 1 and 2 until a suitable trajectory for the desired task is achieved.
4. Play the generated trajectory back on the robot using high gain position feedback.
5. Measure the ground reaction forces, and body orientation and compare to the predicted forces and orientations from simulation.
6. If the robot deviates too far from the predicted motion, change the trajectories to compensate for the deviation.

Advantages

- New tasks can quickly be added through motion capture, or through a motion generating interface.
- Predicted stability margins can be determined offline.

Disadvantages

- Not very robust to disturbances since trajectories are rigidly tracked using high gain position servos.
- Typically requires the specification of all the joint trajectories
- High stiffness at joints required means very sensitive to delays and noise.

8.6.4 Mathematical Synthesis

Algorithms for many systems are developed using a mathematical synthesis, which often result in a system which has provable stability and performance conditions. These techniques include root placement, H-2 and H-infinity designs, Lyapunov-function based designs, and robust adaptive control.

The recipe for these types of control is generally the following:

1. Model the system, perhaps including bounds on modeling errors, noise, and disturbances.
2. Define your performance and stability criterion.
3. Synthesize a controller and a proof that it will match the desired performance and stability criterion.

For many systems, such as rockets, airplanes, submersible vehicles, air conditioning units, and a host of other commercial products, this approach works wonderfully. However, for bipedal robots, mathematical synthesis of a controller is difficult. This is because bipedal robots are non-linear, are under-actuated, have a combination of discrete and continuous dynamics, and have vague performance requirements. Because of these difficulties, there are few examples of bipedal algorithms being developed using mathematical synthesis. An exception is BIPER, a robot built by Miura and Shimoyama at Tokyo University (Miura and Shimoyama 1984). This robot walked with straight legs, with small joint angle excursions, so that a linear approximation was appropriate, and linear control synthesis methods could be used.

Advantages

- Mathematically provable stability and performance criterion.

Disadvantages

- Unrealistic assumptions, such as linearity, must be placed on the system, in order to work with today's limited mathematical synthesis techniques.
- Typically requires full specification of all the joint trajectories.
- Very inefficient due to height constraint on hip.

8.6.5 Passive Dynamics

Many dynamical physical systems are stable without any active feedback. For example, a glider will float in the air stably. McGeer at Simon Frazer University performed some pioneering work showing that bipedal robots could be made to dynamically walk down a shallow slope, with no sensing, actuation, or feedback (McGeer 1989;McGeer 1990a;McGeer 1990b). He developed a methodology for studying the stability of such systems as follows:

1. Model a passive dynamic walker and derive equations of motion. These equations include continuous differential equations governing the continuous motion during a support phase; discrete impact equations governing the impact during exchange of support; and the conditions, which cause a discrete impact.
2. Solve the continuous equations and combine with the impact equations, resulting in a single equation, called the stride function. The stride function is a Poincare map relating the state during one part of a step with the state during the same part of the the next step. For some systems, this step can be done analytically, but for most systems, it is done numerically through simulation.
3. Find fixed points of the stride function and linearize it about these fixed points, resulting in a linear Jacobian matrix.

4. Compute eigenvectors and eigenvalues of this Jacobian matrix to determine modes and stability of the system.
5. Examine e_ects of parameter variation on the number of _xed points and their stability. This leads to bifurcation diagrams.
6. Determine the region of convergence and study how parameter variation e_ects the region of convergence.
7. Create an experimental machine. Tweak its parameters based on numerical and real experimentation.
8. Draw conclusions.

This method can be applied analytically to simple systems, but requires numerical simulation and experimentation for more complicated systems. Fixed points, corresponding to periodic gait cycles, can be found through a numerical search by starting the simulation with various initial conditions and examining convergent trajectories. Jacobian matrices can be experimentally determined by disturbing the simulation from the periodic gait and recording state/next-state pairs. A number of such pairs can be obtained under various disturbances and used to solve for a linear matrix fit of the Jacobian.

Many in the passive dynamic walking community have suggested ways in which a passive machine could be equipped with actuators to increase its performance. McGeer suggested using plantar flexion to enable a passive walker to walk on at ground or uphill. Goswami, et. al [Goswami et al. (1997)] presented a simple control law for increasing the region of convergence of a two link passive walker. Van der Linde [der Linde (1998)] built a machine which walks on level ground by pumping energy each step into a passive mechanism. Camp [Camp (1997)] showed that open loop ankle torque during toe-off results in stable walking on at ground. Asano, et al.[Asano et al.(2001)] have been experimenting with mostly passive walkers with motors which apply torques according to "virtual gravity". However, no one has yet built a passive dynamic
walking machine which has the capabilities of powered robots, such as being able to walk at various speeds on various terrain.

