Biologically Inspired Obsolescence Management in Mobile Agent Systems

A dynamic, service oriented approach

Duncan Anthony Coulter and Elizabeth Ehlers
Academy of Computer Science and Software Engineering
University of Johannesburg
Johannesburg, South Africa
dcoulter@uj.ac.za

Abstract— **Ubiquitous computing heralds an era marked by the increasing pervasiveness of computational hardware throughout a given environment. In order to capitalize on the increased abundance of the underlying infrastructure multi-agent systems will be required to reflect the characteristics of the ubiquitous networks upon which they operate. Due to the potential for limited communication capacity experienced by agents in the wild it will become increasingly important for mobile agents to migrate to devices in greater proximity to the problem upon which they are working or the resources they require. A result of highly mobile agents operating in potentially constrained computational and communication environments is that widely used command and coordination structures are no longer able to scale efficiently. Engineers are currently struggling with aspects of managing the physical devices which comprise such networks, particularly the obsolescence management of their constituent, highly dispersed, hardware. Analogously, the obsolescence management of deployed agents is of increasing concern. The paper examines and synthesizes several biological metaphors which may be employed in order to mitigate the inherent complexity of managing deployed mobile agent systems and presents this functionality in a service oriented manner.**

Biologically inspired computing, multi-agent systems, dynamic obsolesence management.

I. Introduction

The management of deployed systems is inherently complex. Even in the case of monolithic systems the heterogeneous nature of the hardware to which instances of the system are deployed coupled with a multitude of potential environmental configurations quickly poses serious challenges for ongoing system maintenance [11]. These problems are amplified in the context of highly distributed systems comprised of autonomous elements such as multi-agent systems.

The widespread adoption of the Internet has, on the whole, facilitated remarkable improvements in the management of deployed systems. Key beneficiaries of this have been systems which require regular critical updates such as antimalware suites and operating system security updates. Unfortunately these systems typically adopt a somewhat centralized approach analogous to the classical client-server model making them susceptible to the model's attendant performance bottlenecks and single point of failure concerns.

In order to address such performance, robustness and reliability issues deployed systems have employed techniques such as local caching, location aware redirection to mirror repositories and the incorporation of peer-to-peer content distribution overlay networks [8].

The suitability for such techniques to be applied within a highly distributed, mobile, multi-agent context is brought into question due to the highly temporal nature of the update data coupled with the communication constraints encountered by deployed agents within ubiquitous networks. This parallels the maintenance issues experienced in the field regarding the obsolescence management of a ubiquitous network's physical communication devices.

The paper begins with an examination of the life cycle of multi-agent systems within the commonly used Java Agent Development Environment (JADE) agent framework [1] and follows with an examination of how obsolescence management is achieved in the highly distributed biological context of multi-cellular organisms. Key principles are extracted from the biological basis and incorporated into a service-oriented model. Experimental results and conclusions drawn there from are presented in the final two sections.

II. Agent Life-cycle Management

A. Agent Life-cycle

The life cycle of a multi-agent system can be viewed from several levels ranging from the conventional life cycle of the underlying programming constructs (e.g. threads, fibers, processes, actors) through to the system as a whole. Of these layers only two are specific to the agent oriented nature of the system namely the management of an agent by the agent container and the management of the deployed system as a whole.

JADE-LEAP is the variation of the JADE platform specifically targeting mobile devices and the development of mobile agents. The major differences between LEAP and JADE on the implementation level include the use of the proprietary JADE Inter-container Protocol (JICP) together with

978-1-4244-7835-4/11/$26.00 ©2011 IEEE

removing the assumption of continuous communication between agent containers. Fig 1 illustrates the typical life-cycle followed by an individual agent on the JADE-LEAP platform.

Each JADE platform is fitted with a single instance of an agent which undertakes the role of Agent Management Service (AMS). The AMS is responsible for managing the life-cycle of an agent according to the agent management ontology.

Following instantiation agents typically register the services they provide with the Directory Facilitator (DF) agent as well as requesting any initial services they may require. Following that, for as long as the agent remains alive (has not invoked its own **`doDelete`** method or been terminated by the AMS agent) the agent will load and execute cooperatively concurrent behavior instances according to its design. These behaviors may result in inter-agent communication including further registration or deregistration with the DF agent, inter-container mobility via the serialization and deserialization of the agent instance or the dynamic instantiation of new behaviors.

