How Much is Control Knowledge Worth? A Primitive Example*†

Jeffrey A. Barnett
USC Information Sciences Institute, Marina del Rey, CA 90291, U.S.A.

Recommended by Michael Genesereth

ABSTRACT
The most basic activity performed by an intelligent agent is deciding what to do next. Usually, the decision takes the form of selecting, from among many applicable methods, one method to try first, or opting to expand a particular node in a simple search. The most primitive case is selecting between two independent alternatives. Below, this case is examined and the value of the control knowledge that makes the decision is determined. Another result derived is the sensitivity of the expected value of control knowledge as a function of the accuracy of the parameters used to make these control decisions.

1 The Problem

Choosing among alternative methods to satisfy a goal is the most frequent and fundamental activity performed by an intelligent system. Minimizing expected cost is the natural selection criterion, and how to accomplish this is determined by control knowledge. There is a tradeoff between resources spent executing a method and those spent selecting the method since both

*This research was supported by Defense Advanced Research Projects Agency contract MDA903–81–C–0335. Views and conclusions contained in this document are those of the author and should not be interpreted as representing the official opinion or policy of DARPA, the U.S. Government, or any other person of agency connected with them.

†This article appears in Artificial Intelligence 22 (1984) 77–89 0004–3702/84/, Elsevier Science Publishers B.V. (North-Holland)

method selection and execution are part of the problem-solving activity. The question to be addressed is: Under what conditions are the resources spent on method selection returned to us, with a profit, by savings in method execution?

This question is investigated for the most primitive and perhaps the most important control decision of all: selecting between two independent methods that both might satisfy the same goal; if one method is tried and is successful, the other is not executed. Therefore, the choice between methods is actually a choice of their execution order.

Methods or activities are independent if their expected behavior does not depend on the order in which they are executed. A mathematical model of the effects of execution order for independent methods is presented and used to investigate three cases:

(1) method ordering is randomly selected—this is the case where no control knowledge is used;

(2) method ordering is selected using a priori information about the methods—this is the case where the best compile-time decision is made by control knowledge;

(3) method ordering is selected using situation-dependent information—this is the case where control knowledge makes the best run-time decision.

Several results are derived: (i) the criterion for selecting the execution order, (ii) the sensitivity of the expected cost as a function of the accuracy of estimating the parameters used to select the ordering, and (iii) the average expected reduction in method execution resources when the proper choice is made using situation-dependent information instead of a priori information. Since the average expected reduction is an upper bound on the average amount that should be spent on order selection, the original question is answered for the primitive case where the methods are independent.

2 Expected Cost of an Ordering

Let x and y be two independent methods that might satisfy the same goal. Define p_x as the probability x will satisfy the goal and e_x as the expected cost to execute x. Define p_y and e_y similarly for y. Since x and y are independent, p_x, e_x, p_y, and e_y do not depend on the order of execution selected for x and y.

Call $E(xy)$ the expected cost of executing x before y, i.e., $E(xy)$ is the expected cost of the program:

```
EXECUTE(x);
If goal not satisfied THEN EXECUTE(y);
```

It is easy to see that

$$E(xy) \;=\; e_x + (1 - p_x)e_y$$

because

(1) x is always executed by this program and therefore the expected cost e_x must always be incurred;

(2) y is executed and the expected cost e_y is incurred only when x fails to satisfy the goal;

(3) x fails to satisfy the goal with probability $(1 - p_x)$.

Similarly, $E(yx)$, the expected cost of the strategy that executes y first, is

$$E(yx) = e_y + (1 - p_y)e_x.$$

3 A Coin Flip

If method execution is ordered randomly, e.g., by a coin flip, then certainly no control knowledge is used. In this case, the expected cost CF is

$$\begin{aligned} \mathrm{CF} &= \tfrac{1}{2}[E(xy) + E(yx)] \\ &= \tfrac{1}{2}[e_x + (1 - p_x)e_y + e_y + (1 - p_y)e_x] \\ &= e_x + e_y - \tfrac{1}{2}[p_x e_y + p_y e_x]. \end{aligned}$$

because the orderings xy and yx are equally likely, and therefore the expected cost is just the average of $E(xy)$ and $E(yx)$.

