



University of Pennsylvania
ScholarlyCommons

Technical Reports (CIS)

Department of Computer & Information Science

January 1995

From Active Perception to Active Cooperation Fundamental Processes of Intelligent Behavior

Ruzena Bajcsy
University of Pennsylvania

Follow this and additional works at: https://repository.upenn.edu/cis_reports

Recommended Citation

Ruzena Bajcsy, "From Active Perception to Active Cooperation Fundamental Processes of Intelligent Behavior", . January 1995.

University of Pennsylvania Department of Computer and Information Science Technical Report No. MS-CIS-95-07.

This paper is posted at ScholarlyCommons. https://repository.upenn.edu/cis_reports/255
For more information, please contact repository@pobox.upenn.edu.

From Active Perception to Active Cooperation Fundamental Processes of Intelligent Behavior

Abstract

In the ten years since we put forward the idea of *active perception* (Bajcsy 1985, Bajcsy 1988) we have found that cooperative processes of various kinds and at various levels are often called for. In this paper we suggest that a proper understanding of cooperative processes will lead to a foundation for intelligent behavior and demonstrate the feasibility of this approach for some of the difficult and open problems in the understanding of intelligent behaviors.

Keywords

cooperation, hybrid systems, modeling agents, perception-action cycle, signal-to-symbol transformation

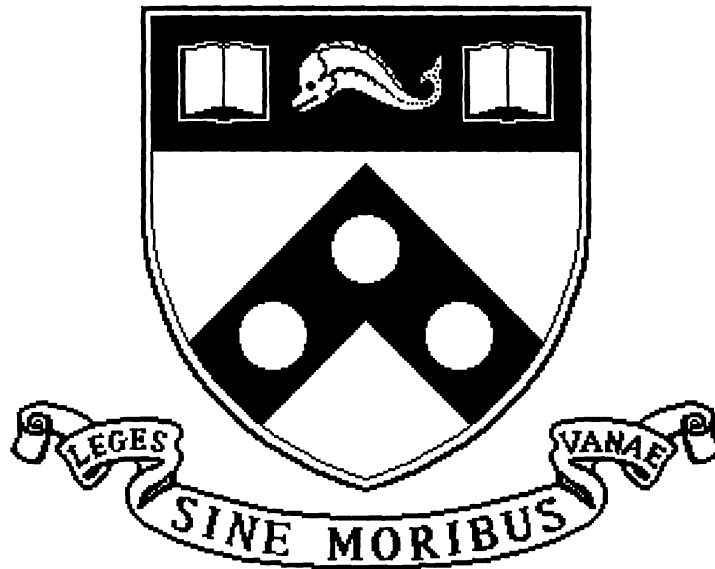
Comments

University of Pennsylvania Department of Computer and Information Science Technical Report No. MS-CIS-95-07.

**From Active Perception to Active Cooperation
Fundamental Processes of Intelligent Behavior**

MS-CIS-95-07

Ruzena Bajcsy



University of Pennsylvania
School of Engineering and Applied Science
Computer and Information Science Department
Philadelphia, PA 19104-6389

1995

From active perception to active cooperation — fundamental processes of intelligent behavior

Ruzena Bajcsy*

General Robotics and Active Sensory Perception Laboratory,
Department of Computer and Information Science,
University of Pennsylvania, Philadelphia, PA 19104, USA

In the ten years since we put forward the idea of *active perception* (Bajcsy 1985, Bajcsy 1988) we have found that cooperative processes of various kinds and at various levels are often called for. In this paper we suggest that a proper understanding of cooperative processes will lead to a foundation for intelligent behavior and demonstrate the feasibility of this approach for some of the difficult and open problems in the understanding of intelligent behaviors.

Keywords: cooperation, hybrid systems, modeling agents, perception-action cycle, signal-to-symbol transformation.

1. INTRODUCTION

Approximately ten years ago we put forward the idea of *active perception* (Bajcsy 1985, Bajcsy 1988), which, contrary to the widely held views of the time, argued that the problem of perception was not necessarily one of signal processing but of control of data acquisition. There were three points to our argument:

1. The agent (human or artificial) not only sees and feels but looks and touches, i.e. perception is an active process of seeking information from the environment. This idea finds its origins in human psychology, especially as formulated by Gibson (1950).
2. If one accepts the first premise, then the next question is this: What are the strategies for selecting the *next* or the *best* view?
3. If one takes several views or other measurements, then another question follows. Namely, how does one combine these different views or measurements? That is,

*I would like to thank my colleagues Profs. Vijay Kumar, Max Mintz, Richard Paul and Xiaoping Yun for their unlimited support and valuable discussions that led to the issues presented in this article. The students involved in the multi-agent project, Julie Adams, Jana Kořecká, Robert Mandelbaum, Chau-Chang Wang and Yoshio Yamamoto, were also critical to the evolution of the ideas presented here. I also want to thank Dr. Yasuo Kuniyoshi and his colleagues at Electrotechnical Laboratory for giving me the peaceful and stimulating environment to pull together this paper. Finally, it is difficult for me to find the right words of thanks to Craig Reynolds, GRASP Lab Administrator, for everything he does. This research has been supported by the following grants: ARO grant DAAL03-89-C-0031, ARPA grant N00014-92-J-1647, and NSF grants STC SBR89-20230, CISE CDA88-22719, and IRI93-07126.

how does one integrate or fuse this information into a coherent representation?

