INTERNATIONAL ENCYCLOPEDIA OF ROBOTICS

Applications and Automation

Volume 2

RICHARD C. DORF

INTERNATIONAL ENCYCLOPEDIA OF ROBOTICS: APPLICATIONS AND AUTOMATION

VOLUME 2

Richard C. Dorf, *Editor-in-Chief*

Shimon Y. Nof, *Consulting Editor*

WILEY

A Wiley-Interscience Publication

John Wiley & Sons

New York / Chichester / Brisbane / Toronto / Singapore

Library of Congress Cataloging in Publication Data:

International encyclopedia of robotics.

"A Wiley-Interscience publication."
1. Robotics—Dictionaries. I. Dorf, Richard C.
II. Nof, Shimon Y., 1946–
TJ210.4.I57 1988 629.8'92'0321 87–37264
ISBN 0–471–87868–5 (set)
ISBN 0–471–63513–8 (vol. 2)

Printed in the United States of America

10 9 8 7 6 5 4 3

INFORMATION FRAMEWORK FOR ROBOT DESIGN

STEVEN H. KIM
Massachusetts Institute of Technology
Cambridge, Massachusetts

INTRODUCTION

A robot must make judicious use of the information available from the environment to exhibit intelligent behavior in the pursuit of its objectives. To this end, it is imperative to address the design of information acquisition and utilization strategies in a coherent, rational fashion. This article describes a general framework for considering information issues in a systematic way. The framework exhibits the following characteristics:

1. A two-dimensional structure that partitions information issues by arena into internal and external factors, and by attribute into effectiveness, efficiency and timeliness.
2. A multimodal architecture to accommodate the hierarchical decomposition of functional requirements and design parameters into a series of sublayers and subtrees.
3. A generalized measure of information based on the probability of attaining a set of functional requirements.

This information model may be used to study the implications for the design of robotic systems in different classes of applications, ranging from mobile robots to automated process control. A quantitative measure is based on the concepts of classical information theory and on the interpretation of information in the context of purposive behavior (1–3). The following discussion presents the relationship between purpose and information, interprets some critical information attributes within the framework, and illustrates the application of these concepts to specific examples (see also CYBERNETICS, DESIGN AND MODELING CONCEPTS, LEARNING AND ADAPTATION).

The last section of this article discusses two generic strategies for managing complexity in the design and operation of robots. These decomposition techniques relate to the hierarchical and layered strategies.

TWO-DIMENSIONAL FRAMEWORK FOR INFORMATION

The information needs and uses for a robot may be partitioned into a two-dimensional framework in terms of arena and attribute. The first dimension, *arena,* refers to the spatial realm of information. Its major components are the internal and external realms. Each of these components may, in turn, be subdivided to arbitrary levels of specificity for purposes of design. To illustrate, the internal arena may be partitioned into reasoning, sensing and manipulation modules, among others. The external arena may be categorized, for example, by activity into animate versus inanimate objects, or by distance to near, mediate or far.

The second dimension, *attribute,* is a set of characteristics which determine the utility of information in relation to its referent functional requirements. This parameter is, in turn, a three-dimensional quantity consisting of the effectiveness, efficiency and timeliness of information. These traits determine the utility of a block of information depending on the particular application area and its impact on the functional requirements. Although the three attributes are conceptually separable, they tend to affect each other in determining the level of usefulness of an information item.

The two-dimensional framework is depicted in Table 1, which includes examples of relevant information issues in each major category. The diagram indicates, for example, that the fulfillment of overall functional requirements determines the effectiveness of information relating to the external sphere. In a similar way, parsimony in monitoring proprioceptive activities pertains to the efficiency of information in the internal arena. The following discusses these information characteristics in greater detail.

ARENAS OF INFORMATION

The information arenas refer to the physical sources and destinations of information quantities. The two components, internal and external information domains, are interrelated; for example, improved internal processing may compensate for sparse information from the environment.

To be effective, a robot must adapt to external changes. However, only for the most simplified applications is it reasonable to build a robotic reasoning system which can be preprogrammed in anticipation of every potential consequence. Hence a robot must be able to learn from its experience.

The scope or extent of learning in an intelligent system may vary widely. An adaptive technique, for example, may pertain to the simple adjustment of parameters in a set of built-in equations modeling some facet of the external world. On the other hand, the learning may involve a sophisticated strategy to evolve a knowledge base by generating new decision rules and discarding ineffectual ones. In this spectrum of adap-

Table 1. Two-dimensional Information Framework for Robot Design with Referent Examples

	Arena	
Attribute	Internal	External
Effectiveness	Activation of hydraulic subsystem to support mobility	Fulfillment of overall functional requirements
Efficiency	Parsimony in gathering and processing proprioceptive data	Matching information acquisition to pertinent status, eg, maximal for emergency, minimal for quiescence
Timeliness	Compensation for inertial effects in a moving manipulator	Avoidance or interception of a moving object

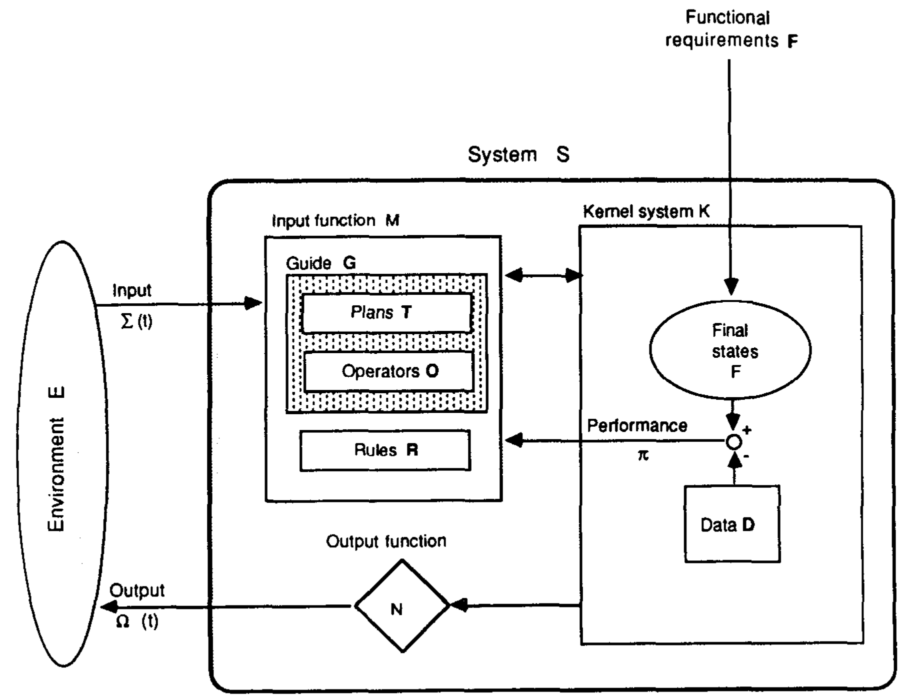

Figure 1. General framework for intelligent robots.

tive behavior, a robot having no learning capability whatever may be considered to be a special limiting case.

External Arena

This section presents a general framework for the structure and operation of intelligent robots incorporating learning or adaptive capability (4–10). The model takes account of the dynamic nature of the environment, and of the adaptive behavior of the system necessary to deal with a complex, changing world in the pursuit of its objectives.

A *robotic system* S is an entity which operates in some *environment* E (see Fig. 1). The robotic system is designed to fulfill a set of *functional requirements* **F.** For a watchdog application, a mobile robot may be instructed to seek out intruders. In a factory application, the functional requirements may relate to the assembly of a product from its components.

To discharge its responsibilities in a dynamic, noise-filled environment, the robot S must make use of information available from its surroundings. This activity may be represented as stream of *input* stimuli Σ. The domain of Σ is a set of physical parameters such as visual or audio intensity.

Conversely, the robot will have some observable impact on its environs. This activity may be represented as an *output* stream Ω.

Internal Arena

The robot S, as an adaptive unit, must incorporate mechanisms which specify its own behavior patterns. Hence S encompasses the *input function* M and the *output function* N, as well as the *kernel* K. The kernel system comprises the set F of final states and some data **D**. The terminal cluster F is specified by the functional requirements **F**, while the data **D** is determined by the messages from the input function **M**. The data component may contain low-level items such as a partial history of the input stream Σ, or high-level information such as the current status in a task planning activity.

The basic component of the input transition function M is the set of transition rules **R**. These rules incorporate domain knowledge for tasks such as factory production operations, and thereby govern the reasoning activities of the robot.

In systems lacking learning capability, the value of a rule is specified in advance by the system developer and remains fixed. In a learning system, however, the relative utility of each rule is a vital item of information which must be determined by the system itself. To this end, the function M includes the *guide* **G**, a metalevel component which serves to evolve the basic decision rules over time.