4 The Best-Order Criterion

The optimal strategy picks the ordering with the minimum expected cost: If $E(xy) < E(yx)$, then x is executed first; if $E(yx) < E(xy)$, then y is executed first; and if $E(xy) = E(yx)$, it makes no difference because the expected cost of both orderings is the same. Hence, an optimal strategy executes x before

y whenever $E(xy) \leq E(yx)$. Therefore, if $\phi_z = p_z/e_z$ is defined for $z = x$ and $z = y$, a necessary criterion to execute x first is

$$\begin{aligned} E(xy) &\leq E(yx), \\ e_x + (1 - p_x)e_y &\leq e_y + (1 - p_y)e_x, \\ p_y e_x &\leq p_x e_y, \\ p_y/e_y &\leq p_x/e_x, \\ \phi_y &\leq \phi_x. \end{aligned}$$

Since these steps are reversible, $\phi_x \geq \phi_y$, is both a necessary and a sufficient condition to execute x before y and achieve minimum expected cost.

The result generalizes in a simple way. Assume that the problem is to select the order of execution for n mutually independent methods, any of which might satisfy the goal. Let ϕ_i be the ratio of the probability of success to expected cost for method i, where $1 \leq i \leq n$. Then the optimal least-cost strategy is the following: Execute first the method with the largest ϕ; if that method fails, execute next the method with the second largest ϕ, and so on. The ordering among methods with equal ϕ's is immaterial. This and other related results are developed in [1].

5 A Priori Information

When the measures p_x, e_x, p_y and e_y are given, the best ordering is xy if $\phi_x \geq \phi_y$ and yx otherwise. Further, the expected cost AP using this a priori information, is

$$\text{AP} = \min(E(xy), E(yx)).$$

The savings in expected cost using the optimal versus the alternative control strategy is easy to compute. If $\phi_x \geq \phi_y$ the expected savings S is

$$\begin{aligned} S &= E(yx) - E(xy) \\ &= [e_y + (1 - p_y)e_x] - [e_x + (1 - p_x)e_y] \\ &= p_x e_y - p_y e_x. \end{aligned}$$

On the other hand, if $\phi_x < \phi_y$, the reduction in expected cost using the optimal ordering (executing y first) is $S = E(xy) - E(yx) = p_y e_x - p_x e_y$.

Therefore, the expected savings using the best strategy versus the alternative strategy is

$$S = |p_x e_y - p_y e_x|. \tag{1}$$

If the best ordering is picked using the ϕ's instead of a coin flip, the expected savings is CF − AP $= S/2$, and this is the expected value of the control knowledge supplying the ϕ's.

6 Situation Dependency

The very best control knowledge tells us, for the current situation, that either $p_x = 1$ or $p_x = 0$ and that either $p_y = 0$ or $p_y = 1$, i.e., whether x, y, neither, or both will satisfy the current goal. This is equivalent to dividing the set of possible situations for method x into two categories: those situations where x is sure to succeed and those where it is sure to fail, and similarly for y. Such knowledge is almost never available until after execution and then it is too late for it to be valuable.

The next best thing is situation-dependent refined estimates of the p's and the e's. For the sake of simplicity, two assumptions are made. First, the situation-dependent statistics for the various methods are independent of each other. The second assumption is that e_x and e_y are essentially constant. Though this is not true in general, the expected cost of a method is usually not as sensitive to the dynamic situation as is its probability of success: Methods often employ some sort of simple search, and therefore the expected cost is related mostly to the size of the problem space rather than to the particular goal. On the other hand, the probability of success of many methods is more strongly related to the dynamic situation. For example, x solves the problem 75% of the time when the goal state mentions endgame pawn structures, 5% of the time for other pawn problems, and fails miserably ($p_x < 0.1\%$ say) elsewhere; dynamic control knowledge examines the situation to decide which of the three cases exists and assigns p_x the appropriate value.