Once formulated, several others set to work on this new paradigm, such as Aloimonos (Aloimonos et al. 1987, Aloimonos 1993), Ballard (1991) on animate vision, and Blake (Blake and Yuille 1992) on dynamic shape.

During the period 1985 through 1988, as we proceeded to follow through with this research program, we found that cooperative processes of various kinds and at various levels were called for. For example, in Krotkov's work (1989) we found that cooperative processes are needed to integrate the two different behaviors of focus and control of vergence for the recovery of robust 3D information. Later we employed the same cooperative philosophy to integration of multimodal sensory information, such as vision and touch (Stansfield 1986, Allen 1987). Since then, we have embarked on studies of cooperation between different manipulators and different agents (Bajcsy et al. 1992, Adams et al. 1995). We have found that cooperation is a fundamental process in organisms both artificial and biological. In this paper, we wish to elaborate on this problem and suggest that a proper understanding of cooperation will lead to a foundation for intelligent behavior.

2. A DEFINITION OF COOPERATION

Cooperation is the process of taking different observations and *a priori* information from the *shared world* (our world includes the agents) and combining it in such a way that the process achieves a *common task*.

Note that there are two critical assumptions in our definition of a cooperative behavior:

1. The shared world and/or knowledge do not have to be identical, but there must be sufficient overlap such that the participating agents have a common understanding/interpretation of the input information and the task.
2. Agents agree on what the common task is, even if the means to accomplish it differ from agent to agent.

Another obvious implication of the cooperative process is that individual agents are distributed in space and/or in time. In this regard, there has been a great deal of work done in the area of Distributed Artificial Intelligence, as witnessed by several workshops and books. Examples include (Gasser and Huhns 1989, Brauer and Hernandez 1991, IJCAI 1991, von Martial 1992).

What, then, are the problems associated with cooperative processes? There are several:

1. Design of individual agents, including such questions as: (a) Should agents be homogeneous or heterogeneous? (b) What degree of autonomy should each agent have? (c) What communication abilities should each agent have? (d) What architecture should be selected for each agent?
2. Control of the cooperative process, including such questions as: (a) Should control be centralized or decentralized? (b) How should tasks be distributed? (c) What should be the means of communication?
3. Modeling agents and their cooperative behaviors.

4. Design of a task specification language and its compiler.

The study of cooperative processes is not new. Operation researchers interested in modeling organization have attempted to do so for the last 25 years, but due to computational limitations they were only able to consider a small number of players, as is the case with the prisoner's dilemma and its like. Recent computer technologies, however, have rekindled the scientific community's interest in this area. It is an interest motivated by two different forces: on the one hand, multidistributed computer networks, and on the other hand, the possibilities of distributed robotics systems in various applications. These two motivations impose different requirements when formulating the problem of cooperative processes. The former leads to software agents, whereas the latter leads to physical agents that interact with a dynamic world often unpredictable in its changes.

As mentioned in the introduction, cooperation takes place on many levels, namely: cooperative sensing, cooperative processing, cooperative manipulation, cooperative behaviors, and cooperative agents. We shall discuss each of these cases below.

3. COOPERATION ON DIFFERENT LEVELS

3.1. Cooperative sensing

Cooperative sensing occurs when one has distributed sensing, meaning either sensors of a single type that are distributed in space, (e.g. several cameras spatially distributed around a scene or object), or several different types of sensors that make observations from the same vantage point or again are spatially distributed (e.g. a camera, laser, ultrasound, and infrared scans taken from the same or different positions in space). The literature is full of such examples: Draper et al. (1989) describe a system for image interpretation by a distributed cooperative process; in the GRASP Laboratory, Mandelbaum (Mandelbaum and Mintz 1994, Mandelbaum 1995) employs ultrasound, a light stripier and stereo for map recovery of the laboratory space, and Bogoni (Bogoni and Bajcsy 1994, Bogoni 1995) makes sequential observations of visual tracking, position encoding and force measurements, and integrates them into one coherent framework in order to asses whether a tool is penetrating an object. The operative principle of all these efforts is that in order for cooperative sensing to work the following must hold:

- In the case of homogeneous sensors, there must be a common shared framework and/or coordinate system.
- In the case of heterogeneous sensors, measurements must be converted into a common representation.
- Different observations must overlap so that continuity is preserved.

3.2. Cooperative processing

The whole field of cooperative processing was initiated by the distributed operating systems community, which confronts the problem of coordinating multiprocessing tasks and the issues of deadlocks, conflicts (i.e. when the same process is concurrently assigned two different jobs), scheduling, synchrony versus asynchrony, and so on. A good text on this subject is (Peterson and Silberschatz 1985). We shall not discuss these related

works, but only mention that some of their formal tools and approaches such as petri nets, temporal logic and constraint nets are in fact useful in our domain as well.

