# Deliberative Decision Making in "Conscious" Software Agents[1]

**ABSTRACT:**
When we humans are faced with a problem to solve, we often create in our mind different strategies or possible solutions. We imagine the effects of executing each strategy or trial solution without actually doing so. It's a kind of internal virtual reality. Eventually, we decide upon one strategy or trial solution and try solving the problem using it. This whole process is called deliberation. During the deliberation process several, possibly conflicting, ideas compete to be selected as the strategy or solution of the problem. One such is chosen voluntarily. In 1890 William James proposed a model that describes this voluntary decision-making calling it the ideo-motor theory. In this theory the mind is considered to be the seat of many ideas related to each other either favorably or antagonistically. Whenever an idea prompts an action by becoming conscious, antagonistic ideas may object to it, also by becoming conscious, and try to block that action. Or, other favorable ideas may become conscious to support it, trying to push its selection. While this conflict is going on among several ideas we are said to be "deliberating". Software agents, so equipped, should be able to make voluntary decisions much as we humans do. This paper describes a computational mechanism for this deliberation process, including James' ideo-motor theory of voluntary action. It also describes an implementation of the mechanism in a software agent. Some preliminary experimentation is also reported.

## Categories:
Action selection and planning, Human like qualities of synthetic agents

## Keywords:
Deliberation, voluntary action, ideo-motor theory, IDA, "conscious" software agents

## 1. INTRODUCTION

Deliberation is the process of debating with oneself concerning alternative courses of action. Humans do deliberate and make decisions by deliberation. Humans often think about the pros and cons of several courses of action before taking a decision. We think about various alternatives for solving a problem and finally decide on one alternative. For example when we are hungry, we may think of several alternatives, like going to a Japanese restaurant, a Mexican restaurant or a Chinese restaurant. A fondness for Sushi may finally push us to choose the Japanese restaurant.

The sequence of steps leading to a possible solution is called a *scenario*. Building a scenario and evaluating it to see if it is really a feasible solution is a significant part of the deliberation process. If a scenario is successful we consider the course of action (ideas) it represents a possible alternative for solving the current problem. These ideas compete with each other in the decision-making battle. We finally take the action suggested by the idea that won the battle. That action is called a "voluntary action" because it occurred as a result of the deliberation process.

One of the main goals of Artificial Intelligence is to produce intelligent software that can think and act like humans. One step in this direction is to produce intelligent software capable of deliberating about alternative courses of action for solving a problem, and of choosing a good one.

An autonomous agent is a system situated within, and a part of, an environment. The system senses that environment, and acts on it, over time, in pursuit of its own agenda. It acts so as to possibly effect what it senses in the future [8]. In this paper we describe an implementation of the deliberation process in an autonomous software agent called IDA (Intelligent Distribution Agent).

The paper is organized as follows: In Section 2 we describe the autonomous software agent IDA. In Section 3 we describe the process of scenario creation. Section 4 is devoted to ideo-motor theory and voluntary action. Some preliminary experiments and their results are described in section 5. Section 6 contains conclusions drawn.

---

[1] Primary Author is a graduate student at the University of Memphis

## 2. IDA

IDA is a "conscious" software agent (to be defined below) being designed and implemented by the "Conscious" Software Research Group at the University of Memphis. It's being developed for the US Navy. Each year thousands of Navy personnel are assigned new jobs at a cost of some $600 million dollars in moving expenses. This process of directing the movement of individuals to fill vacancies in ships, submarines etc. are called distribution [9]. The Navy employs people called *detailers* to perform this distribution task. Detailers try to keep both the sailors and Navy satisfied by keeping track of sailors' preferences while conforming to Navy policies. IDA is designed to completely automate the job of a detailer. As IDA's main goal is to be positioned as an alternative to a detailer, it must be able to communicate with sailors in natural language, access personnel and job databases, calculate the fitness of each job with respect to sailor preferences and Navy policies, deliberate about the temporal possibility of a job, select one or two jobs, and compose and send an email offering the selected jobs.

