Mobile Agents: Are They Useful for Establishing a Virtual Presence in Space?

Todd Papaioannou

MSI Research Institute
Department of Manufacturing Engineering
Loughborough University
Loughborough,
Leicestershire, LE11 3TU, UK
Email: T.Papaioannou@lboro.ac.uk

Abstract

The launch of the Deep Space One (DS1) probe heralded the beginning of a new phase for NASA, which will be embarking on a series of missions to test and deploy low cost, demonstration technologies. In an attempt to cut mission costs from billions to under 100 million, NASA is looking towards autonomous spacecraft and robots to establish a virtual presence in space. This paper describes mobile agents and discusses their uses in tackling the problems of adjustable autonomy and long distance interaction. The concepts of autonomy and emergence are then briefly discussed, before combining them with a mobile agent based framework within which the properties of adjustable autonomy and interaction at a distance are inherent. Finally, mobile agents are examined within several plausible contexts to demonstrate their applicability.

Introduction and Motivation

It is the stated intention of NASA to establish a virtual presence in space through the use of a plethora of robotic probes. The increasing motives for space exploration, and the diversity of the environments and missions being encountered require machines that are both robust mechanically, as well as computationally. The recent paradigm shift from billion dollar missions with massive ground crews, to cheaper, targeted missions with much smaller ground crews has required the creation of spacecraft that are largely controlled by intelligent and self sufficient entities, known as Remote Agents [1].

Current research is investigating how to control the interaction between agents, their internal sub-systems and humans. Adjustable autonomy is of particular interest when dealing with systems that may have to function for several years unattended [12], or may be forced to halt their mission until they can get further instructions from human operators.

However, the increasing distances and time delays involved with space exploration make remote operation of robotic probes and vehicles logistically impossible. New methods to support assuming full control of a probe, or subtle alteration of its operating constraints, are required that can overcome amongst other things, the problem of long distance interaction.

This paper describes mobile agents, contrasting them to simple mobile code, and discusses their uses in tackling the problems mentioned above. The concepts of autonomy and emergence are then briefly presented, before introducing them to a mobile agent based framework within which the properties of adjustable autonomy and interaction at a distance are inherent. Finally, mobile agents are examined within several plausible contexts to demonstrate their applicability.

Mobility

Mobile Agents have, until recently, been a fringe topic for discussion in mainstream autonomous agent research. Nonetheless, there is an expanding community dedicated to investigating the potential of this new paradigm [16]. Although there is still no definitive answer as to whether mobile agents will live up to the expectations placed upon them, there is a growing body of work [2], and frameworks [17] **-** [21], to support further investigation of the technology and the benefits it may bring.

The notion of Mobile Agents was established in 1994 with the release of a white paper by White [3] that described a computational environment known as *Telescript*. In this environment, running programs were able to halt their execution and transport themselves from host to host in a computer network, resuming execution at the new host.

Copyright © 1999, American Association for Artificial Intelligence (www.aaai.org). All rights reserved.

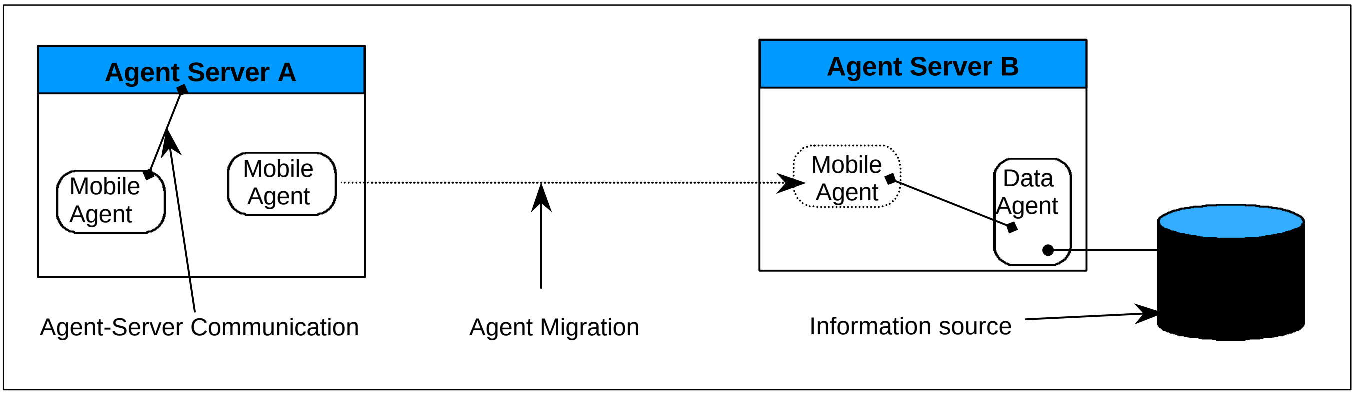

Figure 1. Mobile Agents in action.