Advantages

- Smooth looking motion.
- Highly efficient.
- Inexpensive

Disadvantages

- No control of speed or heading direction yet.
- Lack of power makes passive dynamic robots impractical for many tasks, like climbing stairs.

8.6.6 Physics Based Heuristics

Due to the lack of textbook control techniques for controlling bipedal walking robots, many researchers have relied on developing heuristic controllers based on simple physical models, and intuition. Examples include Timmy at Harvard (Dunn and Howe 1994;Dunn and Howe 1996), Meltran at the Mechanical Engineering Lab in Tsukuba, Japan [Kajita and Kobayashi 1987;

Kajita et al. 1990; Kajita and Tani 1991a; Kajita and Tani 1991b; Kajita et al. 1992; Kajita and Tani 1995a; Kajita and Tani 1995b; Kajita and Tani 1996; and Kajita et al. 2001), and almost all of the robots developed at the MIT Leg Laboratory. A typical methodology for developing such controllers consists of the following steps:

1. Develop simple physical models of the system, both quantitative and qualitative.
2. Break the problem into smaller subtasks.
3. Determine one or more control strategies for each of those subtasks.
4. Develop a controller to implement each strategy.
5. Tune the parameters of each controller either manually or through brute force search.
6. Iterate until the desired performance is achieved.

Advantages

- Often results in smooth biological-looking motion.
- Typically more robust to disturbances since exact joint trajectories are not important.
- Can use physical insight and intuition in algorithm development.
- Leads to further understanding of the relevant physics in the problem.

Disadvantages

- Stability and performance proofs are often skipped, and left for others to do.
- A new algorithm is required for each new task.
- Dependent on expert knowledge from the algorithm developer.
- Requires manual or brute force tuning of many control parameters.

8.6.7 Case Studies

We will take a look a four different bipedal robots and how they are controlled. We will look at the control system in several ways

1. What is the control system?
2. How do the control system and mechanical system influence each other?
3. How biologically plausible is the control system?
4.

8.6.7.1 Modulated Playback Case Study: The Honda Robots

(Hirai 1997;Hirai 1998;Hirai et al. 1998;Hirai et al. 1999;Hirai 1999)

Honda unveiled their first biped, P2, in 1996 and has subsequently releases two more versions, P3 and ASIMO. The robots are different sizes but all roughly the same degrees of freedom and walking controllers.

The walking control system for the Honda robot is essentially a modified playback system. The Walking Pattern Generator(WPG) and Dynamic Robot Model(DRM) generate joints angle trajectories that are to be "played" on the robot. These joint angle trajectories are chosen such that the Zero Moment Point(ZMP) or center of pressure(COP) always remains within the support polygon. As long as the ZMP remains within the support polygon, the robot will be stable. The robot is always monitoring the actual center of ground reaction force and comparing this to the desired ZMP. This is used to update the joint torques as well as update the desired joint angle trajectories.

The walking controller for the robot has several pieces:

- Walking Pattern Generator
- Dynamic Robot Model
- Model ZMP Control/Foot Langing Position control
- Ground reaction force control
- Joint Angle Displacement control
- Body inclination control

The Walking Pattern Generator(WPG) generates desired ZMP position and foot placement positions. These are entered into the DRM which generates desired joint angles and these are fed through the Joint Angle Displacement Control(JADG) which feeds into the robot joints.
The Model ZMP control and Foot Landing Position control vary the input into the WPG and DRM based on the actual robot performance.
Honda has had great success walking on flat terrain and known sloped and staired terrain. It is questionable whether or not the tight position control and feed-forward trajectory following will ever be able to walk robustly over uncertain terrain.
Inoue et al(2001) outline the Honda P3 robot's role in the ongoing Humanoid Robotics Project in Japan. Walking control seems to be considered a finished problem and development efforts are focused around teleoperation, environment awareness and high level tasks.