3.3. Cooperative manipulation

The need for cooperative manipulation comes from the desire to improve the performance of robotic systems in areas such as dexterity, payload capacity, grasping, and so on. In principle, there are two types of cooperation, one when manipulators cooperate without the interaction of forces, i.e. manipulators do not hold the same object at the same time, and the other when they cooperate with interaction of forces, i.e. manipulators hold the same object. In the first case, the task of planning is the most important technical issue, while in the second case control of the two robot system is most critical. Historically, Tarn, Bejczy and Yun first dealt with this latter problem in the mid-80's (Bejczy et al. 1986, Tarn et al. 1987a, b). Later several other researchers designed different dynamical controllers to control a nonlinear redundant chain of mechanisms, such as (Dauchez and Uchiyama 1987, Arai and Osumi 1991, Hashimoto 1993, Yamamoto 1994). Cooperation here takes place through physical coupling of the manipulated object, when the manipulators communicate through state variables, i.e. position and forces.

3.4. Cooperative behaviors

In his seminal paper Brooks (1986) challenged the prevailing wisdom regarding architectures for building intelligent agents. The old paradigm considered separate functional units where the information flow was strictly horizontal (the output of one functional unit was the input to another) and the external feedback loop was closed through the environment (Moravec 1983, Nilsson 1984). The functional units included sensing, mapping the sensor readings into the world representation, planning, task execution and motor control (see Figure 1a).

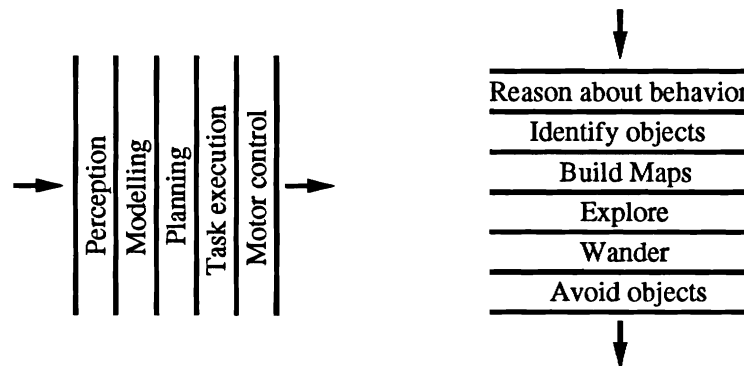


Figure 1. Architectures for robotics control: (a) horizontal (b) vertical.

Even though this kind of horizontal decomposition provides a nice modular partitioning of the problem, it fails to produce robust working systems. An alternative bottom-up approach was motivated by the process of natural evolution. Several researchers observed that a seemingly intelligent behavior can emerge from many simple ones and the interactions between them (Braitenberg 1984, Arbib and House 1987, Arkin 1987, Reynolds 1987, Brooks 1991, Levy 1993). The key idea of Brooks' approach (1986) was to make the loop between perception and action through the world much tighter by developing

perception-action processes with different levels of competence ranging from simple to complex. Decomposition of the design of autonomous agents is done in a vertical fashion (see Figure 1b) where the component behaviors are activated in parallel, issuing appropriate commands to the agents' actuators. The main characteristics of this approach are the simplicity, modularity and independence of elementary behaviors.

It is this independence, however, that is not always guaranteed when dealing with cooperative behaviors. For example, consider two simple behaviors: one of avoiding obstacles, the other of tracking a target. In both cases, different perceptual stimuli control the same actuators, i.e. the steering wheel. The problem arises when the two stimuli direct the motion in opposite directions! The cooperative process turns here into a prioritization process in which one behavior must give priority to the other in order to avoid conflict. Hence, the cooperative process must turn into an arbiter or supervisor that properly schedules the sequence of behavior execution (i.e. enables and disables the right behavior at the right time). There are many instances of this kind of conflict whenever two or more perceptual events direct the same actuator or actuators in conflicting directions, such as reaching for an object while keeping the body balanced.

3.5. Cooperative agents

Here we shall consider autonomous agents that are not physically coupled, for the simple reason that if agents are physically coupled then it is a case of cooperative manipulation as discussed in Section 3.3. There are several issues one must address: communication, autonomy, and cooperative task specification. We shall discuss them in turn.

3.5.1. Communication

In biological systems, it is well known that animals can communicate many different ways. We shall not, however, review biological systems, but rather take an engineering point of view in the hope that it will give insight into one plausible solution to this problem. For communication to occur there must be a sender and a receiver. In principle we can distinguish between an active and passive sender/source. In the case of an active source, the agent sends messages with given meanings (e.g. alarm, hunger, sexual desire, etc.); this act is under its control. In the case of a passive source, the agent does not send any messages except indirectly, i.e. by existing in an environment in which other agents can detect its existence. The communication channels cover the whole spectrum of sensory capabilities, from smell, touch, temperature, sonar/audio to vision. The open problems in the communication of distributed networking systems are throughput, bandwidth, delays, packet losses, and congestion. The common term encompassing all these parameters is *quality of service*. The question is, for a given application, how to guarantee the quality of service such that it will minimize communication in the distributed robotic system and thereby enable the system to perform the given task. The new challenge in robotic applications is guaranteeing quality of service in order to meet the stringent requirements for delivery of sensory feedback and force feedback in particular (Nahrstedt and Smith 1995). These issues are discussed in many variations in (Asama et al. 1994).