Within the scope of the this paper then, a mobile agent will be defined as:

> *a software agent that is able to autonomously migrate from one host to another in a computer network.*

By their very nature, mobile agents are inherently distributed. As such, they must be executable across a variety of platforms and operating systems to achieve their full potential. This need has influenced the way in which mobile agent systems are created in that these systems are usually written in some type of script or bytecode that can be interpreted. Interpretation removes the need to recompile the agent on arrival at a new host, and thus simply requires an environment at the host that is capable of uniformly executing the agent on arrival. Most examples of mobile agent systems have a server or some type of running environment in which the agents are executed [17][4].

These servers act as workplaces where agents of all types can communicate with each other, see Figure 1. The servers provide a means for hosting and managing agents in an environment that is secure from malicious[1] agents. Through this server, an agent is able to get information about its environment, and to send and receive messages to that environment and other agents currently active in its proximity. Mobile agents are able to communicate by passing messages between themselves or via the server environment.

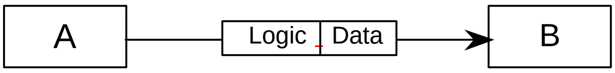

Figure 2. Mobile Logic **and** Data

Mobile Agent systems are quite different to existing traditional distribution systems prevalent in industry today. These systems may be characterised as providing **local interaction** for communicating agents, and providing facilities for mobile **logic *and* data**.

It is this distinction that makes mobile agents particularly useful. In contrast to a mobile code system, for example a Java applet, or a command sequence, only the logic is transferable. In a traditional distributed system, for example CORBA, only the data is transferred as a message.

Advantages of Mobile Agents

Some of the advantages of Mobile Agents include the simplicity of the installed server base, server flexibility, local and real time interaction. Further analysis of these and other advantages can be found in [5][6]. Mobile agent frameworks also tend to support the use of cloning, persistent storage and multicast messaging [8].

Most of the program logic usually encapsulated by current distribution paradigms in static clients or servers has been moved into the mobile agent, ensuring the installed server base for mobile agent systems can be quite simple. All that a server must do is support the migration of agents and provide access to any resources at its disposal. This in turn allows for much great flexibility at the server, since an incoming agent can effectively alter the behaviour of the server 'on the fly'. Systems such as Straum [7] take the view that a server is an 'island' upon which a mobile agent may alight to take advantage of the local resources. These resources are not available across the network, although inter-agent communication is, meaning an agent must transport it self to a resource or send a slave.

Perhaps the most quoted reason, at least during peer group discussion[22], for not using mobility is the view that at some point in the future the Internet will have sufficient capacity to support all the traffic upon it, and that client/server interactions across the web will be the same as local interactions within a single machine, thus negating the advantage of local interaction available with

---

[1] A malicious agent is one that may attempt to take control of another agent, or access its private internal data.

the mobile agents approach. This is a contentious viewpoint [14], but we may discard it in this paper since the network we are considering is extremely vast, i.e. a network of spacecraft distributed across the solar system. With a network this large factors such as the speed of light will come into effect. Thus, local interaction rather than tele-operation must have a role to play in the future of our virtual presence in space.

Autonomy and Emergence

The issue of defining autonomy is one that has dogged the agents community since its inception. There is still no *de facto* definition, although this does not seem to have been a great obstacle to groundbreaking research [11]. Although the focus of recent agent research, and historically AI research, has been to produce fully autonomous, reactive, adaptive entities, be they robots or agents, it turns out that this requires rather a lot of resources and is very complex. By entity, this paper means robot, spacecraft, probe, etc. More recently, the focus has shifted towards building much smaller, task specific entities. Robotics researchers have been drawing inspiration from nature and building extremely small insectoid like robots that have extremely simple but distributed control mechanisms [2].

This new direction is becoming increasingly popular since its minimalist approach allows massive redundancy to occur within a group of robots (as opposed to over engineering a single more complex robot). By spreading the load over a large group, it is possible to avoid catastrophic failure. However, since we are now dealing with many robots within a group as opposed to a single point of control, the issue of autonomy has changed somewhat. We must now consider how to control a ‘swarm’ of simpler robots, and how they will interact with each other. This issue is equally applicable to agents.