B. An Infotaxic Mobile Multi-agent System Life Cycle

The behavior of individual multi-agent systems are extremely variable depending as they do on the nature of the goal the system is attempting to achieve coupled with the effects of design decisions such as the algorithms employed, the agent-wise decomposition of the problem and the social structure of the resulting agents. The life cycles of such systems still follows the general phases of the software development life cycle (Planning, Analysis Design, Implementation & Deployment, Maintenance) [11].

Treating the life cycle of multi-agent systems in exactly the same manner as conventional software fails then to consider some of the salient aspects inherent to those systems. The major source of difference is the post-deployment ongoing maintenance and regulation of a system composed of largely self directed, self deployable, self organizing entities. These

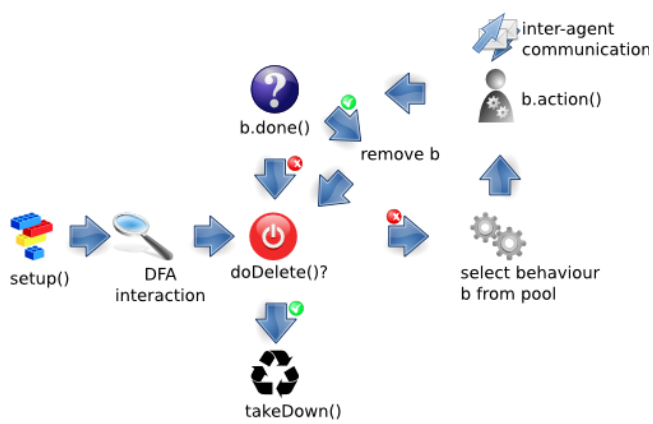

Figure 1. JADE-LEAP agent life cycle adapted from [1]

considerations are compounded with the fact that multi-agent systems may be composed of highly mobile agents operating on devices of limited communicative and computational capacity such as may be found in ubiquitous computational environments.

The behavioral characteristics of mobile agents operating in such environments will need to reflect these realities in that agents will utilize their mobility to better achieve their goals and hence the goals of the overall system. Agents' migration toward sources of required information will be increasingly common even in cases where the cost of transferring an agent's state is reasonably high.

The paradox of executing a high-cost migration in lieu of a moderate cost message transfer is a result of the intermittent availability of communication channels within ad-hoc network environments. Based on the information at their disposal agents should be able to optimize away some avoidable migrations, as it is impossible to determine in advance which migrations may be omitted agents will need to employ a degree of reasoning under uncertainty.

This tendency to favor migration toward pertinent information sources will be referred to in this text as infotaxis in an analogous manner to the behavior exhibited by biological organisms when the direction of their locomotion is governed by some external gradient for example chemotaxis, galvanotaxis and phototaxis are governed by chemical, potential difference and illumination gradients respectively.

Implementations of the multi-agent oriented search algorithm stochastic diffusion search often demonstrate this infotaxic property [3].

The complexity added by the increased motility of infotaxic agents is reflected in a modified system lifecycle for highly mobile multi-agent systems must accommodate the diffusion and ongoing regulation of such agents.

Such a life-cycle augments the deployment phase with dispersal and regulation phases which occur concurrently to maintenance of the system. Dispersal and regulation are antagonistic concerns which reflect the need for agents to both exhibit self directed infotaxic behavior balanced against the need of the system operator to coordinate the running of the system.

By placing maintenance at the same hierarchical level as dispersal and regulation reflects the need for maintenance to be an ongoing parallel activity in light of both the potentially long lived nature of mobile multi-agent systems and the intermittent availability of their members.

III. OBSOLESENCE MANAGEMENT IN BIOLOGICAL SYSTEMS AND THEIR AGENT ANALOGUES

Deployed systems of any non-trivial complexity suffer from the prospect of obsolescence at one level or another. A system or aspect of a system is considered obsolete when it no longer satisfies its intended purpose to an adequate degree. This may be due to the changes in the environment within which it operates, changes within itself or changes within the purposes for which it was created. Within the system itself those changes may occur at any level from the organization of the system through to the composition of its members down to the functioning of their constituent components.