3.5.2. Autonomy

In our definition, autonomy is directly proportional to the number and variety of elementary behaviors available to an agent. Remember that an elementary behavior is

defined as the connection between perception and action. It is instructional to consider the extreme cases. One extreme is when an agent has no autonomy whatsoever; it receives all inputs as commands from another agent. In this case, the agent's communication is only one sided, i.e. the agent only receives messages and has no control over them. The other extreme is when an agent is completely autonomous; it has complete control over its input and output messages and hence acts independently. This agent, with such a degree of autonomy, typically does not cooperate with other agents except by accident. Unlike either extreme case, the cooperative agent will give up some of its autonomy for the good of the group!

3.5.3. Cooperative task specification language

The problem at hand is how to encode and/or communicate to the agents the cooperative task. Again, it is instructive to consider extreme cases. One extreme is when cooperation is defined or specified through a common goal. This is the situation, for example, when agents are told meet at a common location. Here, each agent is autonomous regarding how it carries out the task, as long as it reaches the *goal* state. The other extreme is when agents are coupled very tightly and must share every step along the way. The language that describes these cooperative tasks must be expressive enough to encode the degrees of coupling. We have so far observed that most systems do not possess this expressiveness in the task description language, rather the degree of coupling is encoded in the supervisor or coordinating architecture. This is a challenge for the future.

4. MODELING AGENTS AND THEIR BEHAVIORS

In the previous section we discussed cooperative processes as they occur at different levels. We also defined what we mean by behavior and considered what problems may arise when one has two simultaneous behaviors, even though cooperating, with conflicting actions. In order to make some predictions, we need a theory of behaviors. What, then, is the problem? The connection of perception to action in its simplest form is the classical control theoretic loop. The typical example is the standard automatic control of temperature or pressure, that is of a single parameter (scalar or vector) control. The mathematical model is a linear differential first order, or sometimes second order, equation. In the context of robotics, the typical scenario is motion control, the mechanism being either a manipulator or a vehicle.

The motion equations are derived from Lagrangian mechanics, which formulates the problem as follows: the Lagrangian $L = K - P$, where K is the kinetic energy and P is the potential energy of the system. The kinetic and potential energy of the system may be expressed in *any* convenient coordinate system, not necessarily in Cartesian coordinates. We call this system *generalized coordinates*, denoted by q_i . In his classic text, Paul (1981) gives a detailed derivation of the motion equations for a robotic manipulator. The dynamics equations relate forces and torques to position, velocities and acceleration. These equations represent six coupled non-linear differential equations which are impossible to solve except in the most trivial cases. We are interested in *effective inertias* at each joint and the inertial couplings between them and the acceleration at a joint, the relationship between torque and acceleration at a joint, and the relationship between torque at a joint and accelerations at other joints. If the coupling inertias are small with regard to

the effective joint inertias then we can treat the manipulator as a series of independent mechanical systems. This is the goal!

The goal, then, of the designer and/or modeler of an agent is to decouple individual mechanism/actuators and their corresponding sensors in such a way that they can be described, preferably by linear transformation or simple computable functions. As Paul (1981) points out, a multilink manipulator can be considered as a series of weakly coupled independent servo mechanisms. The coupling effects are reduced by the actuator inertia and by ensuring that the outer joints have greater servo bandwidth than the inner joints. The servo bandwidth is determined by friction and gravity load effects. In order to compensate for gravity loading and to calculate the effective inertia of the joints, the mass and moments of any load must be known. This, of course, is possible by a wrist force sensor that measures all six forces and torques. Thus, given the number of links, one can model and control the dynamics motion of such manipulators, including rigid load, in one continuous fashion.

However, the moment the manipulator is connected to another nonrigid object, such as another manipulator, the whole model *changes* and subsequently the control changes! Hence, we come to the problem of how to structure the agent-environment, where the environment includes different agents, so that one can systematically synthesize a model and controller of the dynamic motions of agents. This is the classical dilemma of the division between *continuous* models versus *discrete* models. In Artificial Intelligence terminology, this is the problem of *signal-to-symbol* transformation and vice versa.

Why do we need discrete, symbolic representations? Symbols are not only a simplified shorthand of the signal for communication purposes, but they are also economic for representation purposes. They are necessary as part of a language that describes the task and in general enables communication amongst agents and humans. The open question still remains, however, of determining at which level of the signal one should attach the labels/symbols.