These assumptions allow us to explore questions about limits on the value of dynamic control knowledge in a more straightforward way. Section 13 reviews the assumptions and their implications. Next, we look at an example of situation-dependent behavior, then we define a summary statistic for it.

7 Situation Taxonomies—An Example

Assume that a method suggests repair plans for electrical equipment. There are four exclusive and exhaustive equipment categories known by the system in which the method is used: major appliances, small appliances, tools, and radio and TV. The problems presented to the system occur in the mix 10%, 40%, 20%, and 30% respectively.

The knowledge in the assumed method is primarily about solid state components. Therefore, its effectiveness depends on the situation category. It works very well on radio and TV, problems and solves 60% correctly. For both major and small appliances it doesn't do so well and only gets 30% right. Its success rate is even worse for tools; it handles only 10% of the problems correctly since most of the mechanism is electromechanical rather than solid-state.

The situation-dependent behavior of this method is presented in Table 1. For each situation category in the situation taxonomy, two statistics are

Table 1. Situation-dependent example

Situation category	Frequency	Probability of success
Major appliance	10%	30%
Small appliance	40%	30%
Tool	20%	10%
Radio and TV	30%	60%

given: the frequency of occurrence of that situation and the probability that the method will perform correctly in that situation.

8 A Summary Statistic

Situation-dependent information about method x, such as that in Table 1, is conveniently summarized by F_x, where $F_x(p)$ is defined to be the probability that method x will be applied in a situation where its probability of success is p or less. Thus, F_x is a probability distribution of a probability measure. As always in such cases, $F_x(0) = 0$ and $F_x(1) = 1$ is assumed.

Fig. 1 shows F_x for the example in Table 1. The graph is generated by noting the following:

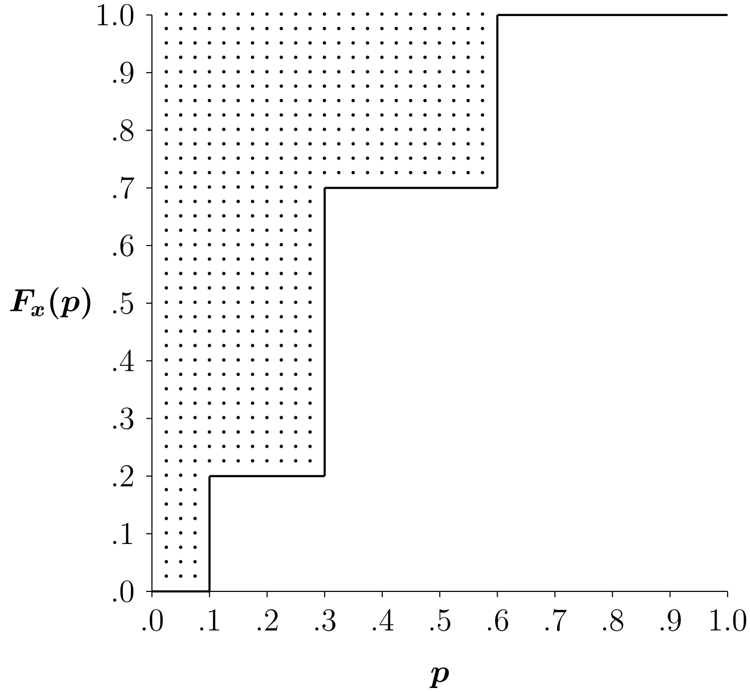

Fig. 1 F_x for example in Table 1.

– In no case is there a situation where method x's probability of success is less than 10%.

– The probability of success is 10% or less in 20% of the situations (tools). Therefore, F_x jumps to a value of 0.2 at $p = 0.1$.

– The probability of success is 30% or less in 70% of the situations (the union of tools and major and small appliances). Therefore, F_x jumps to a value of 0.7 at $p = 0.3$.

– Similarly, F_x jumps at $p = 0.6$ to a value of 1 because all situations have a success probability of 60% or less.