Like a human detailer, IDA must deliberate about the jobs to offer to the sailor. This deliberation should be done "consciously." "Conscious" software agents [5,6,9] are cognitive agents that implement global workspace theory, a psychological theory of consciousness [2,3]. Global workspace theory postulates that a mind is composed of small special processes, which are usually unconscious. It has two main constructs; this set of distributed unconscious processes and a global workspace or blackboard.

Baars used the analogy of a collection of experts, seated in an auditorium who can solve different problems. We do not know who can solve which problem. The global workspace strategy is to make the problem available to everyone in the auditorium by putting it on the blackboard. Then the expert who is expert on this particular problem can identify the problem and solve it. One of the main functions of consciousness is to recruit resources to deal with a novel or problematic situation. When a novel situation occurs, an unconscious process broadcasts that to all other processes in the system by trying to put it on the blackboard, which causes it to become conscious. Only one problem can be on the blackboard at any time. So Baars theory explains why consciousness is a serial system.

We call each of these unconscious processes *codelets*. A codelet is a small piece of code capable of performing some basic task (analogous to an expert in the above analogy). An attention codelet is a kind of codelet whose task is to push some information into "consciousness".

At any time in the mind different processes (attention codelets) could be competing with each other, each trying to bring its information to "consciousness". Processes may form coalitions depending on the associations between them. The activation of a process measures the likelihood of it becoming "conscious". The coalition with highest average activation finds its way into "consciousness."

Like we humans, voluntary decision-making in "conscious" software agents is done via deliberation. Different attention codelets, each representing a possible solution, compete with each other to come into "consciousness." That is, they compete with each other to be offered as a solution to the current problem.

## 3. SCENARIO CREATION

During deliberation we think about different possible solutions to a problem. These solutions take the form of scenarios, interpreted broadly. A familiar example for scenario creation in humans might be creating scenarios of different routes to a destination. When we think about different routes from one place to another, images of scenes along those different routes will be formed in mind. We start from the source location and try to build the travel route to the destination step by step. After building a scenario we evaluate the scenario to see if it can be a possible solution. If the route fails to reach destination we discard the scenario and starts a new one. Finally we select the best route among the successful routes.

Scenario creation in IDA follows Barsalou's perceptual symbol systems [4,7]. Scenarios are composed of scenes. Each scene depicts a particular place, object or event in whole scenario.

IDA starts deliberation with the job with highest fitness. This fitness value depends on many factors like sailor's preferences, Navy policies, sailor's qualifications etc. While deliberating about the jobs to offer to the sailor, IDA creates temporal scenarios about each job. Before offering a job to the sailor, IDA checks to see if the estimated date of departure from the current job falls before the estimated date of

arrival for the next job. IDA builds scenarios for each job by selecting a detach month for the current job, and then adding leave time, travel time and proceed time (extra time for a shore to sea move). Scenarios are constructed and stored as frames in IDA's working memory. Figure 1 shows a typical temporal scenario in IDA.
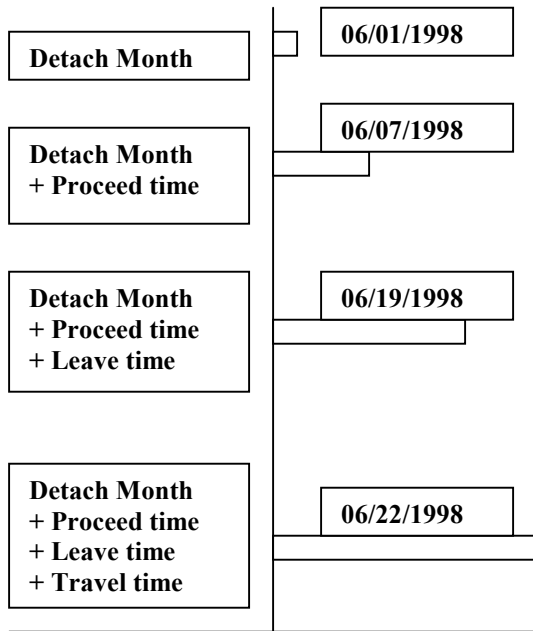


**Figure 1: A typical scenario in IDA**

Some scenarios are more complex, containing training time also.