8.6.7.2 Mathematical Synthesis Case Study: University of Tokyo Biper

(Miura and Shimoyama 1984)

BIPER-1,2,3,4, and 5 are bipedal walking robots developed at the University of Tokyo. These robots are some of the closest examples to using mathematical synthesis to develop a controller that is provably stable. Doing so requires some major assumptions, particularly that the dynamics are linear. Assuming that the dynamics are linear requires use of the small angle assumption. In biological bipeds, the joint angles typically make very large excursions, and thus this assumption is not valid. With the BIPER robots however, the angle excursions are small, resulting in a "stilt-like" walking motion.

Other assumptions that are used in order to employ this method include the following:

- The motions about roll, pitch, and yaw are independent. This assumption is not necessary, but does simplify the mathematics a bit.
- Motion about the yaw axis was neglected.
- Foot contact is via a point, modeled as a universal joint.
- Exchange of support is instantaneous.
- Inelastic collision with the ground. Conservation of momentum is approximated such that the discrete exchange of support matrix is constant.

During single support, the equations of motion are approximated by those of an inverted pendulum with small angle approximation. Written in matrix form, we have:

$$\dot{X} = AX + Bu + E$$

where X is the state vector with m degrees of freedom. A is an m by m matrix, B is an m by p matrix, where p is the number of actuators and p<m, and E is a m by 1 constant vector.

Due to the limited number of actuated degrees of freedom, this system is hard to stabilize (balance the pendulum upright). However, the goal is to walk. At the exchange of support, the state variables undergo a discrete transition, governed by the exchange of support matrix:

$$X_i = \Psi X_f$$

where X_f is the final conditions of the continuous dynamics and X_i is the initial conditions for the next continuous dynamics.

Next, trajectories are planned that satisfy the dynamics. In (ref), this was done through further simplification of the dynamics and through parametric spline equations. In general, one determines reference trajectories X^*, control input u^* and valid transition conditions such that:

$$\dot{X}^* = AX^* + Bu^* + E , \quad X_i^* = \Psi X_f^*$$

If the dynamics were exact and the actuators perfect, then applying these inputs would result in the desired trajectories. However, these equations in general will be unstable, and thus some sort of stabilizing feedback must be employed.

Let $\Delta X = X - X^*$ and $\Delta u = u - u^*$. Rewriting, we get

$$\Delta\dot{X} = A\Delta X + B\Delta u , \quad \Delta X_i = \Psi \Delta X_f$$

If $\Delta u = 0$, then we can solve these equations of motion to get:

$$\Delta X_f = e^{AT}\Delta X_i ,$$

where T is the time to go from the beginning of single support phase to the beginning of the transition phase. Adding the transition, we get

$$\Delta X_i^{(n+1)} = \Psi e^{AT} \Delta X_i^{(n)}$$

This is a discrete linear transition equation. It will be stable if the eigenvalues of the matrix Ψe^{AT} are all less than unity in magnitude. However, with BIPER, these equations are not stable.

These equations can be stabilized by appropriate control. Miura and Shimoyama determined that foot placement could be used to stabilize the robot. One can achieve foot placement by the addition of control, based on the error in the robots state:

$$\Delta u = -G\Delta X$$

In general, G can be used for both foot placement and to make the robot better track its trajectory. If G is a constant gain matrix, then the continuous dynamics with feedback become

$$\Delta X_f = e^{(A-BG)T}\Delta X_i$$

Adding in the discrete transition matrix, we get

$$\Delta X_i^{(n+1)} = \Psi e^{(A-BG)T}\Delta X_i^{(n)}$$

If a G is chosen such that the eigenvalues of $\Psi e^{(A-BG)T}$ have magnitude less than one, then the system will be proven stable, modulo the assumptions that were made in developing the controller. Note that the above equation is independent of the reference trajectories! Therefore, once a feedback matrix G is chosen that results in a stable system, that G should stabilize other sets of reference trajectories. Since the system is linear, G may be easily chosen through one of many textbook control techniques, including root locus, pole placement, or some optimization routine. Note that since the system is underactuated, there may not be a G which stabilizes the system. With the BIPER designs, however, a suitable G was found.

In (ref), G was chosen through intuition and iteration. Since speed can be controlled through foot placement, G was chosen to effectively take a longer stride when speed was increased. The eigenvalues of $\Psi e^{(A-BG)T}$ where then verified to be less than unity, thus proving stability of the system.