Traditionally, AI researchers have described the agent by an agent function from percept sequences to actions (Agre and Chapman 1987, Geneserth and Nilsson 1987, Brooks 1986). This is coupled with Rosenschein and Kaelbling's theory of situated automata (1986) and the indexical-functional aspects of the reactive planner *Pengi* proposed by Agre and Chapman (1987). All these approaches partition perception-action (behaviors) with respect to situations that are recognized based on perceptual observations. Indexicalization is a winner in terms of space only if the number of situation referents and object referents (indexical-functional aspects) is much smaller than the total number of situations and objects described in the theory T in the situation calculus (Subramanian and Woodfill 1989). In fact, this is at the heart of our question. In the past we modeled elementary behaviors, such as the behaviors GOTO (with some given heading and assuming a free path) and AVOID obstacles for a mobile base (Košecká and Bajcsy 1994). This approach, however, is too granular! We now have a *fundamental process* for a mobile base that includes three different control laws or strategies:

1. **GoTo**. This control strategy is implemented by a procedure that, given a goal and information about obstacles in the vicinity, generates the desired linear and turning velocity commands.

2. **GoToHead**. This basic strategy is a pure rotation in order for the mobile platform to reach a desired heading. This mode is needed since the **GoTo** strategy cannot guarantee the desired heading of the mobile base, due to the fact that final goal configuration is specified only in terms of position and the orientation of the mobile base must obey the nonholonomic constraints.
3. **GoToMarch**. This control law generates commands for the mobile base while marching in parallel formation with another base and keeping the distance between them constant.

The fundamental process is modeled in the language of Discrete Event Systems, where each of the strategies is a different state (in addition to the states **Initialize** and **Wait/Ready**).

The question before us now is this: should the fundamental process be one continuous dynamic model? That is, should behaviors be partitioned based on the dynamics of the system? It is the dynamics motion equations that change as the mechanisms' degrees of freedom change. We can envision a supervisor that recognizes the new couplings and automatically generates a new dynamics model of motion and its control. In this way, we have for the same structural mechanism (manipulator or vehicle or mobile mechanism) the same dynamic equations of motion provided that the degrees of freedom do not change. Even if the load changes, as long as it is rigid and one can measure the forces and torques the same equations hold for their control. One can make similar arguments for mobile agents. Yamamoto (1994) models the dynamics of a mobile agent/vehicle with three degrees of freedom and derives both holonomic and non-holonomic constraints. His dynamic equations reflect the masses of the vehicle and inertial forces. This model is adequate if the vehicle moves on a flat surface, i.e. one assumes there are no rolling or slipping forces between the vehicle and the road. On the other hand if the road surface is inclined, then the dynamical system again changes. (Intuitively, if one drives a vehicle then one shifts gears in this situation!) Similarly, the tractor trailer configuration imposes a new dynamical model (DeSantos 1994), as does a unicycle (Sheng and Yamafuji 1994).

Hence, we have established that for any change in the degrees of freedom of the agent/environment interaction, the dynamical model changes. This is reflected in the dynamical equation. Consider that the agent interacts with its environment through visual stimulus. Schönner (1991) studies the perception-action cycle under the following assumptions:

1. The state of the postural control system can be described by position x of the eye measured in a forward-backward direction.
2. Without vision (i.e. eyes closed), the posture control system generates a fixed point $x = 0$ (by the choice of the coordinate system); the dynamics of x without vision is the *intrinsic* dynamics and is assumed to be a second order linear system.
3. The visual information couples additively into the dynamics through the *expansion* rate $e(x, t)$ of the visual surround.

Expressed mathematically:

$$\ddot{x} + \alpha\dot{x} + \omega_0^2 - \sqrt{Q_x}\xi_t = -c_{env}e(x, t), \quad (1)$$

where the left hand side of the equation is the *intrinsic* dynamics, i.e. a linear damped harmonic oscillator. After Dijkstra (1994), the expansion rate can be modeled as:

$$e(x, t) = \frac{\dot{x} - \dot{D}(t)}{x - D(t)},$$

where $D(t)$ is the movement of the visual surround as it projects on the optic array of the subject. The expansion rate depends on the velocity of the subject as well as on the velocity of the surround. Now consider that this optic flow is also modeled by a harmonic oscillator:

$$D(t) = D_0 + D_r \sin(\omega_D t).$$

We can then rewrite the previous equation as:

$$\ddot{x} + \tilde{\alpha}\dot{x} + \omega_0^2 x - \sqrt{Q_x}\xi_t = c' \cos(\omega_D t),$$

which has an asymptotic solution:

$$x(t) = r_0 \sin(\omega_D t + \phi_0).$$

The dynamics of the perception-action cycle can be studied more generally by investigating the function:

$$x(t) = r(t) \sin(\omega_D t + \phi(t)).$$

The important component here is the relative *phase* between the posture sway and the visual motion. Transforming Equation 1 into polar coordinates and using an averaging method we obtain:

$$\dot{\phi} = \tilde{a} + \tilde{b} \sin(\phi(t) - \phi_0) + \sqrt{\tilde{Q}}\xi_t. \quad (2)$$

The important parameters are \tilde{a} and \tilde{b} . If $abs(a) < abs(b)$ then these two oscillators are phase locked. Hence, we can speak of *absolute coordination*. If $abs(a) > abs(b)$ then there is no fixed phase relationship, and hence we can speak of *uncoordinated behavior*. Lastly, if $abs(a) \sim abs(b)$ then we have *relative coordination*.