If F_x is differentiable, then $f_x = dF_x/dp$ is a probability density function. Further, $\int_0^1 f_x = 1$ and $f_x \geq 0$ because F_x is monotonic nondecreasing. Usually, however, F_x is derived from discrete case data and not differentiable everywhere. Therefore, it is assumed below only that $F_x \in C^1$ except at a finite number of places. This assumption allows all the necessary manipulations. Note, $F_x(p) = \int_0^p f_x(z)\,dz$.

It is evident that p_x, the a priori probability that method x solves a stated problem, is

$$p_x = \int_0^1 p f_x(p)\, dp$$

because $f_x(p)$ is the probability density that x is in a situation where it solves the problem with probability p. This form can be integrated by parts to show that

$$p_x = \int_0^1 f_x(p)\, dp - \int_0^1 F_x(p)\, dp = 1 - \int_0^1 F_x(p)\, dp.$$

Thus, for the example in Table 1, $p_x = 0.35$; this is the shaded area shown in Fig. 1.

As another example, assume that p_x is uniformly distributed in $[ri, r_2]$, i.e., $f_x(p_x) = (r_2 - r_1)^{-1}$ when $r_1 \leq p_x \leq r_2$ and $f_x(p_x) = 0$ otherwise. Then $p_x = (r_1 + r_2)/2$ and

$$E(xy) = e_x + (1 - \tfrac{1}{2}(r_1 + r_2))e_y. \qquad (2)$$

9 Dynamic Control Knowledge

When the situation-dependent control knowledge can refine f_x and f_y to situation-dependent estimates of p_x and p_y, the average savings in method execution resources is straightforward to analyze. Assume, without loss of generality, that $e_x \leq e_y$. If $\phi_x \geq \phi_y$ i.e., $p_x/e_x \geq p_y/e_y$, and x is executed first, the optimal strategy is followed. Therefore, a loss is incurred only if $p_x < p_y e_x/e_y$. In that case, the expected loss is $E(xy) - E(yx)$; and $L(xy)$, the average expected loss over all situations, is

$$\begin{aligned} L(xy) &= \int_0^1 \int_0^{p_y e_x/e_y} f_x(p_x) f_y(p_y)[E(xy) - E(yx)]\, dp_x dp_y \\ &= \int_0^1 \int_0^{p_y e_x/e_y} f_x(p_x) f_y(p_y)[p_y e_x - p_x e_y]\, dp_x dp_y. \qquad (3) \end{aligned}$$

Similarly, the average expected loss when y is executed first (with the condition that $e_x \leq e_y$) is

$$L(yx) = \int_0^1 \int_{p_y e_x/e_y}^1 f_x(p_x) f_y(p_y)[p_x e_y - p_y e_x]\, dp_x dp_y. \qquad (4)$$

Therefore, the most on the average that can be gained by refining f_x and f_y to point estimates of p_x and p_y is $\min(L(xy), L(yx))$ and this is the average expected value of the control knowledge using situation-dependent information to make the refinements over the value of the a priori control knowledge supplying the situation-independent estimates of the ϕ's.

10 A Simpler Case

Assume that methods x and y are indistinguishable by their summary statistics, i.e., $f_x = f_y = f$ and $e_x = e_y = e$. For example, both methods are possible continuations from the same point in a simple search, and there are no a priori grounds for distinguishing between two paths. Then $L(xy) = L(yx)$ by an argument of symmetry, and equation (3) assumes an appealing form. Let F be the cumulative distribution for f: $F(p) = \int_0^p f(z)\,dz$. Therefore,

$$\begin{aligned}
L(xy) &= \int_0^1 \int_0^y f(x)f(y)[ye - xe]dx\,dy \\
&= e\left[\int_0^1 yf(y)\int_0^y f(x)dx\,dy - \int_0^1 f(y)\int_0^y xf(x)dx\,dy\right] \\
&= e\left[\int_0^1 yf(y)F(y)dy - \int_0^1 f(y)\left[yF(y) - \int_0^y F(x)dx\right]dy\right] \\
&= e\int_0^1 f(y)\int_0^y F(x)dx\,dy \\
&= e\left[F(y)\int_0^y F(x)dx - \int_0^y F(x)^2dx\right]_{y=0}^1 \\
&= e\left[\int_0^1 F(x) - F(x)^2dx\right] \qquad (5)
\end{aligned}$$

where we have integrated by parts twice.