The guide incorporates the operators and the plan. The *operators* **O** are first-order metalevel rules which act on basic domain rules, defining when a particular domain rule should be activated or discarded. For example, an operator may specify, "If a rule does not work five times in a row, then discard it."

The *plan* **T** is a second-order metalevel component which acts on operators by deciding which operator to invoke at any point in time. The adaptive plan yields a sequence of selections from a set of *operators* **O**. In other words, the plan invokes particular operators to modify structures in **R** and K. An example of an operator is the splicing and merging of 2 rules in **O**; this operator is called a *genetic algorithm* (5,6). The degree to which the functional requirements are met is indexed by a *performance measure* π. If a binary range for π is chosen, for example, then $\pi = 1$ might be used if all the functional requirements are met, and $\pi = 0$ otherwise.

A key factor in learning systems is the determination of effective adaptive plans for differing environmental conditions. For a specific environment E, the input to the plan must contain some measure of the efficacy of the plan in fulfilling the functional requirements **F**. In other words, a component of the data **D** must contain the payoff π which measures the degree to which the functional requirements are met. The selection of an adaptive plan is a nontrivial task. For example, consider the following two plans: one performs superbly in some environments but poorly in others, while the other performs reasonably well in all cases. In this case, the selection of one plan over another should be guided by the particular application.

The final component of the general adaptive framework is the output function N. This function specifies the output stream Ω and determines the external behavior of the system.

The preceding concepts may be clarified through simple examples of automated mechanisms incorporating learning capability.

Example (Mobile Robot). Suppose that a robot R is placed in an amusement park with the purpose of picking up litter such as empty soda cans, or in an automated factory to seek out inoperative machine tools. By rewarding decision rules which lead to successful behavior and penalizing others, **R** will eventually end up with an effective set of rules. This is particularly true when the environment exhibits some orderly—rather than purely random—behavior. For example, suppose that litter is to be found to the east of every pair of blocking objects. The robot will ultimately behave in a way that takes account of this regularity in the environment.

Example (Automated Process Control). The adaptive framework may be applied to production control applications such as polymer processing (see Fig. 2). Although standards exist for polymers, their characteristics vary from lot to lot even when they are produced by a single manufacturer. Because the chemical processes and curing characteristics are not fully understood, final part quality may differ significantly.

Control strategies to date have usually relied on simple open-loop processes. However, a more effective methodology would incorporate the following features: (*1*) closed-loop control, (*2*) realtime sensing and activation of effectors, and (*3*) learning techniques. The adaptive process is designed to take material inputs and produce polymer parts with specific functional requirements **F** relating to thickness H and density ρ. The functional requirements are to be met by design parameters such as the mold temperature T and pressure P.

A FUNCTIONAL MEASURE OF INFORMATION

Purpose (Function) and Information

Systems in isolation tend toward equilibrium and maximum disorder. A living cell, cut off from any other resources, will eventually die, as will an autonomous robot. Hence it can be asserted that any useful object must be constructed with some expenditure of energy. Further, the application of the energy should not be random, since such wanton action may damage the original components. Rather, the energy must be structured; that is, it must be directed through information.

Take as axiomatic the proposition that a purposive system must consist of well-defined components exhibiting nonrandom structure or behavior. Purposive or purposeful behavior is one which serves to attain a goal. A distinction, however, may be made between the two:

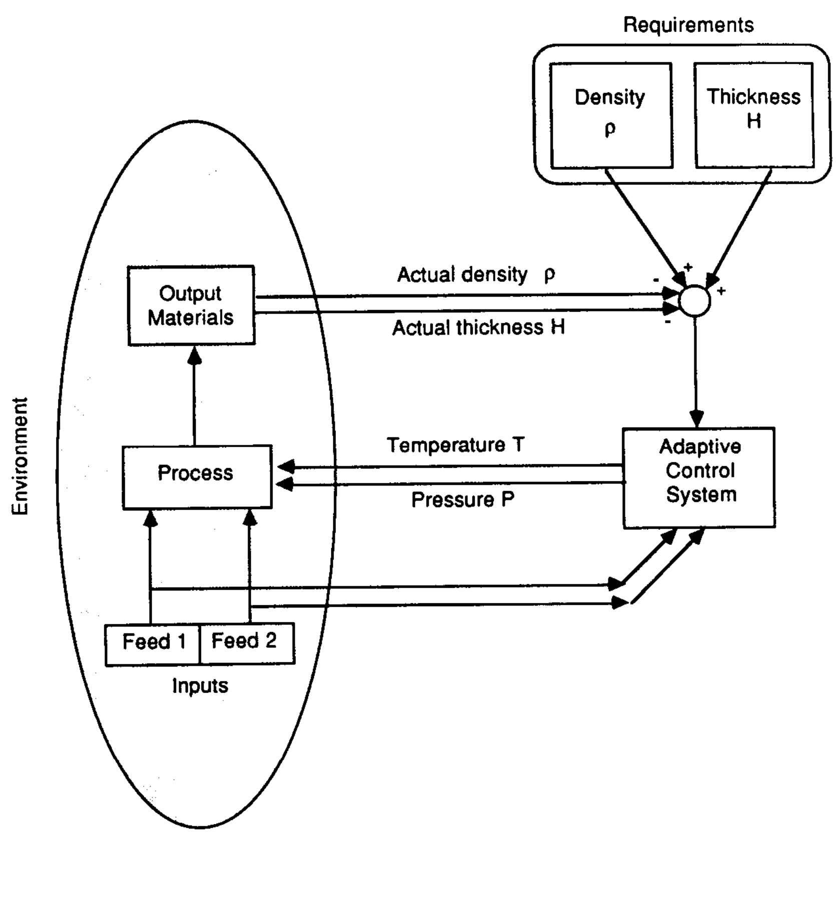

Figure 2. Some critical parameters for polymer processing.

1. *Purposive* behavior pertains to a physical, natural or artificial system whose objectives are assigned or inferred.
2. *Purposeful* behavior applies to a system which can choose its own goals.

A purposeful system can select its own goals and the means with which to pursue them, independently of environmental conditions. Such a system may be said to exhibit will, as exemplified by human beings (11). This article, however, focuses on robots whose functions are assigned by an external human agent.

A purposive system S may be characterized by the following definition.

Definition. An action **a** of a system S has the *purpose, functional requirement,* or *goal* **f** if and only if (iff) the following conditions hold:

1. S may choose to perform action **a**, or not.
2. S has learned that executing **a** results in, or increases the likelihood of, attaining **f**.

Take the view that any artifact which fulfills a set of functional requirements is a purposive object. For example, any robotic system as well as its components, are purposive objects.

One may distinguish knowledge, information and data as follows. Knowledge refers to an awareness of the state of the world or of procedures for attaining goals. Information is a subset of knowledge that pertains only to the state of the world. Data is a quantity that can be transformed into information by the use of knowledge.

Definition. Let S be a system and **F** a set of goals. Then

1. S has *knowledge* K iff K is a fact relating to the state of the world (whether inside or outside S), or if K can be used to attain some goal **f** in **F**.
2. S has *information* I iff I is a type of knowledge pertaining to the state of the world.
3. S has *data* d iff it can use its store of knowledge to transform D into information.

The nonrandom nature of purposive systems may be encapsulated by a quantity called structural information. For example, structural information represents the deviation from randomness in the structure or behavior of a purposive system.

Typology of Purposive Systems

Purposive systems may be identified along one dimension in terms of corporeity: *hardware, software,* or *hybrid,* where the last category refers to an amalgam of the first two. Additionally, purposive systems may be partitioned along a second dimension based on the degree of initiative that they exhibit:

1. *Passive.* The system does not operate on the environment in reponse to input.
2. *Active.* The system operates on the environment in response to input.

These categories are useful for purposes of discussion but may not always have clear-cut boundaries. For example, consider a software package that logs transactions by making changes in a data set or data file. If the package is considered to consist of the algorithmic procedure as well as the data set, then all changes are internal to the system; hence the package is passive. On the other hand, if the data set is part of an autonomous database, then the data set is external to the logging package; consequently the transaction package is an active system. Our primary concern here relates to active, hybrid systems exhibiting intelligent behavior.

A Quantitative Measure of Information

Information accounts for a large fraction of the total cost of intelligent systems such as production plants (12). If information is to be managed judiciously both in the design and operation phases of intelligent systems, then it must be measured properly. Despite the central position that information occupies in engineered systems, relatively little has been written about the nature and characteristics of information as they relate to the fulfillment of functional requirements and system performance.

The definition of information used by communications engineers offers a quantitative measure of information (13,14). This formulation, however, depends strictly on the probabilistic nature of a predefined set of symbols. By focusing only on the statistical properties and ignoring the semantic content of the symbols, one is left with a metric which is of limited use in modeling decision making situations.