In summary, the model of the agent should be based on its dynamics. We have the following equation:

$$M(q)\ddot{q} + D(q, \dot{q})\dot{q} + Uq + G(q) = -c_{env}e(q, t),$$

where the left side represents the agent's dynamics and contact with its environment and the right side represents the coupling of the agent with the environment via non-contact sensing. If either side of the equation changes, then the *model changes*; hence, we have a *new symbol* and/or *new state*.

Now we can offer a new definition of behavior: behavior is the *harmony/coordination*, or *lack* thereof, of an agent with its environment modulo the given *task*. If an agent and its environment filtered by the task can each be modeled as an *active* (non-linear) oscillator, then the interaction between the agent and its environment in carrying out the task can be measured by the *phase* relationship between these two oscillators. Abrupt changes in the parameters a and b detected by some "edge measure" imply a *new symbol*. Note that in this manner the symbols are *situated* and *embedded*. This new perspective on behavior has, of course, implications on how we can describe the task. We discuss them below.

5. TASK DESCRIPTION LANGUAGE

The idea of representing tasks and plans as networks of processes was originally proposed by Lyons (1989) in his *RS* (Robot Schema) model. The RS model is essentially a robot programming language with the basic unit schema representing a single locus of computation.

What should such a task specification contain? First, the task must specify the agent's capabilities (the left side of Equation 2). Second, the task must specify the environment (the right side of Equation 2). Third, the task must specify the interaction between the agent and its environment, which will imply the granularity or the sensitivity of the perception and action in carrying out the task. The task specification must also recognize the difference in spaces, i.e. the difference between the task or environment space and the agent's space.

6. CONCLUSION

We have put forward the premise that cooperation is a process fundamental to both biological and artificial intelligent systems. In Section 2 we defined the necessary conditions for a cooperative process, namely that agents must share a common task or goal, and agents must share some common knowledge and its representation. We also argued that cooperation takes place on several levels internal to the agent: sensation, manipulation (two-handed), locomotion, and behavior, e.g. when two or more behaviors must cooperate because they share the same resources within the agent. Finally, cooperation also takes place on a level external to the agent, specifically, when there are several agents cooperating in a society of agents.

Why is cooperation so fundamental? The chief reason is that the alternative is chaos or disorder. Disorder leads to inefficient use of resources/energy and ultimately degrades performance and impedes achievement of goals.

In our study of appropriate models of agents and their behaviors (see Section 4), we have concluded that the proper models must include *dynamic* interaction of the agent and its environment, which in turn allows the agent to *actively control* its behaviors and thereby achieve its performance goals. We have shown inductively that the models and their respective control modes are discrete with respect to the degrees of freedom of the system. Furthermore, we have shown that as the system is coupled with its environment it must be in *harmony* with and *cooperate* with its environment in order to perform its task efficiently. Hence, we can extend our definition of cooperation from agent-to-agent cooperative behavior to agent-environment cooperative behavior. This perspective unifies the process of interaction of agents with their environment in the same *cooperative framework*. This insight into models of dynamic interaction of agents and their environment also offers a *systematic* approach to understanding signal-to-symbol transformation anchored in physical principles because each control mode can be viewed as a different symbol. The open question that remains, however, is how to deal with the discretization of the environment that does not follow from dynamic agent interaction yet is useful for *task* description.

We believe that the basis for understanding intelligent agents lies in understanding the representations/models of these agents and their interaction with their environment dur-

ing task execution. This interaction is dynamic and hence must be modeled as such. Different degrees of freedom of the system (agents and environment) imply different discrete models, i.e. symbols. Efficient performance demands that agents and their subsystems (sensors, actuators, and elementary behaviors) cooperate with the environment. Thus, the journey continues — from active perception to active cooperation.