The purpose of biological systems is to facilitate the replication of the genes of its members. This holds true from the genetic level through the cellular, tissue and organ level to the social interactions between organisms. When any of these levels fail to facilitate that goal they may be considered obsolete. For this reason there are mechanisms in place within biological systems to facilitate obsolescence management at all levels of a biological system.

A. Organisational Level

Obsolescence management at an organizational level pertains to the modification of the social interactions between members of that organization. This occurs within gregarious animal communities such as avian, canine, non solitary large felines and primates.

Typically these social changes influence the access to mating partners within the group and hence correct the ability of the organization to facilitate its mandate of genetic propagation. At the more complex level of human social systems the effects may be more rarefied such as resulting in a change in the economic dispensation or cultural attitudes within the group. Ultimately all of these effects may be traced to their impact on reproductive fitness.

Within software agent systems social changes may be accomplished via changes in either the communication patterns between agents or the roles played by certain agents. The impact of changes in communication patterns on evolvable multi-agent systems has been explored previously in the literature [2]

B. Organism Level

Obsolescence management within an organism tends to focus on the regeneration and replacement of tissues rather than on the organs and systems they comprise. Over time these replacements are unable to keep pace with the loss of functionality of these tissues and the overall utility of the organism decreases. The process of deleterious change over time which an organism undergoes is referred to as senescence. Eventually the number of harmful changes which have occurred render the organism unable to maintain homoeostasis and hence it perishes.

Selection within Darwinian systems operates only on those genetic changes which have an effect prior to the end of an organism's reproductive phase. For this reason mutations which provide early life benefit are positively selected for even if those same mutations may have profoundly negative later life consequences.

Due to the digital nature of multi-agent systems and the inherent error correction capabilities of many agent frameworks the probability of positive or negative changes to an agent's instruction set spontaneously occurring is rather low.

It should be worth noting that this probability is non-zero as the effects of external influences such as radiation have been shown to spontaneously switch the bit-state of volatile media. However the requirements that an agent must meet are highly mutable and as such any agent will eventually be rendered obsolete.

KEY OBSERVATION 1: Although agents are autonomous in terms of action selection their raison d'être is externally derived as they are intended to serve purposes beyond mere replication.

KEY OBSERVATION 2: Lacking any intrinsic mechanism for mortality an agent's life cycle must be extrinsically managed.

KEY OBSERVATION 3: Due to the increasing degree to which future systems will be composed of loosely coupled, autonomous components such as agents their obsolescence management is expected to become increasingly important.

C. Cellular Level

The life cycle of individual cells are predominantly internally managed, governed as they are by the cell's genetic code via a process of programmed cell death (apoptosis) and their eventual removal from circulation via ingestion by specialized phagocytic cells in a process called efferocytosis.

The triggering of apoptosis is thought to be linked to the terminal region of a cells genetic code (the telomere) which shortens following every cellular division and is renewed by the action of an enzyme (telomerase). When the rate of renewal falls below the rate of reproduction cells begin to undergo apoptosis.

Failure of a cell's apoptosis mechanism usually due to mutations in genetic code related to the production or responses to telomerase do occur and result in immortal, cancerous cell division.

Apoptosis and efferocytosis are unfortunately not sufficient biological bases for obsolescence management in multi-agent systems due to violations of both key observations presented so far. A cell's genetic instructions, including those which code for telomere renewal and apoptosis, are entirely internal and in general fixed for the life of the organism as a whole. In order to directly implement an artificial analogue to apoptosis an agent would need to be coded to recognize the conditions for its own obsolescence in advance.

Erythrocytes, also known as red blood cells, are the oxygen carrying component of blood. As part of their maturation and specialization process they lose the capacity for both self replication and to initiate apoptosis. Over time these cells begin to lose their capacity for carrying oxygen due to oxidative stress (i.e. their ability to fulfill their purpose).