Agents and robots can work as non-interactive individuals or as a collective. When working in isolation, the observable behaviour is simply the predefined entity following its particular course of existence. However, when working as a collective the possibility of something new and different can result in what is being termed as emergent behaviour. Within the scope of this paper emergence is defined as:

> *Emergence is the existence of a coherent pattern that arises out of interactions among simpler objects.[9]*

Typically, emergence happens unpredictably, although it is possible to ‘design in’ the emergence you would like to see. An example of this can be seen in [10], where a manufacturing system is controlled by agents that differ along two axes of description: level of abstraction and scope of reasoning of an individual agent. Type A agents, the lowest abstraction that usually reflect physical entities, work as a decentralised collective. If the collective encounter an intractable problem they create a Type B agent to solve the problem. If the problem is solved, B notifies the Type A agents and then expires. Otherwise, B must widen its scope, and in doing so may encounter other Type B agents. If these B agents have conflicting goals a Type C agent may be created that is capable of interacting with both A and B agents in such a way that the local agent goals are sacrificed if necessary for the good of the global goals. Ultimately, the creation of type B and C agents is dynamic, and controlled by the type A agents themselves. The layered control architecture gives rise to emergent behaviour from the static, simple lower level agents. This would appear to be a suitable framework for controlling a swarm of robots.

Fundamentally though, there are some short falls to this approach. Firstly, any behaviour that resulted in the successful resolution of a particular problem is lost once the type B or C agent expires. Secondly, this approach is not ideal when the agents are distributed over a wide physical area. For example, imagine a type C agent trying to co-ordinate several type A and B agents over the Internet, when the agents are geographically dispersed. The network latency, bandwidth problems and data integrity are all issues that can affect the smooth resolution of the problem.

If the agents are not static, but can be mobile, another solution would be for all the agents to send a representative or clone of themselves to a central location where the negotiation could take place. Equally, we may imagine a mobile ‘supervisor’ agent patrolling the fringes of the network, controlling, directing and negotiating with other much simpler entities.

A Mobile Agent Control Framework

In presenting a control framework inhabited by mobile agents this paper takes what is perhaps a contrary position in viewing the entities within our virtual presence as ‘islands of resources’ (see [7]). It is accepted that current missions and spacecraft are engineered to a much higher degree of complexity, with little or no redundancy afforded through sheer numbers. Each is specific instead of generic. However, a shift towards smaller, re-useable robots made from off the shelf components seems inevitable [15].

The proposed framework is shown in Figure 3. Each entity within the system provides a simple hosting mechanism for a mobile agent through which the agent is able to migrate to the host. When in-situ the agent is then able to take full control of the hosting entity or simply access its local resources, in an effort to complete its tasks.

---

[2] See the work of Mark Tilden at Los Alamos National Laboratory, New Mexico.

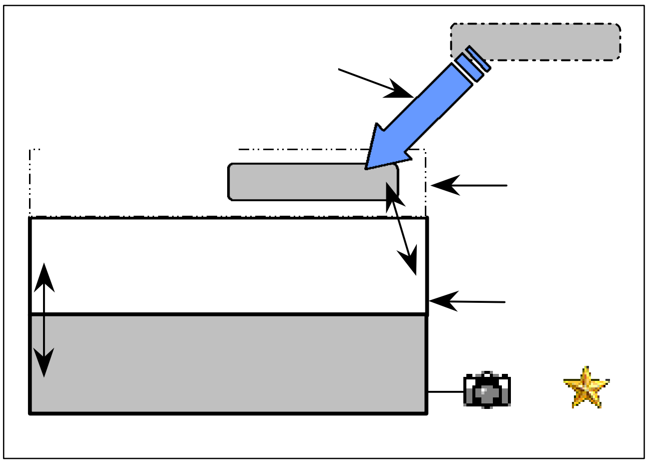

Figure 3. Mobile Agent environment layered over original control architecture

Resources used may vary, but could be as simple as temporarily controlling an onboard camera or soil probe. This framework is generic, simple, flexible, and provides a mechanism for possibly re-tasking the entity.

However, in contrast to mobile code that is just blindly executed, a mobile agent is able to display all the features associated with agency, including reactivity, pro-activity and learning. At this point, it should be stated that the existing level of autonomy would control the level of interaction available. If the incoming agent is solely wishing to utilise a simple resource on a complex spacecraft, the underlying control and autonomy systems are responsible for exposing the 'resources' of the entity and may remain responsible for ensuring actions are not taken that might endanger the entity, or prohibit it from completing its major mission. On the other hand, the entity may be a simple dedicated robot, and the mobile agent may wish to take complete control. These issues are currently being examined and will remain central to the success of the proposed framework.