REFERENCES

- Adams, J., R. Bajcsy, J. Košecká, V. Kumar, R. Mandelbaum, M. Mintz, R. Paul, C. C. Wang, Y. Yamamoto and X. Yun, 1995, Cooperative Material Handling by Human and Robotic Agents: Module Development and System Synthesis, in: Proc. International Robotics and Systems Conference, Pittsburgh, PA, Aug. 1995.
- Agre, P., and D. Chapman, 1987, Pengi: An implementation of a theory of activity, in: Proc. of the AAAI, (Lawrence Erlbaum, Hillsdale, NJ), 1987.
- Allen, P.K., 1987, Robotic Object Recognition Using Vision and Touch (Kluwer).
- Aloimonos, Y., 1993, Active Perception (Lawrence Erlbaum, Hillsdale, NJ).
- Aloimonos, Y., I. Weiss and A. Bandyopadhyay, 1987, Active vision, in: Proc. DARPA Image Understanding Workshop, p. 552-573.
- Arai, T., and H. Osumi, 1991, Heavy Work Handling by the Cooperative Control of a Crane and a Robot, in: Jrnl. Japan Society of Precision Engineering, **57**, 3, p. 467.
- Arbib, M.A., and D.H. House, 1987, Depth and Detours: An Essay on Visually Guided Behavior, in: Vision, Brain and Cooperative Computation (MIT Press, Cambridge, MA).
- Arkin, R.C., 1987, Motor schema based navigation for a mobile robot, in: Proc. IEEE International Conference on Robotics and Automation.
- Asama, H., T. Fukuda, T. Arai and I. Endo, 1994, Cooperative Operation, in: Distributed Autonomous Robotic Systems (Springer-Verlag, Tokyo).
- Bajcsy, R., 1985, Active perception vs. passive perception, in: Proc. 3rd Workshop on Computer Vision: Representation and Control, Bellair, MI, Oct. 1985, p. 55-59.
- Bajcsy, R., 1988, Active perception, in: Proc. of the IEEE, **76**, 8, p. 996-1005.
- Bajcsy, R., V. Kumar, M. Mintz, R. Paul and X. Yun, 1992, A small-team architecture for multiagent robotic systems, in: Workshop on Intelligent Robotic Systems, Design and Applications, SPIE Intelligent Robotics Symposium, Boston, MA, Nov. 1992.
- Ballard, D.H., 1991, Animate vision, in: Artificial Intelligence, **48**, 1, p. 57-86.
- Beer, R.D., H.J. Chiel and L.S. Sterling, 1990, A biological perspective on autonomous agent design, in: Designing Autonomous Agents, ed. Pattie Maes (MIT Press, Cambridge, MA).
- Bejczy, A.K., T.J. Tarn and X. Yun, 1986, Coordinated Control of Two Robot Arms, in: Proc. IEEE International Conference on Robotics and Automation, San Francisco, CA, Apr. 1986.
- Blake, A., and A. Yuille, eds., 1992, Active Vision (MIT Press, Cambridge, MA).
- Bogoni, L., 1995, Functionality Recovery through Observations and Interaction, Dissertation Proposal, GRASP Laboratory, Computer and Information Science Dept., Univ. of Pennsylvania, Philadelphia, PA.
- Bogoni, L., and R. Bajcsy, 1994, Active Investigation of Functionality, in: Proc. Workshop on the Role of Functionality in Object Recognition, IEEE Computer Vision and Pattern Recognition Conference, Seattle, WA, June 1994.

- Braitenberg, V., 1984, *Vehicles, Experiments in Synthetic Psychology*, (MIT Press, Cambridge, MA).
- Brauer, W., and D. Hernandez, eds., 1991, *Distributed AI and Cooperative Working*, Proc. 4th International GI-Congress (Springer-Verlag).
- Brooks, R.A., 1986, A robust layered control system for a mobile robot, in: *IEEE Journal of Robotics and Automation*, **2**, 1, p. 14-23.
- Brooks, R.A., 1991, Intelligence without reason, in: A.I. Memo, Apr. 1991.
- Dauchez, P., and M. Uchiyama, 1987, Kinematic Formulation for Two Force Controlled Cooperative Robots, in: 3rd International Conference of Advanced Robotics, Versailles, France, Oct. 1987.
- DeSantos, R.M., 1994, Path-Tracking for a Tractor-Trailer like Robot, in: *International Journal of Robotics Research*, **13**, 6, Dec. 1994, p. 533-544.
- Dijkstra, T., 1994, *Visual Control of Posture and Visual Perception of Shape*, Ph.D. dissertation, Katholieke Universiteit Nijmegen, College van Decannen, CIP-Gegevens Koninklijke Bibliotheek, Den Haag, Netherlands.
- Draper, B.A., R.T. Collins, J. Brolio, A.R. Hansen and E.M. Riseman, 1989, The Schema System, in: *International Journal of Computer Vision*, **2**, 3, Jan. 1989.
- Gasser, L., and M.N. Huhns, eds., 1989, *Distributed AI* (Pitman/Morgan Kaufman).
- Geneserth, M.R., and N.J. Nilsson, 1987, *Logical Foundations of Artificial Intelligence* (Morgan Kaufman).
- Gibson, J.J., 1950, *The Perception of the Visual World* (Houghton Mifflin, Boston, MA).
- Guckenheimer, J., and P. Holmes, 1982, *Nonlinear Oscillations, Dynamical Systems and Bifurcations of Vector Fields* (Springer-Verlag, New York).
- Hashimoto, K., ed., 1993, *Visual Servoing: Real Time Control of Robot Manipulators Based on Visual Sensory Feedback*, World Scientific Series in Robotics and Automated Systems, **7** (World Scientific Press, Singapore).
- von Holst, E., 1937/1973, On the nature of order in the nervous system, in: *The behavioral physiology of animals and man*, R. Martin, trans., **1** (Methuen, London).
- Hooman, J., 1992, Top-Down Design of Embedded Real Time Systems, in: *IFAC AI in Real-Time Control*, Delft, The Netherlands, p. 453-458.
- Huhns, M., ed., 1990, Proc. 10th Workshop on Distributed AI, Technical Report ACT-AI-355-90, MCC, Austin, TX.
- IJCAI (International Joint Conference on Artificial Intelligence), 1991, Workshop on Intelligent and Cooperative Information Systems, Sydney, Australia, Aug. 1991.
- Jeannerod, M., 1994, The representing brain: Neural correlates of motor intention and imagery, in: *Behavioral and Brain Sciences*, **17**, p. 187-245.
- Khatib, O., 1995, Inertial Properties in Robotic Manipulation: An Object Level Framework, in: *International Journal of Robotics Research*, **14**, 1, Feb. 1995, p. 19-36.
- Košecák, J., and R. Bajcsy, 1994, Discrete Event Systems for Autonomous Mobile Agents, in: *Journal of Robotics and Autonomous Systems*, **12**, p. 187-198.
- Krotkov, E.P., 1989, *Active Computer Vision by Cooperative Focus and Stereo* (Springer-Verlag).
- Levy, S., 1993, *Artificial Life* (Pantheon Books).
- Lyons, D.M., 1989, A formal model of computation for sensory-based robotics, in: *IEEE Transactions of Robotics and Automation*, **5**, 3, p. 280-293.