The resulting chemical changes expressed on the surface of such cells mark them for ingestion by phagocytic [7] cells in organs such as the spleen

The lifecycle management of erythrocytes differs from other cells in general due to the fact that the recognition of obsolescence is externally imposed as opposed to resulting from an internal process such as apoptosis. A salient property shared by the both forms of efferocytosis is the fact that it is the transportation mechanism (the blood and lymphatic system) which in addition to conveying mobile cells to their destination also ensures that they are subjected to analysis of their obsolescence. An approach to mobile multi-agency should incorporate obsolescence management into the underlying transportation mechanism used by mobile agents. This is especially true in light of the fact that mobile agents may not always be contactable but a mobile agent which has elected to migrate from one host to another at least has an expectation of the availability of an open communication channel.

As mentioned previously an agent in a multi-agent system becomes obsolete when its behaviors no longer align with the overall goals of the system. The previous discussion focused on such changes occurring over time due to requirements drift and its analogue to biological senescence. An agent's behaviors may also cease to be aligned with the overall goals of the system due to a sudden, complete change in the goals of the system. Additionally the agent may cease to act in the interest of the system due to error related corruption of its instructions or indeed due to active interference and subversion by other agents.

Viral diseases achieve reproduction by way of inserting themselves into the DNA of their hosts and by extension co-opting the replication machinery of the host cell [12]. An infected cell no longer devotes its finite resources to achieving the goals of the multi-cellular system as a whole and as such should be withdrawn from circulation. A problem presents itself from the fact that the DNA of a cell is not directly visible making detection of subversion difficult. The adaptive immune system circumvents this problem by way of a bill-boarding mechanism using the molecule MHC (the major histocompatibility complex) which provides an external representation of the current DNA configuration of the cell and as such a way for immune cells to detect and engulf their infected counterparts.

Immune system cells are themselves subject to comprehensive life-cycle management based on their intended role in the overall system. For example instances of specialised BCells which have successfully bonded to an antigen are retained in the blood stream for many times longer than their typical life span so that they may function as memory cells by speeding up subsequent immune response to related pathogens at the risk of potentially stalling the immune system in a suboptimal configuration. At the other end of the scale, first line response cells such as the innate immune system's neutrophils have very short life spans fulfilling their role in stalling an infection while the adaptive immune response gets underway [12].

IV. The BIOMMAS Model

A. Problem Interpretation

The Biologically Inspired Obsolescence Management for Mobile agent Systems (BIOMMAS) model draws from both the processes of erythrocytic senescence and the adaptive immune system via a modified positive selection, BCell based algorithm.

Agents in the model are viewed as independent and self interested but with the provision that the utility function they are attempting to maximise is of benefit to the overall system. Agents operate within a network of computational nodes of limited computational capacity, constrained information resources and intermittent connectivity. These agents are mobile and are capable of migrating to other nodes within the network. The problem of maintaining the maximum population of agents with the highest utility (and hence maximise the degree to which the system's overall goals are met) becomes akin to the NP-Hard Knapsack problem with the addition of multiple knapsacks and self modifying, mobile items which are not always available.

Negative Selection

As implied by the previous statement it is expected that agents are able to modify their own behaviours in an attempt to reach maximum utility. It is also expected that such modifications, coupled with changing requirements may reduce the utility of agents dynamically.

In addition to responding to such changes with further self modification agents will attempt migration to neighbouring nodes based on the advertised computational and informational resources of those nodes. Agent migration is handled by the agent circulation service which accepts a serialised byte stream representing the agent's state and logical code.

The circulation service maintains a database of agent signatures which migrate due to low utility, should an agent (or closely related variation thereof) attempt migration frequently the implication is that the agent is no longer able to fulfil the goals of the system adequately for whatever reason (instructional genetic drift, changing requirements or active subversion by a third party). These agents are removed from the system by the circulation service in a manner akin to erythrocytic efferocytosis.

In those cases where an agent is capable of evaluating its own performance and deems itself no longer capable of meeting the system's goals the agent is able to remove itself from the system analogous to cellular apoptosis.

Those cases where an agent is able to self-diagnose its own inadequacy are obviously preferable to relying on removal during migration due to the lower overhead it imposes on the limited communication infrastructure of the network. However agents cannot be expected to correctly perform such an analysis in the long term due to the risks already identified (subversion, instructional genetic drift and changing requirements).

The circulation service is capable of more than merely removing frequently migrating agents from the system. Obsolescence management is identified as a cross-cutting concern in that it would impact across multiple layers of abstraction within the design of an agent.