Adjustable Autonomy

Typically, adjustable autonomy is described as the property of an autonomous system to change its level of autonomy to one of many levels while the system operates [12]. This may mean that this level is adjusted by a human user, another system or the system itself. In general though, adjustable autonomy has been considered within an encapsulated system, for example a Mars rover's control system, or perhaps the life support systems of a Lunar base.

In the above examples, autonomy may be adjusted locally in the Lunar base by human intervention or remotely by ground control. Whilst in the Mars rover the autonomy may only be adjusted remotely (not withstanding the interplay between the internal autonomous systems). In fact, at up to 40 minutes for a radio signal round trip, tele-operating a rover from Earth is not practical, so any adjustment must be made by uploading command sequences or a new program, which as we have seen is simply mobile code. It is precisely this autonomy adjustment from a distance that is the forté of mobile agents.

To examine the use of mobile agents in this scenario we will place them in a hypothetical context. As we have seen above, the trend in current robotics is towards ever simpler systems that are task specific and when employed in large numbers can handle failure in parts, since the whole is able to absorb the loss. A common example of swarm like behaviour for space exploration may be the clearing of a landing site by an army of cheap and simple robots [13]. With the in-built redundancy afforded from their numbers it would not matter if any were damaged or died, ultimately their remains would also be cleared away.

However, since the terrain of other planets may hold many surprises we will interject an intractable problem into the equation. For example, during the clearing operation a particularly large boulder is presenting the robots with a problem of how to clear it away. On their own, none is strong enough to perform the task, whilst their numbers can overcome the problem, it involves some complicated interaction and co-operation between the robots.

Certainly, if all the robots had been built with complex autonomous control systems that were capable of peer negotiation and co-operation this problem may well be solved by the robots themselves, as would be the case with the manufacturing control system discussed earlier. However, for our purposes this solution is prohibited by cost and practicality. The remaining options involve attempting to clear the rock by tele-operating a large number of the robots by a ground or base crew, but this in itself is impractical for reasons of time delay and complexity, or to devise a program that can be uploaded to the robots which co-ordinates their combined efforts to shift the rock. Yet, whilst perhaps possible, this complex choreography appears equally complex as tele-operation. The insurmountable problem is the distance to the problem and the time delay involved.

Mobile agents offer a method to overcome the time delay, because they are able to migrate to the locality of the problem and thus interact with the robots in real time. The proposed solution is thus: once the problem has been encountered, the robots realise they are not capable of shifting the rock alone and signal home that the problem exists. Here, home may be considered a ground crew on Earth, or perhaps a base crew on Mars, or even the lander from which they emanated. An assessment of the problem suggests that dispatching a supervisory agent to take control of the robots temporarily and co-ordinate their efforts will solve the problem.

The mobile agent is dispatched, and arrives taking control of one of the robots to act as its host. The agent is then able to tackle the problem by direct control of the robots, acting as a conductor, or by cloning itself and allowing each of its clones to take control of another robot host, at which point the newly resident agents co-operate to shift the boulder. On completion of the task, the agent (or a clone) is able to return home to report on the success of its mission and can divulge any lessons learnt. The knowledge gained on how to tackle the problem can be added to the agents' database for future reference. This type of problem may re-appear during the clearing of a large area and it may be that the agent has remained on site and is able to infer from the previous problem how to tackle the new situation without reporting back to base. Armed with its mobility the agent is able to cruise around the robot swarm appearing at the crux of the problem to troubleshoot and co-ordinate the robotic efforts, analogous to the role a foreman might play in a manufacturing plant. In the case of two problems arising at the same time the agent is able to clone itself to tackle both concurrently and recombine at a later date to assimilate all the new knowledge.

However, let us assume that the assessment of the crew has been mistaken, and on arrival the agent finds that the rock is actually showing less than half of it's true size. To solve the problem the robots are going to have to dig out the bottom part of the rock before they can transport it. This eventuality may eventually have been surmountable through tele-operation, since the humans involved could have reached this conclusion. If the initial problem had been tackled by uploading choreographed command sequences, it is likely that a significant time delay would be incurred before the new problem was even reported, let alone a new set of commands produced.