It is interesting to note that $e^{(A-BG)T}$ will likely have eigenvalues with magnitude greater than unity, yet $\Psi e^{(A-BG)T}$ will have eigenvalues with magnitude less than unity. Intuitively, this means that the robot is unstable unless it transitions from step to step. This is the mathematical equivalent of the often heard observation that walking is a sequence of unstable falling motions, stabilized by the next support foot braking the fall.

The above control system design methodology can be applied to any biped with linearized dynamics. However, as the robot gets complicated, and the number of degrees of freedom increases, choosing a reference trajectory that is consistent with the dynamics, and a feedback matrix that stabilized the dynamics, gets more and more complicated.

5.6.8 Passive Dynamics Case Study: Passive Dynamic Walking

(McGeer, 1990a)

McGeer built and analyzed several passive dynamic walkers(PDW). He successfully demonstrated straight legged as well as kneed walking. He also investigated ways of pumping energy into a mostly passive walker so it would be able to walk on flat ground.
For a simple straight legged, two linked passive walker, a simple linearized analysis is possible. Linearity is assumed due to the small angles swept during a normal cycle as well as the low relative speed($<< \sqrt{l/g}$).

Below is a brief outline of the analysis procedure. The full derivation is left to the original papers. The key adjustable parameters are R(foot radius), r_gyr(Leg radius of gyration), c(leg com height), w(fore/aft com offset), and m_h(hip mass fraction). These will affect the dynamic equations of the robot.
Linearized equations about vertical can be written as follows:

$$M_o\dot{\vec{\Omega}} + C\vec{\Omega} + K\Delta\vec{\Theta} = K\Delta\vec{\Theta}_{SE}$$

$\vec{\Omega}$ is the vector of link speeds and $\Delta\vec{\Theta}$ the vector of angles from the surface normal. $\Delta\vec{\Theta}_{SE}$ is the static equilibrium. The previous equation can be solved to relate start and end of step via transistion matrices.

$$\Delta\vec{\Theta}(\tau_k) = D_{\Theta\Theta}[\Delta\vec{\Theta}_k - \Delta\vec{\Theta}_{SE}] + D_{\Theta\Omega}\vec{\Omega}_k + \Delta\vec{\Theta}_{SE}$$

$$\vec{\Omega}(\tau_k) = D_{\Omega\Theta}[\Delta\vec{\Theta}_k - \Delta\vec{\Theta}_{SE}] + D_{\Omega\Omega}\vec{\Omega}_k$$

Two things then occur, we have an inelastic impulse at touchdown and the stance and swing legs are flipped.

$$\vec{\Omega}_{k+1} = \Lambda\vec{\Omega}(\tau_k)$$

Λ takes care of both for the speeds and F takes care of flipping for the angles.

$$\Delta\vec{\Theta}_{k+1} = F\Delta\vec{\Theta}(\tau_k)$$

Combining the last four equations, the step-to-step relations are created.

$$\Delta\vec{\Theta}(\tau_{k+1}) = F(D_{\Theta\Theta}[\Delta\vec{\Theta}_k - \Delta\vec{\Theta}_{SE}] + D_{\Theta\Omega}\vec{\Omega}_k + \Delta\vec{\Theta}_{SE})$$

$$\vec{\Omega}(\tau_k) = \Lambda(D_{\Omega\Theta}[\Delta\vec{\Theta}_k - \Delta\vec{\Theta}_{SE}] + D_{\Omega\Omega}\vec{\Omega}_k)$$

Solving the euqations will give us initial conditions which repeat from step to step.

$$\vec{\Omega}_0 = [I - \Lambda D_{\Omega\Omega}]^{-1}\Lambda[D_{\Omega\Theta}[\Delta\vec{\Theta}_k - \Delta\vec{\Theta}_{SE}]]$$

$$\vec{0} = [D' - F]\Delta\vec{\Theta}_0 - [D' - I]\Delta\vec{\Theta}_{SE} + D_{\Theta\Omega}\vec{\Omega}_k + \Delta\vec{\Theta}_{SE})$$

where

$$D'(\Delta\vec{\Theta}_0, \tau_0) = D_{\Theta\Theta} + D_{\Theta\Xi}[I - \Lambda D_{\Omega\Omega}]^{-1}\Lambda D_{\Omega\Theta}$$