The classical definition of information was never intended to capture semantic content or value. A string of bits representing a sales forecast is calculated to have the same "information" content whether received by a production manager, a robot, a donkey, or no one at all. This is somewhat at odds with our everyday notion of "information." For these reasons, the classical notion of information must be modified and/or generalized for use in decision making environments.

In everyday speech, information is associated with the value of a communication. A medical journal is likely to convey more information to a physician than to a carpenter, for example. When one speaks of "value," there is a notion, implicit if not explicit, of value with respect to some purpose or objective. The value of a communication depends on the receiver and the goals of the receiver. A communication that takes a decision maker closer to this goal conveys information; one that takes a person further away conveys disinformation. Since people live in a nondeterministic world, one may express the idea of "moving closer" to a goal by saying "more likely to attain" the goal.

The functional model of information is as follows (1–3). Let a *decision maker* be a purposive entity with a set **F** of goals or functional requirements. Let E be the event that goal **F** is attained, and p the probability that E occurs. Let p_0 be the prior probability before the receipt of a communication C, and p_1 the posterior probability. The information content of C is defined as

$$I = \text{lb}\,(p_1/p_0)$$

where lb denotes the logarithm to base 2. In other words, information is a measure of the probability of success. If a communication reduces the likelihood that E will occur, then $p_1 < p_0$ and therefore $I < 0$. Such a communication conveys disinformation.

When the information equation is written as $I = \text{lb}(1/p_0) - \text{lb}(1/p_1)$, the information is the logarithmic difference in the inverses of probabilities. This interpretation is consistent with the proposition that the information needed by a subsystem to perform a given task is equal to the difference in the information required for the task, versus that which is already available (15).

To acquire a sense for the nature of this metric, it is helpful to consider a number of boundary cases such as the following:

- If a communication does not affect the probability of success, then the information content of the message is 0. This follows from the fact that, if the probability of success is unaffected, then $p_1 = p_0$. Hence $I = \text{lb}(1) = 0$.
- If the probability of attaining the goal state is 1, then no communication can convey positive information. Since $p_0 = 1$, the information content is $I = \text{lb}(p_1) \leq 0$. Here a communication can convey no information at best, and disinformation otherwise.

These concepts may be further illustrated by a simple example in a production application.

Example (Positioning accuracy). Consider a production robot that is to drill a hole in a metal bar. The functional requirement for this task is that the center of the hole should be D in. from a particular edge, with a tolerance $\triangle D$ in. In other words, the hole should be centered in the interval $[D_0, D_1]$ where $D_0 = D - \triangle D$ and $D_1 = D + \triangle D$; this interval may be called the *design range*. However, the robot is able to position the hole in some range $[S_0, S_1]$ which may be called the *system range*. The drilling operation will be successful only if there is some overlap, called the *common range*, between the design and system ranges. The probability of success is then given by (3):

$$P = A/(A + B)$$

where A and B are the respective areas shown in Figure 3. In this figure, the abscissa denotes the position of the hole, and the ordinate the corresponding probability density. When the probability is uniformly distributed, these areas are proportional to the lengths along the dimensional axis. Then:

$$P = \frac{D_1 - S_0}{S_0 - S_0} = \frac{\text{Common range}}{\text{System range}}$$

More specifically, let $D = 900 \pm 2$ in. and the system range be [900 in., 908 in.]. Since the design range is [898 in., 902 in.], the probability of success is

$$P = \frac{902 - 900}{908 - 900} = \frac{1}{4}$$

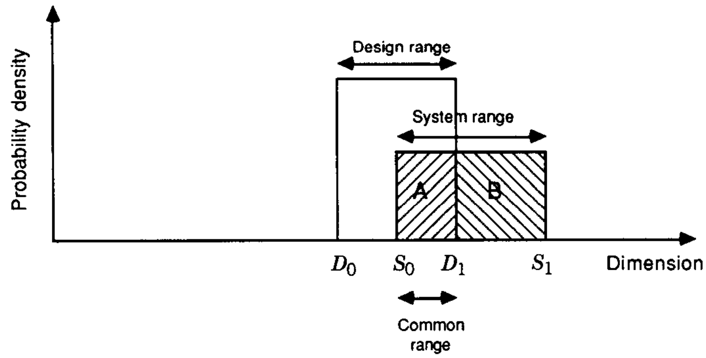

Figure 3. Information for geometric specification.

Then the information is given by $I = \text{lb}(1/P) = \text{lb } 4 = 2$ bits.

ATTRIBUTES OF INFORMATION

Descriptions of physical systems, being idealized models, will often represent reality imprecisely. For example, a physical plant will be subject to disturbances, not all of which can be accounted for in advance (16). Hence our confidence in any component designed to control a physical plant will increase if realtime data on both the model and plant are monitored and reconciled.

Process information would be superfluous in a static situation; the system can be configured at the beginning and trusted to behave as originally implemented. In real situations, however, the system may change (eg, due to wear and tear) as may the environment (eg, as reflected in temperature fluctuations). For these reasons, the system requires information to cope with disturbances.

Even when information about a situation is available, however, it may be misleading by being irrelevant (not effective), incomplete (not sufficient), or delayed (not timely). Usually, however, the controller does not require complete information about a situation in order to make proper decisions. Visual information impinging on the human retina, for example, is aggregated and condensed through several levels until only a small fraction of the original information filters into the visual cortex.

Effectiveness

The *effectiveness* or *criticality* of information may be defined by the elasticity or sensitivity of performance with respect to the information variable. Let π denote a performance index relating to a functional requirement, and I a measure of information. Then the effectiveness of I is given by

$$E = \frac{\partial\pi/\pi}{\partial I/I}$$

where ∂ denotes the partial derivative.

By the property of logarithms, E is equal to $\partial(\log \pi)/\partial(\log I)$. Hence a plot of $\log \pi$ against $\log I$ will highlight the relative effectiveness of different types of information (see Fig. 4).

Let I_0 correspond to the minimal information needed to

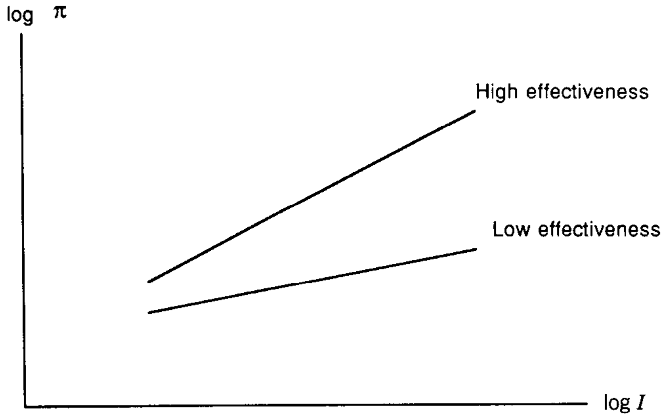

Figure 4. The relative effectiveness of different types of information.

yield the performance level π_0. For large values of I, diminishing returns in π would be expected. Hence the log π versus log I curve will level off.

If the decision-making entity must process the stream of incoming information to sift out the critical components, excessive information input may well decrease performance. In Figure 5, log π begins to decrease for log $I >$ log I_1.

Such a declining curve reflects the phenomenon of *information overload* often observed in biological subsystems. At a higher systemic level, the curve mirrors the proposition that decision makers often receive too much irrelevant information rather than too little in a highly computerized society.

If it is assumed that the acquisition and/or processing of information requires valuable resources, then it may be inferred that a system should avoid information requirements beyond the minimum required to attain its goals. In terms of Figure 5, the principle of information minimization would require that the actual information I' be as close as possible to I_0 without falling below this threshold. This idea is better addressed by the criterion of efficiency.

Efficiency

Efficiency is a measure of the parsimony with which information is conveyed. Let I_0 be the *sufficient,* or minimal, information required to fulfill a particular functional requirement. Then the actual information I' may be less than, equal to, or greater than I_0 (see Fig. 5).

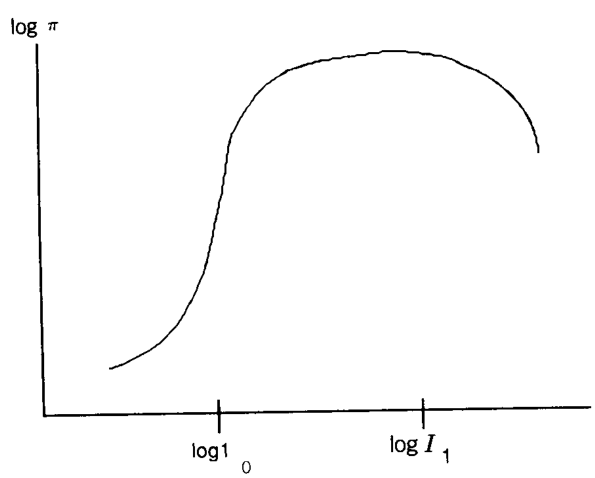

Figure 5. A rising-declining performance curve as a function of information.