It is easy to see which F produce the minimum and maximum values of the integral, since F is monotonic and $0 \leq F(x) \leq 1$. The minimum value is 0, and this value is achieved when $F(x) - F(x)^2 = 0$. Therefore, $F(x) = 0$ or $F(x) = 1$ everywhere, and F is a simple step function, i.e., $F(x) = 0$ when $x \leq p$ and $F(x) = 1$ when $x > p$ for some $0 \leq p \leq 1$. Such step functions correspond to point density functions with $f(x) = 0$ except when $x = p$. Thus, the a priori density function is exactly the same as the point density refinement produced by the dynamic control knowledge: There is no gain in information, so no work is justified.

The maximum value of $F(x) - F(x)^2$ is $1/4$ and is achieved when $F(x) = 1/2$. This case corresponds to the bimodal distribution where the a priori knowledge says that in 50% of the cases the method is sure to fail and that it is sure to work in the remaining 50%, i.e., there is the most to be gained by dynamic control knowledge because (i) the situation-dependent control knowledge can determine with certainty whether x and/or y satisfies the goal, and (ii) there is maximal a priori uncertainty represented by the two disparate equal-sized modes. Therefore, the information gain is maximal.

Another example is useful. A reasonable way to represent a priori ignorance is $f(x) = 1$ (and hence $F(x) = x$), i.e., p is assumed uniformly distributed in $[0, 1]$. Here, $L(xy) = e/6$ from (5) and $E(xy) = 3e/2$ from (2) with $r_1 = 0$ and $r_2 = 1$. Therefore, the dynamic control knowledge is at most worth $\frac{1}{6}e/\frac{3}{2}e = \frac{1}{9}$ of the total expected work for method execution on the average.

11 Determining Control Information

Our attention has been drawn to p_x and its corresponding density and distribution functions, f_x and F_x. The meaning of p_x is clear: p_x is the fraction of the time method x satisfies the goals to which it is applied; this statistic has the distribution of situation occurrences buried in it. The value of p_x is obtained either by experimentation or theory. The significance of f_x and F_x may not be so obvious. Perhaps our intuitions will be sharpened if two hypothetical experiments are described.

Experiment 1 (observation). An intelligent system that includes method x is implemented. The knowledge engineer adheres to good practice and provides statistics-gathering facilities as part of his system. Each time it is run, the system prints a trace of its behavior including, for each method execution, the name of the method, whether it worked, and the resources it consumed. Eventually, the accumulated traces are subjected to analysis; p_x, e_x, and hence ϕ_x are calculated for method x and any other methods with which it competes. Since these computed values are situation-independent and the ϕ's are determined, the preferred a priori ordering of methods is determined and is available at compile time.

Experiment 2 (observation and situation analysis). The traces from Experiment 1 are augmented with information describing the goal that the method attempted to satisfy and other situation-dependent data. The knowledge engineer analyzes the additional information to build a taxonomy of situations.

The methods' statistics are computed for each situation type. At run time, the control mechanism, using the same parameters that were present in the trace, determines the type of situation and can therefore employ the proper situation-dependent ϕ's to select between alternative methods.

The value of performing Experiment 1 over random method ordering is $|p_x e_y - p_y e_x|/2$. (See equation (1) and the text following it.) The additional value of performing Experiment 2 is $\min(L(xy), L(yx))$. (See (3) and (4) and the text following them.) However, the amount of additional savings depends on, the shapes of f_x and f_y, and these in turn depend on the knowledge engineer's *cleverness in selecting the taxonomy*.