Solutions to the steady cycle equation can be found by guessing initial angles and iterating. Linearization of the step-to-step equations can be used for stability analysis. There are obvious cases(feet spread far apart, feet together) that will not lead to a stable walking cycle but many will. Further details on the effects of parameter variances, pumping inputs, and disturbances are available in the literature

5.6.9 Physics Based Heuristics Case Study: MIT Spring Flamingo

(Pratt et al. 1997;Pratt and Pratt 1998a;Pratt and Pratt 1998b;Pratt et al. 2001)

Spring Flamingo is a planar bipedal walking robot developed at the MIT Leg Laboratory. It has six actuated degrees of freedom: one in each hip, knee, and ankle. Physics based heuristic control strategies were used in order to make the robot walk.

The control strategies are based on the observation that five conditions have to be met for a planar bipedal robot to walk. Height needs to be stabilized. Pitch needs to be stabilized. Speed has to be stabilized. The swing-leg needs to move so that the feet are in locations that allow for the stability of height, pitch, and speed. Finally, transitions from support leg to support leg must occur at appropriate times. If these five objectives are achieved, then the robot will walk. A number of different control strategies can be used to achieve each of these five objectives. Further, each strategy can be implemented in a variety of ways.

Several algorithms were used to control Spring Flamingo. These algorithms were all based on simple physical models of walking and employ simple control strategies to achieve the five objectives listed above. For instance, one of the control strategies consisted of the following heuristics:

- Height:
  - Keep the support leg below the robot fairly straight to maintain height. Use the knee joint limit stop to prevent the knee joint from inverting.
- Pitch:
  - Maintain a constant level pitch using a virtual spring-damper mechanism with constant set point.
- Support Transitions:
  - Transition from double support to single support when the body's forward position becomes further than a preset distance from the rear foot or closer than a preset distance from the front foot.
  - Transition from single support to double support when the body's forward position becomes further than a preset distance from the support foot.
- Swing Leg:
  - Start the swing by torquing the hip. Let the swing leg swing through passively. Then brake the swing leg with the hip. Add a little damping at the knee. At the end of stride, hold the leg such that the foot will be placed a desired stride length away from the support foot.
- Speed:
  - Increase the nominal stride length as the robot walks faster.
  - Delay transition to double support if the robot is walking too slowly. Conversely, initiate transition to double support sooner if the robot is walking too quickly.
  - Maintain the center of pressure of the support foot approximately below the center of mass, moving it forward if walking too quickly or backward if walking too slowly.
  - During double support shift the load toward the back leg if walking too slowly or toward the front leg if walking too quickly.

Note that not all degrees of freedom were actively controlled during walking. For instance, the trajectory of the stance ankle was not commanded at all. Instead ankle torque was determined in order to place the center of pressure on the foot at a desired location.

Simple feedback control laws were used to implement the above strategies. The parameters for both the heuristics and the feedback controllers were tuned manually. Using this methodology, Spring Flamingo was able to walk at 1.2 meters per second on flat ground. Simple modifications were made to the flat ground algorithm to allow the robot to walk over rolling terrain.

REFERENCE

Collins, S. H., M. Wisse, and A. Ruina. 2001. A three-dimensional passive-dynamic walking robot with two legs and knees. International Journal of Robotics Research 20: 607-615.

Dunn, E. R. and R. D. Howe. 1994. Towards smooth bipedal walking. Proceedings 1994 IEEE International Conference on Robotics and Automation (Cat. No. 94CH3375-3) 2489-2494.

Dunn, E. R. and R. D. Howe. 1996. Foot placement and velocity control in smooth bipedal walking. Proceedings. 1996 IEEE International Conference on Robotics and Automation (Cat. No. 96CH35857) 578-583.

Hirai, K. 1997. Current and future perspective of Honda humamoid robot. Proceedings of the 1997 IEEE/RSJ International Conference on Intelligent Robot and Systems. Innovative Robotics for Real-World Applications. IROS '97 (Cat. No. 97CH36108) 500-508.

Hirai, K. 1998. Current and future perspective of Honda humanoid robot. Robotics Research. Eighth International Symposium 446-450.

Hirai, K. 1999. The Honda humanoid robot: development and future perspective. Industrial Robot 26: 260-266.