1. $I' < I_0$. The actual information I' is insufficient for the task intended. The *relative deficiency* $\delta = (I_0 - I')/I_0$ is a measure of the extent to which the actual information falls short of the amount needed.
2. $I' = I_0$. The actual information is also the sufficient information. In this case the information minimization principle is satisfied.
3. $I' > I_0$. The actual information is in excess of the minimum required. The *efficiency* $\eta = I_0/I'$ is a measure of information parsimony. Its complement, $R = 1 - \eta$, may be called *redundancy*.

By construction all three metrics, δ, E and R, take on values in the half-open interval (0,1]. Efficiency may fall below 1 when redundancy is required in order to guard against the effects of noise or the possiblity of failure in various components.

Timeliness

Information loses value over time due to changes in internal or external factors. An example of an internal disturbance is drift in a system parameter, such as positioning error resulting from mechanical wear. Examples of external disturbances are changes in temperature or vibration levels. Sometimes the external and internal factors combine into constructive or destructive patterns, as illustrated by the phenomenom of resonance.

In a changing situation, a purposive system must react within some time interval τ to adapt to the change. An intelligent lathe, for example, must respond quickly to tool breakage if it is to avoid ruining the workpiece.

Hence the information must be available quickly to allow enough time for the system to respond. The response lag may be due to both software and hardware effects:

1. The decision making unit needs time in which to process the incoming data and make a decision.
2. Due to inertial effects, the system needs time to accommodate itself to the new decision.

One way to categorize time delays is to classify them into different functional phases in terms of sensing, reasoning, and acting stages (1,2,17).

Figure 6 indicates that *sensing* delays occur in extracting information from the environment. The sensor input is utilized by the *reasoning* subsystem, which consists of the analysis and synthesis phases. The *analysis* of data may be decomposed into *reduction, interpretation,* and *comparison* stages. Often, the sampling rates for incoming data are too high to be fully utilized. Moreover, the data is highly redundant both spatially (eg, continuous regions of a single shade of gray for visual input) and temporally (ie, the environment does not change abruptly every fraction of a second). Hence the data must be reduced, through techniques such as runs encoding, moving averages, or moving medians. A technique such as median

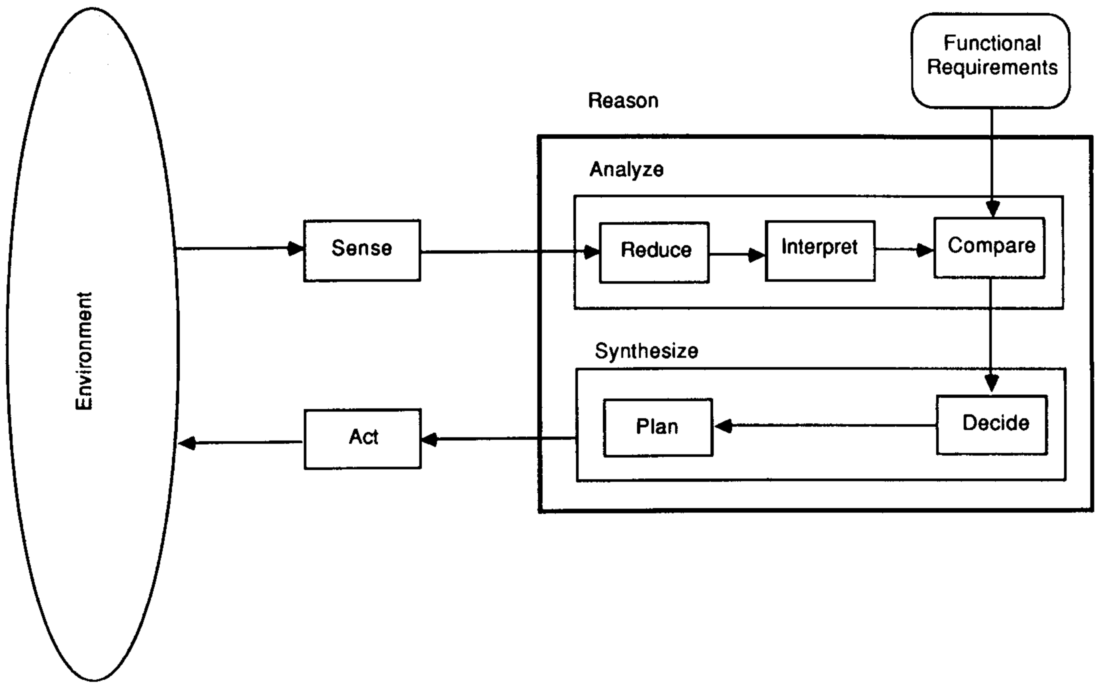

Figure 6. Time delays due to sensing, reasoning, and acting stages.

value extraction is resistant to noise, and will therefore be highly appropriate for certain applications; other techniques will offer differing benefits.

In general, the reduced data must then be combined and interpreted into some higher-level object recognizable by the system. For example, differences in shading may indicate either an edge of a box or a color variation on a single side. The interpreted information must then be compared against the functional requirements to determine how well the system is performing.

The next major stage is *synthesis,* or the formulation of a strategy for action. This stage may be partitioned into the *decision making* and *planning* phases. The *decision* phase involves the selection of a short-term goal (eg, increase the temperature by 5 degrees and lower the humidity by 10 percent). A *plan* of action must then be generated to attain the short-term goal. Finally the plan is implemented in the *acting* stage, which results in an observable change in the system or the environment.

A simple example of a system susceptible to temporal delays is an automated inspection system. The robotic system must first sense the samples arriving on the materials transport system. This information is processed and compared against a canonical state stored in memory. If the match is good, the sample is routed to the "Accept" batch; otherwise it is sent to the "Reject" pile.

The interarrival time between samples constrains the maximum time available for each stage of the inspection function. Moreover, the time available as a sample moves from the sensing area to the "Accept"/"Reject" decision point, defines a window of opportunity for the inspection system. If the robotic system is too slow, it fails to properly discharge its inspection role.

Relationships Among Information Attributes

The attributes of effectiveness, efficiency, and timeliness address the questions of *What, How much,* and *When.* Sufficiency and effectiveness are interrelated. To show this, a performance index π in terms of the associated functional requirements is defined as follows.

Let $\pi_i = 1$ if the i^{th} functional requirement is satisfied, and $\pi_i = 0$ otherwise. If the information for the i^{th} functional requirement is insufficient, for example, then $\pi_i = 0$. Since the functional requirements are independent specifications of the problem, they must all be satisfied in order for the problem to be solved. If there are n functional requirements, then the overall performance index may be defined as

$$\pi = \Pi_{i=1,\ldots,n}\pi_i$$

This formula indicates that $\pi = 1$ if and only if each $\pi_i = 1$, and $\pi = 0$ otherwise.

Sufficiency is defined in terms of fulfilling a set of functional requirements. Hence when information is insufficient, it is ineffective. Conversely, when information is ineffective it makes little sense to speak of sufficiency.

The efficiency and timeliness of information are vital concerns to computer-based systems such as automated factories. For example, the use of microprocessors results in information loss due to temporal lags and the discretization of continuous information. The increasing trend toward computerized control and intelligent devices implies that such concerns will become even more paramount in the years to come.

In short, the performance π of a system is a function of the information attributes of effectiveness, efficiency and timeliness. The particular form of the relationship $\pi = \pi(E, \eta, \tau)$ is determined by the specific application area.

Information and System Performance

Loss of information can result in the degradation of system performance. An extreme example lies in the genesis of instability: a control system of first or second order, which is inherently stable as a continuous-time system, may become unstable as a discrete-time system. One may attribute this metamorphosis to the loss in information resulting from sampling information at discrete intervals rather than continuously.

Another class of instability occurs in various nonlinear elements such as backlash components. Here, the loss in information may be traced to lack of knowledge about the relative displacement of interfacing elements within the dead zone. Obviously, not all nonlinear elements result in instability. An example is found in saturation effects, which yield stable limit cycles.

Example (Sentry Robot). The information requirements for a sentry robot may be readily interpreted in terms of different characteristics. The *effectiveness* of information is related to the robot's ability to perceive an exceptional condition. If the number of smoke particles is too low, for example, then the robot's detectors will be insensitive to the potential fire signal. The effectiveness of the detector rises dramatically in the region of its lower threshold point.

The *efficiency* of each sensor is closely related to its effectiveness. For example, the vision system need not have more pixels than required by the interpretation program under the conditions of lighting and noise present in the environment. Redundancy is employed in order to reach a proper balance between misses and false alarms. For example, tentative detection by infrared sensing may be verified through vision.