*Gap* is defined as the difference between the resultant date after adding leave time, travel time and proceed time to the detach month of the current job, and the estimated arrival date for the next job. A scenario in IDA is considered to be successful if the gap is acceptable. Otherwise the scenario is adjusted to have an acceptable gap. The detach month of the current job can be adjusted within the PRD (Projected rotation date) window which is 3 months early and 2 months late. If the gap is still unacceptable even after adjusting the detach month within the PRD window, the job will be discarded, and IDA will start creating a scenario for the job with the next highest fitness in the list.

## 4. VOLUNTARY ACTION

William James proposed ideo-motor theory [10] to explain the phenomenon of voluntary action in humans. Whenever a person thinks of an action, an idea of it has been formed in his or her mind. If the action follows immediately the notion of it in the mind, we have ideo-motor action. The person is aware of nothing between conception and execution of the act. According to ideo-motor theory, an act will emerge successfully if the conscious image of that action can exist for sometime without competing images or intentions. If there are no antagonistic intentions, the action occurs spontaneously and without any inner-struggle. If there are some antagonistic ideas against an idea then these different ideas or images compete with each other, each becoming conscious in some order. In this case, the person is said to be in a state of "indecision". This state could last for hours, days or even months. At some point of time, some idea will be lucky enough to have no strong opponents and will immediately produce the appropriate motor effect. James called this act of metal consent to movement "fiat". There is no need to explain this. We all know about this and have experienced it.

During many processes of deliberation our consciousness is an extremely complex object containing many ideas related to each other favorably or antagonistically. An idea will eventually win out only if it remains in consciousness for certain amount of time with out any objections.

When IDA deliberates about a job, she thinks of different attributes of the job, such as the moving costs, the job's priority, its fitness and the gap. Each of these attributes may have a corresponding idea in IDA's subconscious, which, if the job is a good one from that particular attribute's perspective, tries to push it into "consciousness". These ideas are implemented by attention codelets representing each of these attributes that are looking at scenarios in working memory in order to propose jobs.

Whenever scenario creation is done for a job, the attention codelets looking at this scenario may propose or object the job depending on the particular attribute's value about which that particular attention codelet is concerned. The strength of a proposal or objection depends on the activation of the codelet, which proposed or raised the objection. That activation, in turn, depends on the value of the attribute or, perhaps, of several attributes. In our current implementation we have five different activation levels for each attention codelet. Figure 2 shows the different activation levels for the attention codelet concerned about moving costs.
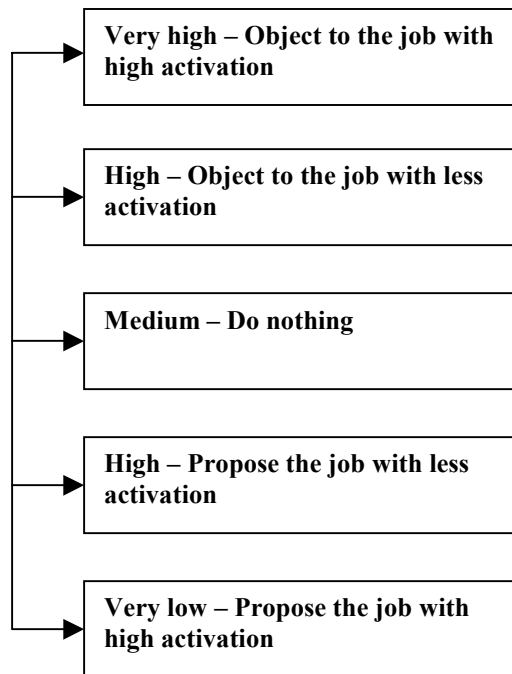
| Very high – Object to the job with high activation |
| --- |

| High – Object to the job with less activation |
| --- |

| Medium – Do nothing |
| --- |

| High – Propose the job with less activation |
| --- |

| Very low – Propose the job with high activation |
| --- |

**Figure 2: Different Activation levels for the attention codelet concerned about moving cost**

The strength of the proposal or objection of this attention codelet depends on the value of the moving cost for the job as shown in the figure.