- Mandelbaum, R., 1995, Sensor Processing for Mobile Robot Localization, Exploration and Navigation, Dissertation Proposal, GRASP Laboratory, Computer and Information Science Dept., Univ. of Pennsylvania, Philadelphia, PA.
- Mandelbaum, R., and M. Mintz, 1994, Sonar signal processing using tangent clusters, in: Proc. OCEANS'94, Special session on automated unmanned vehicles, **2**, Brest, France, Sept. 1994, p. 544-549.
- von Martial, F., 1992, Coordinating Plans of Autonomous Agents, Lecture Notes in AI (Springer-Verlag).
- Moravec, H.P., 1983, The Stanford cart and the CMU rover, in: Proc. of the IEEE, **71**, p. 872-884.
- Nahrstedt, K., and J.M. Smith, 1995, The QoS Broker, to appear in: IEEE Multimedia Magazine.
- Nilsson, N.J., 1984, Shakey the robot, Technical Report 323, SRI AI Center.
- Paul, R.P., 1981, Robot Manipulators – Mathematics, Programming and Control (MIT Press, Cambridge, MA).
- Peterson, J.L., and A. Silberschatz, 1985, Operating systems (Addison Wesley).
- Reynolds, C.W., 1987, Flocks, herds and schools: A distributed behavioral model, in: SIGGRAPH, 1987, p. 25-33.
- Rosenschein, S.J., and L.P. Kaelbling, 1986, The synthesis of machines with provably epistemic properties, in: Proc. Conference on Theoretical Aspects of Reasoning about Knowledge, 1986.
- Schöner, G., 1991, Dynamic theory of action-perception patterns: the “moving room” paradigm, in: Biological Cybernetics, **64**, p. 455-462.
- Sheng, Z., and K. Yamafuji, 1994, Stability and Motion Control of a Unicycle, in: Journal of Robotics and Mechanotrics, **6**, 2, p. 175-182.
- Stansfield, S., 1986, Primitives, Features and Exploratory Features: Building a Robot Tactile Perception System, in: Proc. IEEE International Conference on Robotics and Automation, Apr. 1986.
- Subramanian, D., and J. Woodfill, 1989, Making Situation Calculus Indexical, in: Proc. of Knowledge Representation, 1989.
- Sun, J., T. Nagato and K. Kurosu, 1994, Cooperative Behavior of a Schedule-Based Distributed Autonomous Robotic System, in: Journal of Robotics and Mechanotrics, **6**, 2, p. 162-168.
- Tarn, T.J., A.K. Bejczy and X. Yun, 1987a, Nonlinear Feedback Control of Multiple Robot Arms, in: JPL 1987 Telerobotics Workshop, Jet Propulsion Laboratory, Pasadena, CA.
- Tarn, T.J., A.K. Bejczy and X. Yun, 1987b, Design of Dynamic Control of Two Cooperating Robot Arms: Closed Chain Formulation, in: CRA87, Raleigh, NC, Mar. 1987, p. 7-13.
- Toro, M., and J. Aracil, 1988, Qualitative Analysis of System Dynamic Ecological Models, in: System Dynamic Review, **4**, p. 56-60.
- Wilson, E.O., 1971, The Insect Societies (The Belknap Press of Harvard Univ. Press, Cambridge, MA).
- Yamamoto, Y., 1994, Control and Coordination of Locomotion and Manipulation of a Wheeled Mobile Manipulator, Ph.D. thesis MS-CIS-94-39, GRASP Laboratory, Computer and Information Science Dept., Univ. of Pennsylvania, Philadelphia, PA.