The introduction of an aspect weaver as part of the circulation mechanism allows the system to dynamically inject current byte code instructions for performing apoptosis related checks. The combination of explicit removal by the circulation service and continuously updated, self removal analysis routines is intended to provide the negative selection pressure needed to curtail the system drifting towards negative gradients in its overall utility landscape. Externally imposed apoptosis is therefore identified as an aspect according to aspect oriented programming terminology [11].

POSITIVE SELECTION

The circulation mechanism described will place a negative selection pressure on poorly performing agents to become increasingly sessile in order to avoid removal from the system. Such strategies are combated via the main mechanics of the BIOMMAS model itself even though it is predominantly based on positive selection. The model attempts to address the problem of maintaining the highest utility population of agents in the network according to a variation of the clonal selection algorithm introduced by de Castro [5].

Clonal Selection models the role played by one particular class of lymphocytes (B-Cells) in the mammalian adaptive immune system. Any substance which triggers an immune response is referred to as an antigen and the degree to which an immune cell is able to bond with and hence recognize the substance is referred to as the affinity that the cell has with the antigen.

The chemical structure of the surface of a B-Cell and hence its affinity to various antigens is determined by a specific portion of its genetic code which is known as the V(C)j or variable region. The bone marrow produces large amounts of B-Cells with randomly initialized variable regions which as a pool have affinities to a wide variety of antigens which may or may not be present in the organism. The goal of the clonal selection mechanism is to increase the proportion of B-Cells which code for antigens which are actually present.

This process of maturation occurs by exposing the B-Cells to antigens in the system by way of specialized antigen presenting cells (APCs). A small subset of the large initial pool will express an affinity for those antigens. B-Cells with affinity values above a certain threshold will be selected for clonal expansion in accordance with the degree of their affinity. Clonal expansion involves creating exact duplicates of high affinity B-Cells.

A condensed presentation of the pseudo-code for the generic Clonal Selection Algorithm now follows [5]:

```
Lymphocytes ← marrow(RandomSeed)
while (not done)
foreach(lymphocyte in Lymphocytes)
  foreach(antigen in Antigens)
     affinities ← present(antigen, lymphocyte)
Subset ← selectHighest(affinities, lymphocytes)
Lymphocytes ← clone(subset, affinities)
Lymphocytes.map(mutate(affinities))
Memory ← selectHighest(affinities, lymphocytes)
```

Cells which do not express an affinity are not ignored however as the highest affinity expressed may not be the highest affinity possible. Cells whose affinity falls below the threshold have the variable potions of their DNA modified by a process known as somatic hyper-mutation in an exploratory attempt to cover more of the potential solution space. Following the maturation phase is the meta-dynamics phase in which very low affinity antibodies are completely re-randomized.

As illustrated in Fig 2 the BIOMMAS model adapts the basic clonal selection algorithm for use in a mobile, multi-agent environment as follows. Each node in the system takes on the role of APC and advertises the computational and informational resources available as an antigen. Affinity evaluation is computed as a result of the utility of the agent following its usage of the resources available at a particular node after a given period of time.

Some resources are obviously limited in terms of the number of concurrent users which may make use of it. Agents which fail to properly acquire resources are implicitly penalized due to the reduction in their utility values. The only active resource related intervention undertaken by the system is the periodic deallocation of resources from agents to prevent defective or malicious agents from starving other agents of resources. Agents are also periodically sent messages from neighboring nodes advertising the resources available at those nodes.

A side effect of these characteristics is that affinity evaluation is not guaranteed to be completely deterministic. This is however also the case in nature the evaluation of affinity relies on the intersection of complex three dimensional molecular structures which may be subject to issues of orientation and peculiarities of the folding of a particular protein.