If an attention codelet proposes a job with enough activation, it is able to bring the job into "consciousness". In response to this proposal, other codelets may favor the job, may object it or may propose a different job. If an attention codelet does not like the proposed job, it may look at the previous scenarios created and may propose a different job, if it can find one it likes better. If it cannot find any suitable job to propose, it just raises an objection against the job currently being considered.

If a codelet likes the proposed job, it may send some activation to the codelet that proposed the job so that the proposed codelet has higher activation. Every time a codelet comes to "consciousness", it looses some of its activation. Also the activation of the attention codelets decay constantly. After some point of time they completely loose their activation and die.

According to James, every idea awakens some actual movement (motor effect) to some degree and awakens it in a maximum degree when it is not kept from doing so by other ideas. An idea should remain in "consciousness'" for some time to be selected as the winner in the deliberation process. To keep track of the time a job is in "consciousness", we

have implemented a codelet called the *timekeeper* in IDA. Whenever an attention codelet proposes a job, the timekeeper starts its clock. If there are no objections for the job for a certain amount of time, timekeeper marks that job as "to be offered to the sailor". The amount of time, λ, the timekeeper should wait before marking a job depends on the number of codelets currently running in the system and should be determined via tuning. While IDA is in a state of "indecision" the timekeeper loses patience. Each time another idea becomes "conscious" λ is reset, but reset to a lower value than the default value at which it started. If the timekeeper's patience wears out, it sends activation to the attention codelet that initiates scenario construction. Thus IDA is likely to continue with scenario creation for the next job in the list. After selecting one job, the whole process is repeated and IDA may select a second job. The deliberation generally terminates after IDA finds 2-3 jobs for the sailor. If IDA cannot find any good jobs to offer to the sailor, she'll ask the sailor to wait until the next requisition list of jobs appears.

Our current implementation has three different attention codelets for such voluntary decision-making, one each for moving costs, priority and fitness. Each acts to propose jobs it likes and to oppose those it doesn't. Whenever a scenario is successfully created, these codelets look at the corresponding attributes of the job in the working memory and may try to propose or object the job with some degree of activation. The activation depends on the value of the attribute. If the activation of a proposal or objection is high enough, it may come to "consciousness" and be broadcast. For example, assume that the moving costs and the priority of the current job is very low. The attention codelet concerned with moving cost may like the job and get its proposal into "consciousness." Since the job has a very low priority the attention codelet concerned mostly with priority may search among the jobs with completed scenarios for job with higher priority. If it finds one, it may propose that job by bringing it to "consciousness." So there is a conflict here between the moving cost codelet and the priority codelet. Each may continue to propose its choice, but it does so with less activation each time. The conflict may continue until one of them wins because the other hasn't enough activation to make it into "consciousness." In this case, the winner still has enough activation to get into "consciousness" and to remain there without any

objections for sufficient amount of time. Or the timekeeper may loose patience and the scenario creator may get enough activation to start creating a scenario for the next job.

## 5. EXPERIMENTS

In this section we describe the results of various experiments and compare them with standard measures such as the z-score.

**Experimental Setup:**

We conducted various experiments on a data set obtained from the Navy APMS personnel and job databases. We carried out experiments with different sailors and different jobs. All the experiments described here were run on a Pentium-2 333Mhz, 128 MB RAM processor.