Assessment

By using a mobile agent the simple autonomy of the robots has been adjusted remotely, although the actual control of the robots takes place locally. This local interaction allows the agent better real time control of the robot population, enabling it to react to unforeseen circumstances. The robot population is able to display all the characteristics we associate with agency, but only on demand. Thus, the cost and complexity of the robots has been kept low, whilst the nature and intent of the mobile agent can be dynamically tailored before dispatch to suit the particular problem.

An associated benefit is the return of the agent at which point it can be 'debriefed' to establish why a particular approach succeeded or failed, and learn from any mistakes or lessons. It is possible to imagine a team of troubleshooting agents that are kept dormant in persistent storage until their skills are required.

Long Distance Interaction

By now, it should be clear that mobile agents are inherently suited for long distance interaction. Their particular strengths of being able to react to situations that may have been unforeseen at the time of their dispatch makes them a viable option for interacting with deep space probes, missions into the farther reaches of the solar system, and planetary ground teams (be they human or robotic).

To see their applicability we once again put them into a context for examination. Let us this time imagine that we have moved into the near future, and the solar system has been populated with a network of probes, spacecraft and robots. Circling the moon Europa is a small transport ship from which a group of hydrobots are sequentially dispatched to search for life under the ice rafts. At some point, one of the hydrobots returns to the surface with what looks like some very exciting results. On arrival at Earth, the results cause several scientists to get extremely excited and they pull an all-nighter devising a mission to further examine these results. The command sequence is dispatched to the transport ship, which receives and programs one of the remaining hydrobots with the new mission. The hydrobot is launched and carries out the new mission.

This scenario seems plausible enough, but what is the effect of new results emerging whilst the mission is in transit? Perhaps the initial hydrobot happens to run a routine diagnostic and discovers that one of its sensors was damaged and reports from it have been skewed. In the mobile code instance (the command sequence), the second hydrobot would probably still be launched and might in fact be wasting valuable resources investigating a worthless area. With a mobile agent, the agent would update its knowledge base on arrival with any reports it had missed en route and deduce that the mission it was tasked with was not required. The decision could be made to terminate itself, or perhaps continue with an alternative mission plan. In either case, a potential waste of a valuable hydrobot or other irreplaceable resources would be avoided.

It can be argued that the sensor check should have been done before the report transmission, or that conditional command sequences might alleviate some of the problems with long distance interaction, but Murphy's Law may rear its head at any time. Unforeseen circumstances are a reality in space exploration. Mobile agents offer a solution to long distance interaction by providing a flexible, reactive approach to re-tasking probes and robots in-situ rather than a less flexible mobile code approach.

Commentary

Certainly, this paper is high in 'vision' and low on real implemented systems. It is the author's hope that this does not detract too much from the intended message. That is: the flexibility offered by the proposed mobile agent framework may be invaluable in the future of our virtual presence in space. The mobile agent technology discussed is currently available and under continued research. Off the shelf expendable robots are nearing reality. There is an obvious requirement to investigate methods of interfacing any existing autonomous agent systems with the new mobile agent host, and ensuring viable communication between the two systems. The fundamental shift is from a static, crystalline, matrix of communicating entities to a fluid system where physical resources are anchored but the controlling logic is encapsulated within dynamic mobile agents.

Conclusions

Mobile agents are a burgeoning field of study within the ever-growing autonomous agents arena. Their key advantages include local interaction, server flexibility and real time interaction.

This paper has presented the case for mobile agents to be used in the support of establishing a virtual presence in space. It has examined their role in an ecology of robots and spacecraft that are composed of systems with varying levels of autonomy, and suggested how future mission costs can be kept to a minimum by combining the current trend for large numbers of simple, task specific robots with a flexible, adjustable, autonomic framework. This approach allows a mobile agent to be dispatched to trouble shoot particular problems in situ, by co-ordinating the combined efforts of several lower level robots, or by subdividing itself to temporarily adjust the autonomy of the host robots, before co-operating with its siblings to solve the problem. Knowledge associated with the solution can be integrated into the agent's database, whilst a clone can be 'debriefed' at home base. If the agent acted as a single entity, split itself into several constituent parts, or split into clones the relative merits of each action can be assessed. Sufficient numbers of hosts may give rise to emergent behaviour within a robot 'swarm' where we might see a fluid and mobile presence roaming the interconnections of the underlying robot network, appearing at points of conflict to resolve them.

By their very nature, mobile agents are extremely useful for long distance interaction. The ability to create a new agent locally (to the creator) and dispatch it to a remote host where it can resume operation, interacting with local resources, in an intelligent manner is being touted by the mobility community as one of the fundamental differences in this approach. This is in stark contrast to the traditional method of having static, bound entities communicating at a distance. For space exploration, this shift is even more profound when the distances involved are considered.