Hirai, K., M. Hirose, Y. Haikawa, and T. Takenaka. 1998. The development of Honda humanoid robot. Proceedings. 1998 IEEE International Conference on Robotics and Automation (Cat. No. 98CH36146) 1321-1326.

Hirai, K., M. Hirose, Y. Haikawa, and T. Takenaka. 1999. The development of Honda humanoid robot P2 and P3. 30th International Symposium on Robotics. Celebrating the 30th Anniversary toward the Next Millennium 775-780.

Kajita, S., F. Kanehiro, K. Kaneko, K. Yokoi, and H. Hirukawa. 2001. The 3D linear inverted pendulum mode: a simple modeling for a biped walking pattern generation. Proceedings 2001 IEEE/RSJ International Conference on Intelligent Robots and Systems. Expanding the Societal Role of Robotics in the the Next Millennium (Cat. No. 01CH37180) 239-246.

Kajita, S. and A. Kobayashi. 1987. Dynamic walk control of a biped robot with potential energy conserving orbit. Transactions of the Society of Instrument and Control Engineers 23: 281-287.

Kajita, S. and K. Tani. 1991a. Study of dynamic biped locomotion on rugged terrain-derivation and application of the linear inverted pendulum mode. Proceedings. 1991 IEEE International Conference on Robotics and Automation (Cat. No. 91CH2969-4) 1405-1411.

Kajita, S. and K. Tani. 1991b. Study of dynamic walk control of a biped robot on rugged terrain-derivation and application of the linear inverted pendulum mode. Transactions of the Society of Instrument and Control Engineers 27: 177-184.

Kajita, S. and K. Tani. 1995a. Dynamic biped walking control on rugged terrain using the linear inverted pendulum mode. Transactions of the Society of Instrument and Control Engineers 31: 1705-1714.

Kajita, S. and K. Tani. 1995b. Experimental study of biped dynamic walking in the linear inverted pendulum mode. Proceedings of 1995 IEEE International Conference on Robotics and Automation (Cat. No. 95CH3461-1) 2885-2891.

Kajita, S. and K. Tani. 1996. Adaptive gait control of a biped robot based on realtime sensing of the ground profile. Proceedings. 1996 IEEE International Conference on Robotics and Automation (Cat. No. 96CH35857) 570-577.

Kajita, S., K. Tani, and A. Kobayashi. 1990. Dynamic walk control of a biped robot along the potential energy conserving orbit. Proceedings. IROS '90. IEEE International Workshop on Intelligent Robots and Systems '90. Towards a New Frontier of Applications (Cat. No. 90TH0332-7) 789-794.

Kajita, S., T. Yamaura, and A. Kobayashi. 1992. Dynamic walking control of a biped robot along a potential energy conserving orbit. IEEE Transactions on Robotics and Automation 8: 431-438.

McGeer, T. 1989. Powered flight, child's play, silly wheels and walking machines. Proceedings. 1989 IEEE International Conference on Robotics and Automation (Cat. No. 89CH2750-8) 1592-1597.

McGeer, T. 1990a. Passive dynamic walking (two-legged machines). International Journal of Robotics Research 9: 62-82.

McGeer, T. 1990b. Passive walking with knees. Proceedings 1990 IEEE International Conference on Robotics and Automation (Cat. No. 90CH2876-1) 1640-1645.

Miura, H. and I. Shimoyama. 1984. Dynamic walk of a biped. International Journal of Robotics Research 3: 60-74.

Pratt, J., C. Chee-Meng, A. Torres, P. Dilworth, and G. Pratt. 2001. Virtual model control: an intuitive approach for bipedal locomotion. International Journal of Robotics Research 20: 129-143.

Pratt, J., P. Dilworth, and G. Pratt. 1997. Virtual model control of a bipedal walking robot. Proceedings. 1997 IEEE International Conference on Robotics and Automation (Cat. No. 97CH35992) 193-198.

Pratt, J. and G. Pratt. 1998a. Intuitive control of a planar bipedal walking robot. Proceedings. 1998 IEEE International Conference on Robotics and Automation (Cat. No. 98CH36146) 2014-2021.

Pratt, J. E. and G. A. Pratt. 1998b. Exploiting natural dynamics in the control of a planar bipedal walking robot. Proceedings. Thiry-Sixth Annual Allerton Conference on Communication, Control, and Computing 739-748.