As an illustration, assume that data from 1000 executions of a method have been gathered and we have reason to believe these are a reasonably representative sample of the occurring situations. Two different taxonomies are proposed by the knowledge engineer and summarized in Table 2. The corresponding cumulative distribution function is shown below each taxonomy. For the sake of the illustration, assume that there are two methods with the same statistics. (The meaning of situation classes 1, 2, 3, $1'$, $2'$, and $3'$ are different for each method since independence is assumed.) Then the value of the situation-dependent control knowledge is $e/40$ if taxonomy 1 is used, while the value is $e/20$ if taxonomy 2 is used. Therefore, the value is doubled using the latter.

Table 2. Two different taxonomies

Taxonomy 1			Taxonomy 2		
Type	# Cases	# Satisfy	Type	# Cases	# Satisfy
1	250	150	$1'$	250	100
2	500	250	$2'$	250	150
3	250	150	$3'$	500	300

$$F(x) = \begin{cases} 0, & x < 0.5 \\ 0.5, & 0.5 \le x < 0.6 \\ 1, & 0.6 < x \end{cases} \qquad F(x) = \begin{cases} 0, & x < 0.4 \\ 0.5, & 0.4 \le x < 0.6 \\ 1, & 0.6 < x \end{cases}$$

As this example illustrates, the value of situation-dependent control knowledge depends crucially on the sharpness of its ability to distinguish differences in method behavior from one class of situations to another. Further, the closer the values of p to 1 or 0 for distinguishable classes of situations, the more valuable the control knowledge. Table 3 shows the limiting case where the value of the control knowledge is $99e/400$; there is a significant amount to be gained using the third taxonomy over using either taxonomy shown in Table 2.

Table 3. The best taxonomy

Type	# Cases	# Satisfy
a	550	550
b	450	0
$F(x) = 9/20$		

The onus on the knowledge engineer and the rewards for developing good situation taxonomies are clear. As always, there is no substitute for careful planning, detailed observation, and proper summary and use of the knowledge gained.

12 Estimation Sensitivity

Accurate values of p_x, e_x, p_y, and e_y are not always available, yet the control mechanism must still decide whether to execute x or y first. Control knowledge can either estimate these parameters or directly estimate ϕ_x and ϕ_y. The question is, how sensitive is the expected cost of method execution as a function of the accuracy in estimating the ϕ's by whatever means?

Assume the actual values are ϕ_x and ϕ_y, where $\phi_x = r\phi_y$, such that $r = 1 + \epsilon$ and $\epsilon \geq 0$. As long as the estimate for ϕ_x is greater than the estimate for ϕ_y, the proper control decision is made: x is executed first. However, if the estimate for ϕ_x is accurate but the estimate for ϕ_y is too large by a factor of at least r, the wrong ordering is selected. The worst case occurs when ϕ_y is overestimated by a factor of r^+; the fraction of extra expected work is then

$$[E(yx) - E(xy)]/E(xy) \quad = \quad [p_x e_y - p_y e_x]/[e_x + (1 - p_x)e_y].$$

Because $\phi_x = r\phi_y$, it follows that $p_x/e_x = rp_y/e_y$ and $e_x = p_x e_y/(rp_y)$. Thus, e_x is eliminated on the right by substitution. Since e_y will now appear in every term of the numerator and the denominator, it can be canceled to get

$$[E(yx) - E(xy)]/E(xy) \quad = \quad [p_x - (p_x/r)]/[p_x/(rp_y) + 1 - p_x].$$

Noting that $\epsilon = r - 1$ and simplifying leads to

$$[E(yx) - E(xy)]/E(xy) \quad = \quad \epsilon p_x p_y / [p_x + r p_y - r p_x p_y].$$

The partial derivatives of this form with respect to both p_x and p_y are positive. Further, p_x and p_y must be in the interval $[0, 1]$ because they are probabilities. Therefore, the maximum is achieved as p_x and p_y approach 1 where the value is ϵ.

Thus, if ϕ_x is accurately known but ϕ_y is overestimated, there is an increase of at most a factor of ϵ in the extra expected work that results because the suboptimal ordering is selected and this does not happen unless ϕ_y is overestimated by at least a factor of $1 + \epsilon$ and $p_x = p_y = 1$.