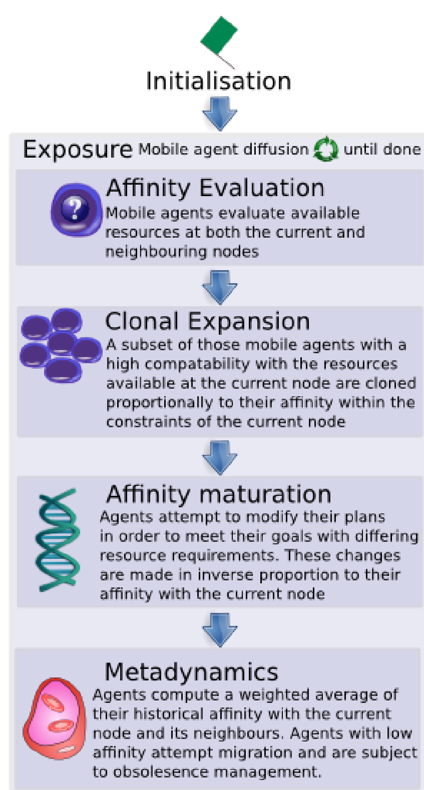

Figure 2. The BIOMMAS model as an instance of the clonal expansion metaheuristic

High utility agents are duplicated by the system as per the clonal expansion algorithm. Duplication of agents in a constrained environment will have an impact on the utility of other agents in the system including its own clones. The system will therefore favor lightweight agents which are less likely to suffer utility degradation due to being selected for cloning.

Somatic hyper-mutation is performed by way of requesting that they modify their behavior in an attempt to improve their resource utilization. Inspection of the resource utilization of an agents behavior amounts to the MHC bill boarding mechanism described earlier. Agents which do not comply will be removed by the AMS agent of the current JADE node.

The metadynamics phase of the algorithm differs in that those agents with very low affinity are presented with the option of either undergoing complete reinitialization or migrating to a neighboring node. Since an autonomous agent by definition is not required to comply with any request sent to it reinitialization amounts to the forced removal of the individual from the node and its replacement with a newly constructed one. Agents opting for migration are subject to the senescence requirements imposed on them by the circulation service described earlier.

The BIOMMAS approach therefore incorporates both positive selection (during its clonal expansion phase) and negative selection (as part of the senescence and apoptotic aspect injection mechanism of the circulation service).

V. Implementation Details

Implementation of a prototype system according to the BIOMMAS approach was realized in the Java programming language together with the Java Agent Development Environment with the following species of agent. `BiommasAgents` facilitate the management of each node's local clonal selection algorithm. There is only one instance of a `BiommasAgent` at each node. A BIOMMAS node is considered to be equivalent to a JADE agent container for the purposes of the prototype. As part of the initialization of a new JADE container a `BiommasAgent` is instantiated. This `BiommasAgent` registers with the local directory facilitator agent that it offers an obsolescence management service.

`BCellAgents` represent the consumers of the computational and information resources offered by nodes in the service. The instantiation of a new `BCellAgent` at a node is brought about either due to this being its initial instantiation, clonal expansion or its arrival from an adjacent node. The first action of a new `BCellAgent` is to query the DFS for any obsolescence management services being offered and to register itself with the local `BiommasAgent` via a relaxed version of the FIPA Subscription Protocol [1].

Agents are prevented from circumventing the need for registering with the local obsolescence management service due to the fact that access to the local node's resources is contingent upon such registration. Furthermore the only mobility service offered between participating JADE containers is the circulation service which integrates both the extrinsic and intrinsic obsolescence enforcement.

Once the `BCellAgent` has subscribed to the obsolescence management service it evaluates the resource bouquet being made available at the current node which is equivalent to the exposure of an immature BCell to an APC during the maturation phase of a primary immune response. In order to maintain their subscription to the circulation service and hence maintain their mobility in the face of a dynamic resource landscape a `BCellAgent` must periodically report its utility to the local `BiommasAgent`.

In the case of self interested agents the actual resource utilization may be measured directly by the `BiommasAgent` and weighted against the agent's reported utility. Agents which report a low utility while maintaining a high resource utilization are given a reduced ranking during the meta-dynamics phase of the algorithm and are thus more likely to undergo mutation or forced migration. Rational self interested agents will therefore be compelled to limit unwarranted resource utilization due to the negative survival pressure which is applied to such behavior.