In a watchdog application, *timeliness* is an obvious information requirement. To illustrate: the reasoning system must not take so long to interpret a visual image that a burglar can escape; nor should the mobility system be so sluggish as to allow the robot to fall over a precipice even after it has detected the void.

These information characteristics may be interpreted throughout each level of the tree of functional requirements, and between adjacent levels. For example, the sonar system must transmit accurate, relevant information in a timely fashion to enable the reasoning mechanism to direct the mobility system away from obstacles.

DESIGN PHILOSOPHY

A framework for robotic design should satisfy the following functional requirements:

1. Serve as a unifying framework which accommodates hardware and software components.
2. Exhibit a modular organization. This allows for the containment of complexity and ease of reconfiguration in adaptation to changing requirements.
3. Provide an open architecture to accommodate new technologies, whether of hardware or software.

This section presents generic techniques for modularity. The discussion involves a multilevel, multimodal architectural approach which satisfies the functional requirements above.

MODULARITY AND SYSTEM ARCHITECTURE

Two generic techniques for modularization relate to hierarchical and layered organizations (1,2).

Hierarchies

The notion of hierarchies permeates the analysis and synthesis phases of engineered systems, as hierarchies arise in both domains. Hierarchical structures are found, for example, in the organization of industrial plants when analyzing manufacturing systems, as well as in functional trees generated when synthesizing robotic designs.

Figure 7 shows the hierarchy of subsystems pertaining to a mobile robot. The overall system can be partitioned into a set of subsystems ranging from sensing to control modules. In turn each module may be further decomposed; for example,

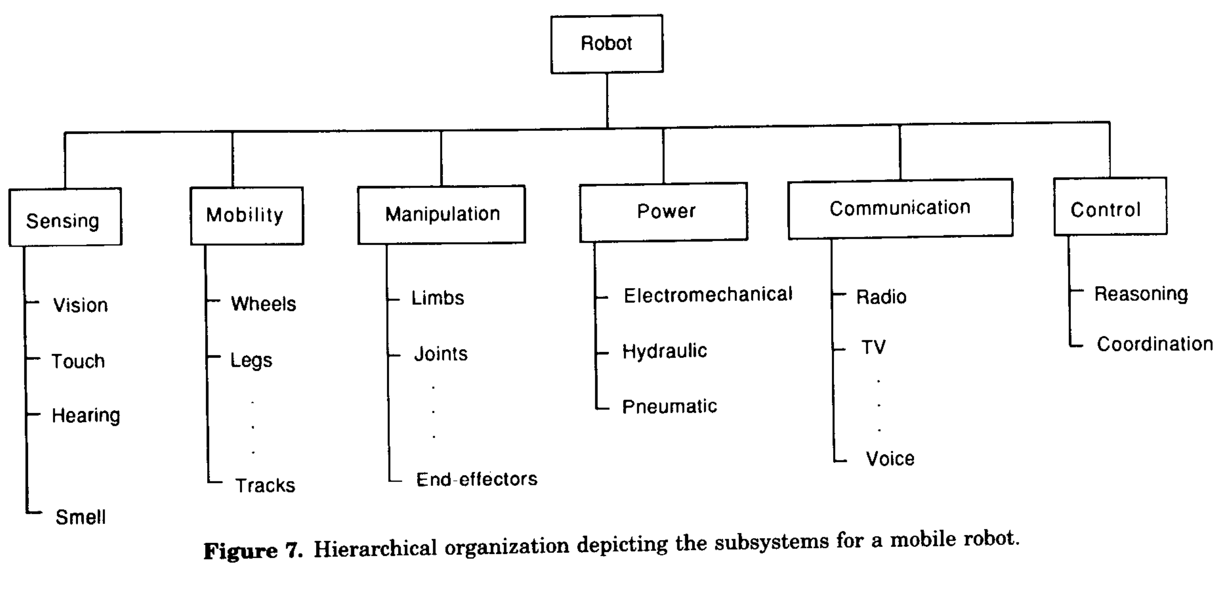

Figure 7. Hierarchical organization depicting the subsystems for a mobile robot.

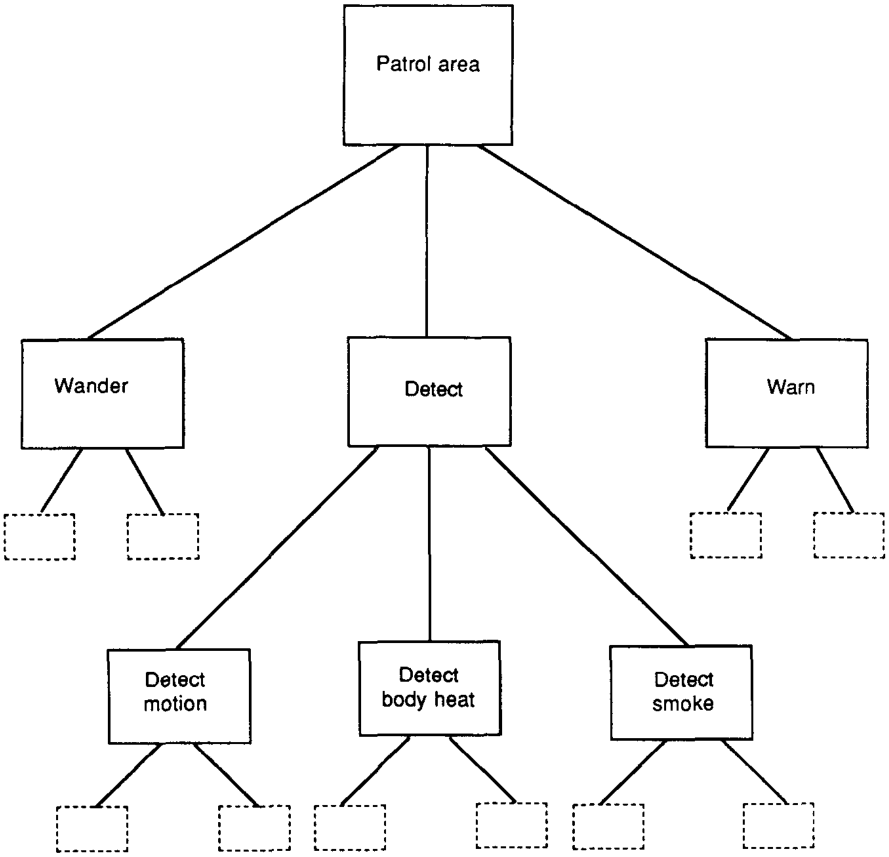

Figure 8. Hierarchical organization depicting the partial tree of functional requirements for a sentry robot.

the sensing subsystem may be partitioned into components for vision, touch, etc.

In a similar way, Figure 8 represents the hierarchical structure of the functional requirements for a mobile robot configured to serve as a sentry. The high-level functional requirement of the robot is to patrol a given territory in order to detect the presence of intruders and other exceptional conditions such as smoke. This function might be partioned into 3 lower-level functional requirements as follows: (*1*) Wander through the security area, (*2*) Detect intruders or other exceptional conditions, and (*3*) Warn a central station of such exceptions. Each of these functional requirements may also be further decomposed. For example, the second functional requirement can be partitioned into these activities: (*1*) Detect motion, (*2*) Detect body heat, and (*3*) Detect smoke.

The hardware architecture may follow a hierarchical configuration which mirrors the functional architecture. Figure 9 depicts the hierarchical structure of physical mechanisms pertaining to the robot. For example, the manipulator unit consists of sensors and activation units coupled with microprocessors for realtime activity control; such units in turn interact with higher-level control units.

Layers

Layering involves the stratification of system components into different slices or levels. A component in one layer can interact only with the adjacent layers, with which it communicates through a standard interface, as shown in Figure 10. Such an interface serves as a gateway which mediates all interlayer communication, and hides the structure of each layer from its neighbors (1,18).

A prime example is found in the levels of computer languages. Consider the progression of languages defined by a machine language M, assembly language A, procedural language P (eg, Fortran or Pascal), and a nonprocedural language N (eg, a program specification language). The sequence <M,A,P,N> represents a total order among the languages, each of which represents a different layer. For example, the statements in a Fortran program need to interact only with each other; the linkages among the corresponding assembly language statements are hidden by the Fortran-to-assembly interface, known as a compiler.

Another example of layering is found in the Open Systems Interconnection Reference Model adopted by the International Standards Organization as a telecommunications protocol. This system consists of 7 levels starting from the physical level pertaining to signal interpretation, and ranging to the applications level dealing with specific end-user packages.

In a similar way, the languages used for robotic control will exhibit a layered structure. The sequence of languages might range from an assembly language for low-level realtime balance control, up to a high-level language for task planning.