Maja's references (incomplete, more to be added):

Agre, P. & Chapman, D., 1987, 'Pengi: An Implementation of a Theory of Activity'. Proceedings, Sixth National Conference of the American Association for Artificial Intelligence Conference, Seattle, WA, 268-272.

Agre, P. & Chapman, D. 1990, "What are plans for?" in Designing Autonomous Agents, P. Maes, ed., MIT Press, 17-34.

Mahadevan, S. and Kaelbling, L. P., "The National Science Foundation Workshop on Reinforcement Learning", 'AI Magazine', 17(4), 89-97, 1996.

Kaelbling, L. P., Littman, M. L., and Moore A. W., "Reinforcement Learning: A Survey", 'Journal of Artificial Intelligence Research', 4, 237-285, 1996.

Arkin, R., 1989, 'Towards the Unification of Navigational Planning and Reactive Control'. Proceedings, American Association for Artificial Intelligence Spring Symposium on Robot Navigation, Palo Alto, CA, 1-5.

Asada, M., Uchibe, E., Noda, S., Tawaratsumida, S. & Hosoda, K., 1994, 'Coordination of Multiple Behaviors Acquired by a Vision-Based Reinforcement Learning'. Proceedings, IEEE/RSJ/GI International Conference on Intelligent Robots and Systems, Munich, Germany.

Beer, R., Chiel, H. & Sterling, L., 1990, 'A Biological Perspective on Autonomous Agent Design'. Robotics and Autonomous Systems, Special Issue on Designing Autonomous Agents: Theory and Practice from Biology to Engineering and Back, P. Maes, ed., Vol. 6, No. 1-2, 169-186.

Brooks, R., 1986, 'A Robust Layered Control System for a Mobile Robot'. IEEE Journal of Robotics and Automation, Vol. RA-2, April, 14-23.

Brooks, R. & Connell, J., 1986, 'Asynchronous Distributed Control System for a Mobile Robot'. Proceedings, SPIE Intelligent Control and Adaptive Systems, Cambridge, MA, 77-84.

Brooks, R., 1991, 'Intelligence Without Reason'. Proceedings, International Joint Conference on Artificial Intelligence, Sydney, Australia, MIT Press, Cambridge, MA.

Brooks, R. & Stein, L., 1994, 'Building Brains for Bodies'. Autonomous Robots, Vol. 1, No. 1, 7-25.

Brooks, V., Hilperath, H., Brooks, M., Ross., H., and Freund, H., "Learning What and How in a Human Motor Task", in 'Learning and Memory', Vol. 2, 1995, 225-242.

Connell, J., 1990, 'Minimalist Mobile Robotics: A Colony Architecture for an Artificial Creature', Academic Press, Boston.

Connell, J., 1991, 'SSS: A Hybrid Architecture Applied to Robot Navigation'. Proceedings, International Conference on Robotics and Automation, Nice, France, 2719-2724.

Firby, J., 1987, 'An investigation into reactive planning in complex domains'. Proceedings, Sixth National Conference of the American Association for Artificial Intelligence Conference, Seattle, WA, 202-206.

Georgeoff, M. & Lansky, A., 1987, 'Reactive Reasoning and Planning'. Proceedings, Sixth National Conference of the American Association for Artificial Intelligence Conference, Seattle, WA, 677-682.

Giralt, G., Chatila, R., & Vaisset, M., 1983, 'An Integrated Navigation and Motion Control System for Autonomous Multisensory Mobile Robots'. Proceedings, First International Symposium on Robotics Research, MIT Press, Cambridge, 191-214.

Laird, J. & Rosenbloom, P., 1990, 'An investigation into reactive planning in complex domains'. Proceedings, Ninth National Conference of the American Association for Artificial Intelligence Conference, MIT Press, Cambridge, 1022-1029.

Maes, P., 1989, 'The Dynamics of Action Selection'. Proceedings, International Joint Conference on Artificial Intelligence, Detroit, MI, 991-997.

Maes, P. & Brooks, R., 1990, 'Learning to Coordinate Behaviors'. Proceedings, Ninth National Conference of the American Association for Artificial Intelligence Conference, MIT Press, Cambridge, 796-802.

Malcolm, C. & Smithers, T., 1990, 'Symbol grounding via a hybrid architecture in an autonomous assembly system'. Robotics and Autonomous Systems, Special Issue on Designing Autonomous Agents: Theory and Practice from Biology to Engineering and Back, P. Maes (Ed.), Vol. 6, No. 1-2, 145-168.