In general, both ϕ_x and ϕ_y are estimated rather than just one of them. Assume that ϕ_x is misestimated by a factor of $1+\epsilon_x$ and ϕ_y is misestimated by a factor of $1+\epsilon_y$, while $\phi_x/\phi_y = r > 1$ in actual fact. (Note, ϵ_x and ϵ_y may be positive or negative.) Then unless $(1 + \epsilon_y)/(1 + \epsilon_x) \geq r$, the proper ordering is selected. Since $(1+\epsilon_y)/(1+\epsilon_x) = 1+(\epsilon_y-\epsilon_x)-\epsilon_x(\epsilon_y-\epsilon_x)+\epsilon_x^2(\epsilon_y-\epsilon_x)+\cdots$, the estimation is to the first order, for small ϵ_x and ϵ_y, about $1 + (\epsilon_y - \epsilon_x)$. Therefore, if $r = 1 + \epsilon$ and $\epsilon_y - \epsilon_x < \epsilon$, the proper control decision is made; and if $\epsilon_y - \epsilon_x > \epsilon$, the maximum increase in expected work is bounded by a factor of ϵ and the worst case can occur only when $\epsilon_y - \epsilon_x = \epsilon$.

Therefore, reasonable estimates lead to reasonable behavior because the fraction of increased extra work is no worse than linearly related to the accuracy of these estimates. This result applies both to the a priori estimates for a method as a whole and to estimates of situation-dependent information.

13 Summary and Discussion

A mathematical model has been developed to investigate the effects of execution order among a set of independent methods, any one of which might satisfy the same goal. Several results have been developed:

(1) the criterion for ordering execution of the methods to achieve least expected cost:

(2) the expected savings using a priori estimates of the methods' statistics;

(3) the sensitivity of problem-solving cost to the accuracy of estimation of these statistics;

(4) the average expected savings in problem-solving cost using dynamic control knowledge;

(5) the value of better situation categories.

Results (4) and (5) are developed with the assumption that the expected cost of executing a method does not depend strongly on the current situation. The mathematics are simplified by the assumption because integration over situations is a messy business and because the appropriate situation taxonomy for expected work may be different from the one for probability of success.

The constant-cost assumption and the restrictions it entails need not extend into actual use of the above results. Experimental statistics-gathering approaches are described in Section 11. The knowledge engineer can and should consider employing similar empirical techniques to estimate situation-dependent values of expected costs in addition to the success probabilities so that he can compute the situation-dependent ϕ's for the elements of the cross product situation space of the categories for expected cost and those for success probabilities. However, more data collection is necessary to justify this move.

A reasonable compromise exists between assuming constant expected cost and performing a full statistical analysis. The knowledge engineer can develop a situation taxonomy based on ϕ values rather than p and/or e values. The objectives of this taxonomy are (i) the ϕ's for the cases within each category should be as nearly alike as possible and (ii) the expected ϕ values between categories should be as widely separated as possible. The compromise position is not only a good idea in practice, but in theory, too, as a careful review of the mathematics will show.

Several issues remain. The first issue is the independence assumption. Not only is the behavior of many methods dependent on the system's previous execution history, but dependency is a reasonable goal for method design: A good method should be able to profit from others' experiences. How to augment the current model to include the concept of dependency is not clear.

Another issue is that the model only minimizes cost. Since system behavior is measured in terms of the merit of the solutions it proffers and the way the solutions are developed, as well as the cost to find those solutions, this model is not realistic. All such tradeoffs are buried in the concept of a goal. The model would provide more guidance if the various facets of system evaluation were unbundled.

These and many other issues deserve our attention because the problem of method selection is the central problem for an intelligent system. The results developed here are a start, but they apply only to the most primitive case where the methods are independent.

References

[1] Simon, H. and J.B. Kadane. Optimal problem-solving search: all or none solutions, *Artificial Intelligence* **6** (1975) 235-247.

Received August 1982