A unified architecture for robotic design is found in the explicit two-dimensional configuration consisting of both hierarchies and layers. The *static* facet of the architecture, such as physical equipment or software modules associated with different functional roles, is usefully envisioned in terms of a

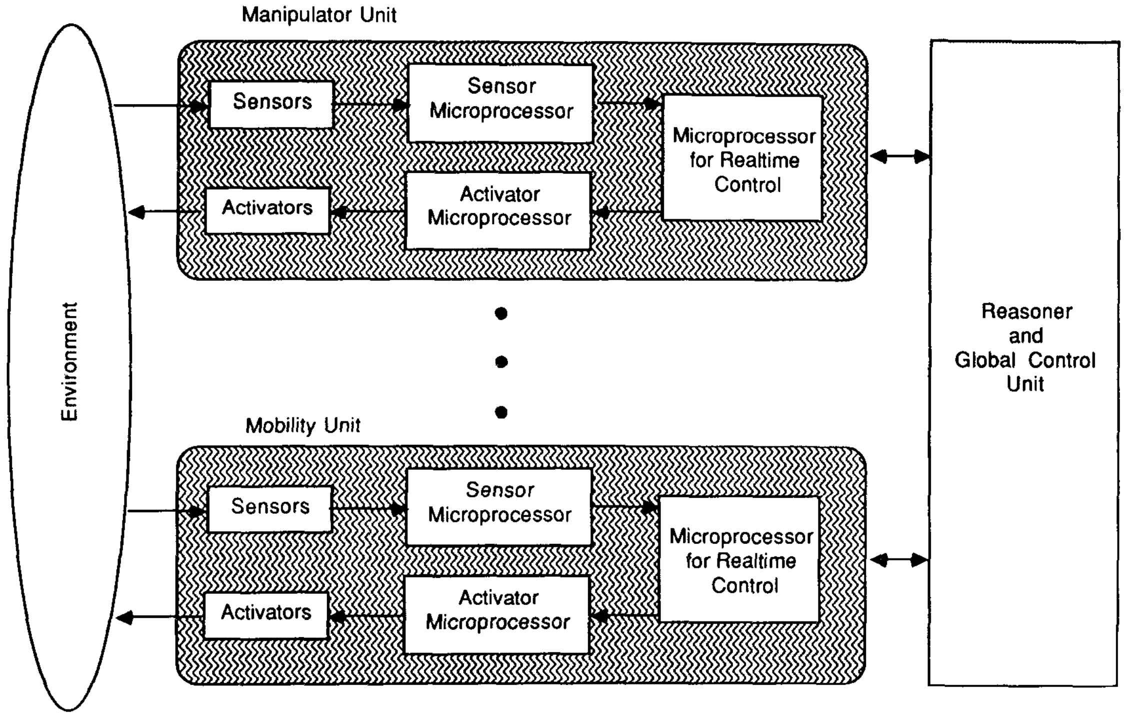

Figure 9. Hardware configuration.

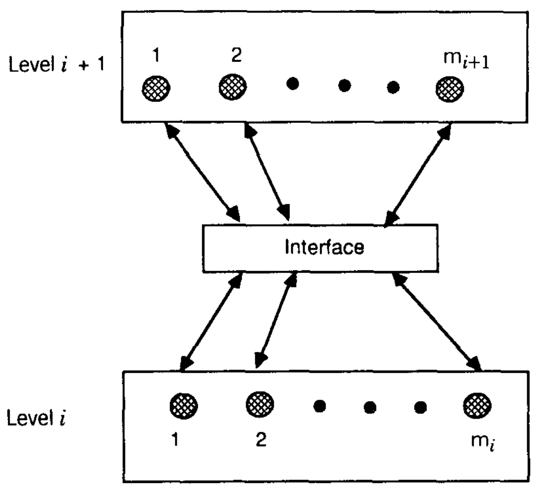

Figure 10. Adjacent levels L_i and L_{i+1} for a layered configuration.

hierarchical organization. This view is complemented by the *dynamic* aspect of software interactions, which is usefully implemented in a layered configuration.

CONCLUSION

This article has presented a generic information framework for robotic design. The framework involves two dimensions relating to arena and attribute. The arena refers to the physical location of a source or use of information; at the highest level of partitioning, this parameter may be subdivided into internal versus external realms. Critical characteristics define the quality of a packet of information; these parameters consist of effectiveness, efficiency, and timeliness. Information attributes may be interpreted in quantitative form in terms of the likelihood of attaining a set of functional requirements. Complexity in robotic design can be managed by employing generic decomposition strategies. The two approaches are the hierarchical and layered architectures.

These concepts are applicable to diverse aspects of robotic design, ranging from the construction of mobile robots to stationary production machinery. The systematic consideration of information issues is key to the design of effective robots, a class of machines which will become increasingly prevalent and significant in the decades ahead.

BIBLIOGRAPHY

1. S. H. Kim, Mathematical Foundations of Manufacturing Science: Theory and Implications. Ph.D. thesis, MIT, Cambridge, Mass., May 1985.
2. S. H. Kim, A Generalized Information-theoretic Framework for Intelligent Systems, Technical Report, Lab. for Manufacturing and Productivity, MIT, Cambridge, Mass., 1987.
3. H. Nakazawa, and N. P. Suh, Process Planning Based on Information Concept, *Robotics and Computer-Integrated Manufacturing* 1(1), 115–123, (1984).
4. D. E. Goldberg, Computer-aided Gas Pipeline Operation Using Genetic Algorithms and Rule Learning, Ph.D. thesis, University of Michigan, 1983.
5. J. H. Holland, *Adaptation in Natural and Artificial Systems,* University of Michigan Press, Ann Arbor, Mich., 1975.
6. J. H. Holland and J. S. Reitman. Cognitive Systems Based on Adaptive Algorithms, in D. A. Waterman and F. Hayes-Roth, eds., *Pattern-Directed Inference Systems,* Academic Press, New York 1978, pp. 313–329.

7. S. H. Kim, A Mathematical Framework for Intelligent Manufacturing Systems. *Proc. Sympoisum on Integrated and Intelligent Manufacturing Systems,* ASME, Anaheim, Calif., Dec. 1986.
8. S. H. Kim, A Unified Framework for Self-Learning Systems, Technical Report, Lab. for Manufacturing and Productivity, MIT, Cambridge, Mass., 1987.
9. S. H. Kim, An Automata-theoretic Framework for Intelligent Systems. Technical Report, Lab. for Manufacturing and Productivity, MIT, Cambridge, Mass., 1987.
10. D. Lenat, EURISKO: a Program that Learns New Heuristics and Design Concepts: the Nature of Heuristics, III: Program Design and Results, *Artificial Intelligence* **21**(2), 61–98 (1983).
11. R. L. Ackoff, and R. L. Emery, *On Purposeful Systems,* Travistock, London, 1972, p. 31.
12. N. P. Suh, "The Future of the Factory," *Robotics and Computer-Integrated Manufacturing.* **1**(1), 39–49 (1984).
13. C. E. Shannon, and W. Weaver, *The Mathematical Theory of Communication,* University of Illinois, Urbana, Ill., 1949.
14. M. J. Usher, *Information Theory for Information Technologists,* Macmillan, London, and Scholium, Great Neck, N.Y., 1984.
15. J. Galbraith, *Designing Complex Organizations,* Addison-Wesley, Reading, Mass., 1973.
16. Y. Z. Tsypkin, *Adaptation and Learning in Automatic Systems,* trans. by Z. J. Nikolic, Academic Press, New York, 1971.
17. J. S. Lawson, Jr., "The Role of Time in a Command Control System," in M. Athans and co-eds., *Proc. Fourth MIT/ONR Workshop on Distributed Information and Decision Systems Motivated by Command-Control-Communications Problems,* vol. 4, Laboratory for Information and Decision Systems, Report LIDS-R-1159, MIT, Cambridge, Mass., Oct. 1981, pp. 19–60.
18. S. H. Kim, A Unified Architecture for Design and Manufacturing Integration, Technical Report, Lab. for Manufacturing and Productivity, MIT, Cambridge, Mass., 1987.

INSPECTION ROBOT

Applications in Industry

JERRY KIRSCH
KERRY E. KIRSCH
Kirsch Technologies
St. Clair, Michigan

QUALITY, INSPECTION, AND TESTING

The concept of product quality can be defined as the totality of features and characteristics of the product that bear on its ability to satisfy a given function. (1). In setting quality standards, an industrial organization must strike a balance between the costs of achieving a given quality level and the benefits accrued from the quality product. The benefits are usually the profits arising from increased sales of the final product. The benefits can also be viewed as improved production effectiveness when parts are processed within a given quality range.