Mataric, M., 1990, 'Navigating With a Rat Brain: A Neurobiologically-Inspired Model for Robot Spatial Representation'. Proceedings, From Animals to Animats 1, First International Conference on Simulation of Adaptive Behavior, J-A. Meyer and S. Wilson, eds., MIT Press, 169-175.

Mataric, M., 1992, 'Integration of Representation Into Goal-Driven Behavior-Based Robots'. IEEE Transactions on Robotics and Automation, Vol. 8, No. 3, 304-312.

Mataric, M., 1994, 'Learning to Behave Socially'. Proceedings, From Animals to Animats 3, Third International Conference on Simulation of Adaptive Behavior, D. Cliff and P. Husbands and J-A. Meyer and S. Wilson, eds., MIT Press, 453-462.

Mataric, M., 1997, 'Behavior-Based Control: Examples from Navigation, Learning, and Group Behavior'. Journal of Experimental and Theoretical Artificial Intelligence, Special Issue on Software Architectures for Physical Agents, Vol. 9, No. 2-3, Hexmoor, Horswill, and Kortenkamp, eds., 1997.

Mataric, M., 1997, 'Reinforcement Learning in the Multi-Robot Domain'. Autonomous Robots, Vol. 4, No. 1, 73-83.

Millan, J., 1994, 'Learning Reactive Sequences from Basic Reflexes', Proceedings, From Animals to Animats 3, Third International Conference on Simulation of Adaptive Behavior, D. Cliff and P. Husbands and J-A. Meyer and S. Wilson, eds., MIT Press, 266-274.

Moravec, H. & Elfes, A., 1995, 'High Resolution Maps From Wide Angle Sonar', Proceedings, IEEE International Conference on Robotics and Automation, St. Louis, MO.

Nolfi, S., Floreano, D., Miglino, O., & Mondada, F., 1994, 'Now to Evolve Autonomous Robots: Different Approaches in Evolutionary Robotics'. Proceedings, Artificial Life IV, the Fourth International Workshop on the Synthesis and Simulation of Living Systems, R. Brooks and P. Maes, eds., MIT Press, 190-197.

Parker, L., 1993, 'Learning in Cooperative Robot Teams'. Proceedings, International Joint Conference on Artificial Intelligence, Workshop on Dynamically Interacting Robots, Chambery, France, 12-23.

Payton, D., 1990, 'Internalized Plans: a representation for action resources'. Robotics and Autonomous Systems, Special Issue on Designing Autonomous Agents: Theory and Practice from Biology to Engineering and Back, P. Maes, ed., Vol. 6, No. 1-2, 89-104.

Payton, D., Keirsey, D., Kimble, D., Krozel, J., & Rosenblatt, K., 1992, 'Do Whatever Works: A robust approach to fault-tolerant autonomous control'. Journal of Applied Intelligence, Vol. 3, 226-249.

Schoppers, M. 1987, "Universal Plans for Reactive Robots in Unpredictable Domains", Proceedings, IJCAI-87, Menlo Park, CA, 1039-1046.

Steels, L., 1994, 'Emergent Functionality of Robot Behavior Through On-line Evolution'. Proceedings, Artificial Life IV, the Fourth International Workshop on the Synthesis and Simulation of Living Systems, R. Brooks and P. Maes, eds., MIT Press, 8-14.

Webb, B., 1994, 'Robotic experiments in Cricket Phonotaxis'. In Proceedings of the Third International Conference on the Simulation of Adaptive Behavior. MIT Press.

Williamson, M., 1996, 'Postural Primitives: Interactive Behavior for a Humanoid Robot Arm'. Proceedings, From Animals to Animats 4, Fourth International Conference on Simulation of Adaptive Behavior, P. Maes, M. Mataric, J-A. Meyer, J. Pollack and S. Wilson, eds., MIT Press, 124-134.

Yanco, H. & Stein, L., 1993, 'An Adaptive Communication Protocol for Cooperating Mobile Robots'. Proceedings, From Animals to Animats 3, Third International Conference on Simulation of Adaptive Behavior, D. Cliff and P. Husbands and J-A. Meyer and S. Wilson, eds., MIT Press, 478-485.