**Evaluation of Results:**

We compared our results with the results obtained using z-score. The z score for an item, indicates how far and in what direction, that item deviates from its distribution's mean, expressed in units of its distribution's standard deviation [1]. The formula for determining the z-score is:

$$Z = (x - \mu) / \sigma$$

(Where x is the observation, u is the mean and $\sigma$ is standard deviation). Since we used three factors during voluntary decision making: priority, moving cost and fitness, we have determined the best job for the given sailor according to the z-score from each of these three perspectives. We have also determined the best job according to the composite z-score, the average of the individual z-score values giving us the best result from all the three perspectives. Then we compared the results of IDA with the composite z-score results as the standard.

The experiments were done using 20 different sailors. In 17 out of those 20 cases the results obtained by IDA were the same as that of composite z-score for the best job. So IDA produced the same results as that of composite z-score in 85% of the experiments. We also compared our results for the second job that IDA selected over 20 runs. IDA produced the same results as that of composite z-score in 90% of these cases.

Recall is defined as the ratio of number of runs the results produced by IDA are same as that of standard z-score to the total number of runs. The graph in the Figure 3 shows IDA's

percentage of recall of with respect to z-scores of priority, fitness, moving cost, and also the composite z-score along the deliberation timeline averaged over 20 runs.
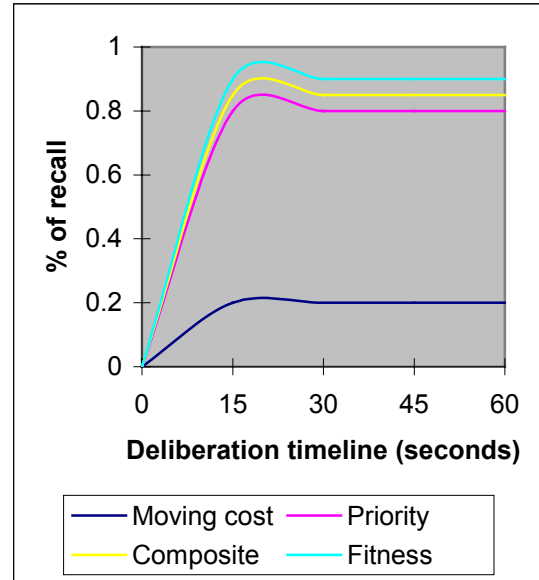


**Figure 3: Performance Comparison of IDA with respect to z-score.**

Generally the voluntary action part in deliberation for one job takes 50-60 seconds in IDA. So we've shown how the percentage of recall varies over a time line of 60 seconds with respect to the z-scores of moving cost, priority, fitness, and the composite z-score. As can see from Figure 3, the recall of IDA with respect to the composite z-score was 85% most of the time. Initially the recall was zero in all four cases because for most jobs some conscious and unconscious objections had to be overcome before any proposal was made. This occurs in humans also. Detailers, while considering a job to possibly offer, can look at the moving cost of the job and can think like "this job has a very high moving cost and I should not offer this job".

Our results coincide with those of z-score-fitness 90% of the time. In most of the cases the first job that IDA has considered was offered to the sailor. Initially there were some objections in all the runs. After a proposal was made, there were objections and supports for it, but there were no alternative proposals made. In many cases job first proposed was the one that was finally offered. This is because our constraint satisfaction module (linear functional) accurately assessed the fitness of the various jobs for the particular sailor. So in most cases the best

job turned out to be the job with highest fitness and, so, the first to have a completed successful scenario created.

Our results also agree with the results of z-score-priority in 80% of the cases. But as the graph shows IDA's results agree with the z-score-moving-cost only 20% of the time. Even though a job's moving cost is high IDA has selected the job because the other factors were judged more important. We give equal weight to all the factors in deliberation. If a job's priority and fitness are high, IDA will select the job even though it has a considerable moving cost. But we can always change IDA's preferences. If Navy does not want IDA to be spendthrift and warns it about the moving cost, IDA's emotions will change giving higher activation to the attention codelet concerned with moving cost. Then IDA will avoid offering jobs with high moving costs.