The ideas in this paper should not, and can not, be viewed as a standalone solution to the problems of establishing a virtual presence in space. They do however illustrate a potential path in which missions can be kept cost effective, and flexible. The problems of, amongst others, modelling and predicting behaviour, situation awareness, and mission criticality are still inherent to field.

Ultimately, this paper presents a vision of space inhabited by an ecology of robots, satellites, space craft, rovers, planetary bases and the like. In contrast to most visions, these entities act not only as fundamental actors in the vision, but as a network of resources. Each entity has some level of autonomy, which may range from very simple task specific instructions, to more complex autonomous agent architectures. Mobile agents live in the network, able to migrate, clone, sleep, wake, but in reality insert a higher layer of control and abstraction over the underlying hardware and software.

In truth, it will be some time before sufficient numbers of these entities inhabit space to make the vision reality. However, the advantages of local interaction, adjustable autonomy and interaction at a distance make mobile agents a particularly useful technology even with a single spacecraft.

Acknowledgements

The author would like to thank the EPSRC for their continued support.

References

[1] Muscettola, N., Nayak, P.P., Pell, B., Williams, B.C., "*Remote Agent: To Boldly Go Where No AI System Has Gone Before*", Artificial Intelligence 103(1/1), August 1998.

[2] Vitek, J., Tschudin, C., editors, "*Mobile Object Systems, Towards the Programmable Internet*", Lecture Notes in Computer Science 1222, Springer-Verlag, 1997.

[3] White, J. E, "*Telescript technology: the foundation for the electronic marketplace*", White Paper, General Magic, Inc. 1994.

[4] Gray, R., "*Agent Tcl: A flexible and secure mobile-agent system*", PhD thesis, Dept. of Comp Sci, Dartmouth College, June 1997.

[5] Chess, D., Harrison, C., Kershenbaum, A., "*Mobile Agents: Are They a Good Idea?*", in [2], also available at http://www.research.ibm.com/massive/mobag.ps

[6] Papaioannou, T., Edwards, J., "*Using Mobile Agent Technology to Improve the Alignment Between the Sales Order Process in a Virtual Enterprise and its IT Systems*", Int'l Journal of Robotics and Autonomous Systems, to appear.

[7] Minar, N., "*Designing an Ecology of Distributed Agents*", Masters Thesis, Media Lab, MIT, 1998.

[8] Lange, D, Oshima, M., "*Programming and Deploying Java Mobile Agents with Aglets*", Addison-Wesley, 1998.

[9] Odell, J., "*Complex Adaptive Systems*", submitted to OOPSLA'98 workshop on Modelling Dynamic/Emergent Distributed Object Systems, available at http://www.cs.caltech.edu/~kiniry/projects/Modeling98/submissions.html, 1998

[10] Interrante, L.D., Goldsmith, S.Y., "*Emergent Agent-Based Scheduling of Manufacturing Systems*", Agent-Based Manufacturing workshop, Agents98, 1998
[11] Jennings, N.R., Sycara, K., Wooldridge, M., "*A Roadmap of Agent Research and Development*", Autonomous Agents and Multi-Agent Systems Journal, 1, 5, Kluwer Academic, 1998
[12] Dorais et al, "*Adjustable Autonomy for Human Centered Autonomous Systems on Mars*", In, Proc. of the First International Conference on Mars Society, August, 1998.
[13] Brooks, R., "*Fast Cheap and Out of Control: A Robot Invasion of the Solar System*", need publisher, 1989.
[14] Waldo, J, et al, "*A Note on Distributed Computing*", in [2].
[15] Comments from Dr. Jacklyn Green, lead scientist on DS4, appearing in "Robot Futures", PcPro Issue 44, Dennis Publishing, UK, pp225-230, 1998.

URLs

[16] The Mobility Mailing List, http://mobility.lboro.ac.uk
[17] Aglets: http://www.trl.ibm.co.jp/aglets/
[18] Voyager: http://www.objectspace.com/voyager/
[19] Odyssey:http://www.genmagic.com/html/agent_overview.html
[20] Mole:http://www.informatik.uni-stuttgart.de/ipvr/vs/projekte/mole.html
[21] Concordia: http://www.meitca.com/HSL/Projects/Concordia
[22] The UMBC Agent Mailing list http://www.cs.umbc.edu/agentslist