To assure quality, parts and products must be inspected and tested. Inspection is defined as a process of careful search for nonconformities or errors, that is, falling outside the specified quality range. Inspection in industry has three primary functions:

1. Preventing nonconforming parts or material from proceeding further in the production process. The purpose is to avoid production of good-quality products.
2. Collecting data on specific characteristics of parts or materials for use in decisions regarding overall quality. Here the purpose is to identify whether imperfections are severe enough to be considered nonconforming.
3. Collecting data on specific characteristics or parts or materials to provide feedback for the manufacturing process. The purpose is to correct the process before more poor-quality parts are produced.

Inspection usually implies measurement of certain part properties, for example, geometric dimensions, surface finish, position accuracy, assembly integrity, and so on for quality control. Testing, on the other hand, implies some active examination of specific operational functions for quality control.

NEED FOR INDUSTRIAL INSPECTION EQUIPMENT

Industrial robots have been accepted for a variety of reasons which vary from task to task. By far the greatest single area of acceptance has been inspot welding. This has come about as a result of the improved consistency of welds made by robots over those made by humans. Spot welding guns are rather heavy and bulky, making this operation difficult and demeaning for humans and resulting in high costs for these operations. Spot welding robots have therefore been accepted for four reasons: flexibility, reduced cost, social aspects of the job, and improved quality of the finished product. Arc welding and spray painting are similar applications which have the added incentive to automate of relatively high skilled-labor cost. Although machine tending robots may or may not improve quality, they often are highly cost-effective in these dehumanizing jobs.

Applying this same thinking to inspection, it is clear that there is an incentive to automate. By allowing machines to perform tasks traditionally done by humans, there is a loss of the operator's intelligence and decisions which normally go without notice. Even in the most mundane of jobs the operator is required to inspect, think, and make decisions. Someone whose job it is to insert a screw in a hole must be sure that the screw has threads and that there is a hole to put it in. This person also has the wherewithal to throw a screw in the trash if it has no threads or put a workpiece aside if it has no hole. Although this example seems trivial to the onlooker, it can be disastrous for an automated system without these capabilities.

Human inspectors also are subjective in their duties. Say, in general, that there is no straight good or bad, only varying degrees of badness with respect to the ideal. Quality is often defined as within specifications. This is clear when looking at almost any blueprint that states a dimension as a specific number qualified with a tolerable variance. This means that if a component meets all of its specifications it is acceptable and can be considered a quality part. Being subjective, humans therefore make a judgment call based on some reference. These references can range from a single scale, or go-no-go gauge, to more sophisticated tools like optical comparators. Experience is also another method humans employ in place of mechanical references. Although experience is valuable to humans, it is subjective and its reliability is highly variable.

Machines, however, are highly objective in their operation. They apply logical judgment to any decisions required of them. The removal of any doubt, which is often injected when inspection is performed by the human inspectors, is always desirable. The consistency of these automated forms of inspection may be the single largest reason for their acceptance. As with all forms of automation there must be economic justification to motivate acceptance. Not only is there substantial labor savings in automating inspection, but there is also the cost of quality as compared with the lack of it, which often alone justifies inspection automation (2).

AUTOMATED INSPECTION

Automated inspection has been around for some time in various forms. Simple mechanical probes have been, and continue to be, used to determine presence or absence in many applications. Currently researchers and industry are approaching the inspection field using more sophisticated techniques. The trend in automated inspection today is toward the emulation of human vision (3). There are currently several suppliers of vision systems which incorporate two-dimensional matrix array cameras to perform gauging and inspection tasks. Early forms of three-dimensional inspection are also beginning to appear utilizing stereo techniques, structured light, and other methods.

There are three major components to a vision system, sensors electronic hardware, and software. Sensors come in many varieties. Currently the most prevalent are solid-state devices. They can be configured in one dimension, typically called a line scan camera, which is a single line of picture elements (pixels or pels) with lengths of up to 2048 elements or more. More common are two-dimensional cameras, which may be typically configured as 128 X 128 or 256 X 256 elements. These sizes are convenient to deal with as they are natural binary numbers; however, many cameras offer nonsquare formats. The imaging device is a solid-state array with individual picture elements that output a voltage corresponding to the light intensity received by the element. Electronics within the sensor scan the voltage signals are then output corresponding to the light intensity received by each pixel. To allow this information to be sorted out by an external device, some reference must accompany the video information. For this reason markers (syncs) are included in the video indicating the end of each line as well as the end of each page (composite video). Some manufacturers of sensors also provide a separate output of digital pulses called a pixel clock to allow one to discriminate accurately the timing between each pixel element.

The standard time to output a single frame of video is one-sixtieth of a second. This means that a 256 X 256 matrix camera which has a total of 65,536 individual elements outputs one element in approximately 200 ns, or at a rate of nearly 4 MHz. This is a great deal of information acquired in a very short time. The trend in sensor technology is toward increasing the size of the array to improve its resolution. By doubling the size of a 256 X 256 array to 512 X 512, the number of elements increases by a factor of 4 to 262,144, and the scan rate is increased to nearly 16 MHz.

There are three basic approaches to vision that are being used in industry currently: image buffering; edge detection; and windowing.

Image buffering (frame grabbing) is a system that digitizes an entire image and stores it in memory, where it may be analyzed by a computer. The gray levels of each picture element are stored as binary numbers in words typically 6 or 8 bits in size. Once an image has been stored, a computer can then read and write to the data. The image buffer approach depends heavily on software to perform a task. For this reason it is limited in speed only by the processor speed and complexity of the program.

Generally most of the information that is supplied from the camera is of little or no interest. What is of interest can often be described in terms of numbers as features. There are many useful features that can be quantified such as the object's area, centroid location, orientation and perimeter, among many others. These features can often be employed in identification, gauging, and inspection tasks.

Edge detection is a method of obtaining information about a scene without acquiring an entire image. Edge-detection-based systems record the locations of transition from black to white and white to black and store these locations in memory. A computer can then connect these points through a process called connectivity. Once the edges are connected the objects within the field of view can be separated into "blobs" which can be analyzed for their respective features.

In a further effort to reduce data and reduce program complexity a windowing approach can be taken. Windowing is a method where only selected areas of the image are analyzed. These areas, or windows, may surround a hole or some other aspect of a part in the field of view. Within a window a simple analysis may occur, such as a counting of the light or dark elements to determine a hole size. Such other operations can also be performed as finding the vertical or horizontal location of a transition between light or dark, or the total number of lines containing light or dark elements.

The artificial vision industry in the early 1980s can be compared to that of the robot industry in the mid- to late 1970s. There is a great deal of interest in vision and much is being said about the potential, yet the current market is quite small. Although sales are being made, the enormous potential has yet to be realized. It is estimated that the artificial vision market will surpass the one billion dollar figure by the early 1990s and this figure may even be low. New companies are beginning to appear on a regular basis in much the same way that robot companies sprang up in the late 1970s and early 1980s. The artificial vision community today is in many ways closely aligned with the robot industry. Many of the vision companies display their products and make technical presentations at the robot trade shows. Vision systems are applied to robotic inspections.

ROCKER COVER INSPECTION

An automotive plant that manufactures engine rocker arm covers had a requirement to increase levels of quality in their process. The covers manufactured are made for two models of engines, each type having both a right- and left-hand part, four part styles. They are assembled manually on two large-diameter dial index tables. Operators stationed around the table insert brackets, clips, and weld nuts and a baffle into fixtures that retain them in location prior to a spot welding operation that affixes the details to the cover.

The spot welding takes place on each of the two dial index tables followed by the automatic removal by a sliding arm having apart gripper jaw. The assembled covers are then visually inspected by an operator viewing the assembly as it moves by on a belt conveyor. This inspection was found to be difficult, especially when running production with the assembly having seven different parts welded to the cover. In many cases the inspector could not detect that parts were missing or that wrong parts were present. In some cases wrong parts would fit into the fixtures on the dial table.

The automatic welder also would occasionally create pinholes at points of spot welds in the assembly. These holes resulted in oil leaks and could not be detected visually by the inspector.

The defective parts then flowed through a washer and dryer, followed by painting and shipping to engine assembly plants. These plants, upon discovering the defects, would then return the entire shipment of the covers for reinspection and repair, followed by reshipment back to the assembly plan—to say the least, a costly quality problem.

The Application Scope and Task Performed

A dial index table was installed in a position where two conveyors transport assembled parts from each of the two assembly welding machines. The finished assembled covers are then manually deposited, two at a time, on positioning fixtures on the dial table. The operator depresses two palm buttons advancing the index table. Proximity sensors positioned over the table at one point detect whether the operator, in fact, loaded both fixtures; this information is also stored in memory. At the following station an overhead ram moves downward and depresses both covers to seal them as they would be found on the engine. Seals are mounted on the dial table around each of the positioning fixtures. Once the ram is in its down position a leak pressure test is performed. The test cavities are charged with 5 psi (0.73 kPa), and pressure readings are individually compared to acceptable limits of pressure decay, which is a result of leakage. Accept or reject decisions are then made for each part and stored in memory. Spot welds having small holes are rejected. The previous method in this testing was to pressurize the part under water and manually inspect for bubbles. This is obviously a costly and time-consuming process which was performed only on an audit basis.