**Number of oscillations**:

During the period of indecision, several proposals, expressions of support and objections may be made for a single job or for different jobs. Referring to each of these as an oscillation, the number of such oscillations occurring during deliberation yields a picture of the dynamics of the deliberation process. The graph in Figure 4 shows the number of oscillations occurring at a given time during IDA's deliberation process averaged over the 20 runs.
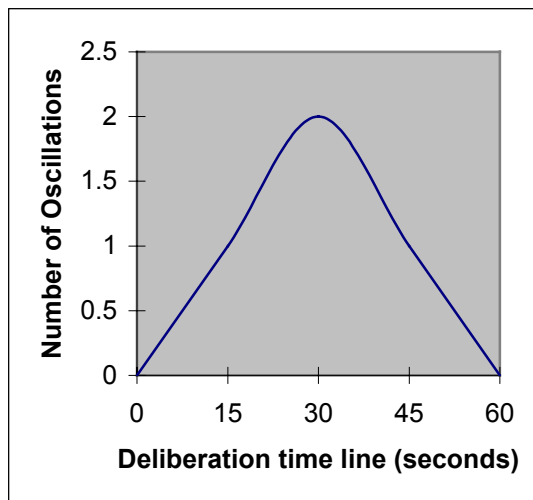


**Figure 4: Number of Oscillations over the deliberation timeline**

As the graph shows, the number of oscillations is initially very low because IDA starts by considering a single job. As time passes the number of oscillations increase until a maximum is reached at just after 30 seconds.

Oscillations then decrease. This shows IDA typically considering several jobs during he middle period. But as time passes IDA's "thinking" stabilizes and she finally decides on one job. This process often occurs in a similar fashion in humans also. We initially consider many things while deliberating, and will be in a state of indecision with many thoughts floating around in our mind. As time passes, we narrow down our choices, and finally end up with one solution. This shows that IDA deliberates in some ways like a human detailer. She is, in this regard, a human-like software agent.

## 6. CONCLUSIONS

This paper describes a more fleshed out conceptual framework for deliberative decision-making in humans, and its computational implementation in a "conscious" software agent, IDA. The experimental results reported show that our approach works in a way reminiscent of what each of us has experienced in our own human decision making. To the best of our knowledge, this is the first implementation of James' "ideo-motor" theory. Our current implementation is very specific to IDA. Our future work involves building a generic toolkit for the deliberation process in "conscious" software agents.

## 7. ACKNOWLEDGEMENTS

## 8. REFERENCES

[1] E.I. Altman, The Z-Score Bankruptcy Model: Past, Present, and Future (John Wiley & Sons, New York, 1997).
[2] B.J. Baars, A Cognitive Theory of Consciousness (Cambridge University Press, Cambridge, 1988).
[3] B.J. Baars, In the Theater of Consciousness (Oxford University Press, Oxford, 1997).

[4] L.W. Barsalou, Behavioral and Brain Sciences 22 (1999) 577.

[5] M. Bogner, Realizing "consciousness" in software agents (Ph.D. Dissertation, University of Memphis, Memphis, TN, USA, 1999).

[6] M. Bogner, U. Ramamurthy, S. Franklin, in: Human Cognition and Social Agent Technology, ed. K. Dautenhahn (John Benjamins, Amsterdam, 2000) p. 113.

[7] S. Franklin, Neural Network World 10 (2000) 505.

[8] S. Franklin, A.C. Graesser, in: Intelligent Agents III (Springer Verlag, Berlin, 1997).

[9] S. Franklin, A. Kelemen, L. McCauley, in: IEEE Conf on Systems, Man and Cybernetics (IEEE Press, 1998) p. 2646.

[10] W. James, The Principles of Psychology (Harvard University Press, Cambridge, MA, 1890).