Moving to the next station on the index table (Fig. 1), the two cover subassemblies are visually inspected to ensure that proper parts were assembled and that all parts are present. Opto-Sense with four matrix array cameras was programmed to inspect all of the part models. A model select switch is positioned on the operator's control panel. This inspection task requires the resolution of four cameras, and they are aligned to overlap a portion of their field of view. Two cameras view each cover.

Incandescent front lighting was found to be most suitable in this application. The windowing technique was also found to be most suitable in performing the inspection task. Both

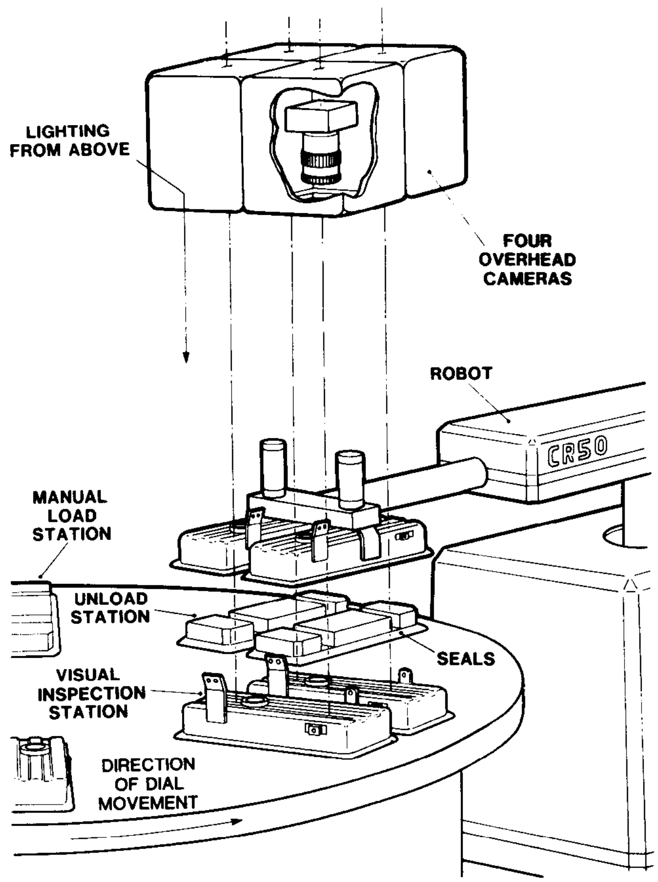

Figure 1. Rocker cover inspection system.

accept and reject for visual reasons are stored in memory. It is important to note that both the leak pressure test and visual tasks were proven and tested before commencing with the design of this system.

At the next station the valve engine covers are unloaded by a point-to-point robot. The robot arm, having two gripper hands, simultaneously grips both parts and removes them from the fixtures on the table. Empty fixtures are then indexed toward the operator's load position.

If a part is not in the fixture, as detected earlier, no inspection tasks are performed, and the robot is signaled as to which gripper hand is not to be activated to ensure that it does not attempt to grip and lift an empty position fixture bolted to the dial table. The robot proceeds loading accepted parts on a take-away conveyor. Once the robot moves the rejected parts, it places the rejects in one of three reject unload positions, sorting by reason of reject—visual, or leak, or both.

The results of this application can be summarized as follows:

Elimination of two quality problems.

Elimination of shipping and return and reshipping parts to car assembly plants.

Elimination of oil leak warranty costs.

Increase in production, with one person loading 600 indexes per hour for an inspection rate of 1200 covers per hour.

Efficient space utilization, as the completed machine is mounted on a 9-ft square unitized base.

Reduction in labor to reinspect because robot sorts rejects by reason.

SPRAY PAINTING APPLICATIONS

A manufacturer of appliances has a group of four spray painters on one shift and three on another. Their job was to spray paint the interior of five different models of rectangular metal containers with powdered paint. Any one of the five models can be found on an overhead moving conveyor. The conveyor's speed was changed before each shift to accommodate the change in numbers of spray painters. Any one of the five models of containers can be loaded onto unevenly spaced conveyor hooks that accommodate all of the models and transport them to the spray-painting booth. This spray painting task was found to be tedious and perhaps dangerous, as the material applied contained toxic chemicals (see also PAINTING).

The Application Scope and Tasks Performed

Two DeVilbiss spray painting robots were installed in the same area in which the spray painting takes place. They were programmed to work together and cover the areas to be painted on all five models. A Copperweld Opto-Sense Camera System was installed approximately 45 ft (15 m) upstream of the spray painting booth. Opto-Sense was programmed to identify any of the models as well as to perform a check determining whether or not any of the containers had previously been painted and not removed from the conveyor line.

In Figure 2, Camera 1 identifies the model of container which is transported between the camera and a translucent plastic screen. This plastic screen diffuses incandescent lights which provide back light directed toward Camera 1.

Opto-Sense was programmed in this application to use a windowing technique discerning the model by correlating the overall length and width with the different openings found in the various models. A hole was provided in the plastic light-diffusing material to provide an opening for Camera 2 so that it could view the interior of each container through it. The same lighting arrangement that provides the back light for Camera 1 provides front lighting for Camera 2, which illuminates the interior of the container. Camera 2, looking for previously painted containers, signals the robots not to repaint containers found already painted.

An encoder was placed on the conveyor's drive motor, and it pulses position information to Opto-Sense's control. With the distance verying between conveyor hooks, a switch is made by the hook at the Opto-Sense station. This switch triggers Opto-Sense to perform its visual inspection, after which the results are stored in its computer's memory. These results are then shifted to Opto-Sense's memory, which, with the pulses of the optical encoder, knows when to provide information as to which task the Trallfa robots must perform. This information is provided by Opto-Sense before the hooks contact switches at each robot station, which triggers each robot to commence with the correct program for the model on the hook.

The results of this application can be summarized as follows:

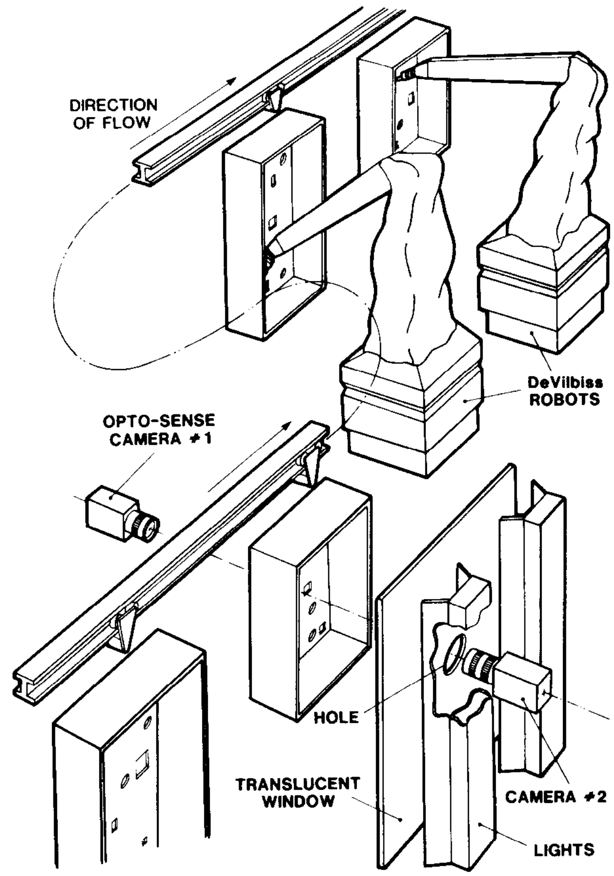

Figure 2. Robotic inspection in spray painting.

Elimination of seven spray painters performing a tedious and possibly dangerous task.

Improvement in quality of spray paint coverage with a reduction in material sprayed.

Removed possibility of respray of a previously painted part.

Opto-Sense vision eliminated the requirement for an operator who would have been required to identify models and manually feed this information to the Trallfa robots.

It is obvious that the user in this case is enjoying a handsome return on his investment.

APPLICATION OF SEALANT

The use of sealants and adhesives in assembly is found to be growing throughout many industries. One application in an automotive plant requires a person to apply beads of material in various zones on a sheet metal subassembly. These subassemblies are positioned in front of the operator who then applies the sealant. The sealant material is automotically dispensed from 55-gal drums. As the operator triggers a hand-held gun to dispense the material, he/she manually moves the gun through various paths of motion in the various zones.

While going through these motions the speed at which the operator works must be controlled to ensure that no less than 1/16 in. (1.6 mm) diameter bead of material is applied on the prescribed paths. It was found that, on occasion, gaps would appear on the paths. These gaps were created by the presence