In order to facilitate inter agent communication the system makes use of a subset of the FIPA performative communication types which are based on Speech Act Theory [10]. The following semantic interpretations of the general performatives apply:

- CFP: The call for proposals perfromative has been semantically repurposed in order to signal a change in the overall objective of the system. Following such a signal the likelihood that a given `BCellAgent's` current computational node will have the exact resources to meet the demands of the new objective are low. CFPs are hence a driver for mutation and migration behavior.
- INFORM: Used by `BCellAgents` to update the local `BiommasAgent` with the sending `BCellAgent's` reported utility.
- PROPOSE: Updates a `BCellAgent` with the available resources at the current computational node.
- PROPOGATE: Used by a `BCellAgent` to request a transfer to another node.

Apart from the speech act itself messages sent between agents and service realizing agents contain additional content. For the purpose of the prototype system this content took the form of serialized byte code streams which represented Java classes unique to the prototype itself. This design decision obviously limits the cross platform communication capacity of the system as serialized objects are not guaranteed to be restorable on different JVM implementations or even versions.

In addition the `setContentObject()` method of the `ACLMessage` class is unique to JADE which limits communication with other FIPA compliant agent systems. In order to address these issues future non prototype versions of the system will need to establish a formal ontology for obsolescence management and constrain any content oriented portions of their messages to open XML based formats.

The system implementation allows for a greater degree of parallelism than the conventional clonal selection algorithm of de Casto [5]. This is due to the fact the agents each operate with their own thread of control and indeed operate concurrently on separate computational devices. Due to this fact a greater search space can be explored than would otherwise be possible particularly in light of the impending physical limitations which must constrain the observed transistor density trend known as Moore's Law [9].

The implication of this is that at any given point in time various nodes will be undergoing different phases of the clonal expansion algorithm. This is more in line with the biological basis for the metaphor as the maturation of immune cells within germinal centers is ongoing without the need for a continuous synchronization signal [7].

VI. RESULTS

Computational nodes were arranged in a linear topology with random periods of communication permitted between non contiguous nodes. In cases such as these the emergent properties of the underlying topological definition function become significant [4]. . The simulation assumes that the computational resources are fungible and accurately reported by each node.

The simulation was run on a network of 20 nodes with an initial population of 200 `BCellAgents` prior to clonal expansion. Each node contained a vector of computational resources which where randomly initialized at the start of the simulation and subject to periodic alterations at run-time.

Each `BCellAgent` exposed its required computational and communication resources via a publicly visible vector of resources needs. This vector is analogous to the MHC molecule exposed by lymphocytes. This vector is only mutable by the agent itself. The agent's affinity for the local resource bouquet was computed by a selection of similarity measures depending on the objective of the system namely the Hamming Distance or alternatively the sum of the absolute differences between each of the agent's needs and the local resource pool.

The first similarity measure may be used in cases where agents are deemed to be unable to make use of mismatched resources at all whilst the second may be used in cases where agents are deemed to be able to make some use of sub optimal resource allocations. Resources were represented as multi-sets allowing the system to keep track of the degree of utilization of each resource and apply mutual exclusion or limited availability rules as needed.

The results presented in Fig 3 shows an archetypical run of the prototype system with increasing propensities towards migration or mutation. The data show the punctuated equilibria [6] which are typical of the improvement patterns brought about by evolutionary systems or their computational equivalents.

The lower the metadynamic threshold (M_τ) the greater the probability that agents will undergo obsolescence management P(O).

$$M_\tau \ \alpha \ P(O)^{-1} \qquad (1)$$

Since there are long periods of stable behavior occurring - concurrently across multiple nodes it is difficult to know when data collection should occur. It is for this reason that the X-axes of the graphs are labeled Plateau Selection Point as the system's data collection routines were triggered based on the volume of `ACLMessage` traffic which is an indicator of the immune response's intensity.

An absolute timer does occasionally mandate data collection ensuring that steady state periods are recorded but do not dominate the dataset. The end of plateau based sampling does mean that the convergence rate appears to be slightly slower than it would otherwise be due to the increased proportion of data recorded during periods of high system activity.

The Y-axis is a measure of the overall utility CU of the system as a function of the degree to which the resource needs of each agent (r_a) are matched by the available resource boquet of the node it currently occupies (r_n) over the total number of nodes (t) and agent population per node.

$$CU = \sum_{n=0}^{t} \left(\sum_{a=0}^{p} \frac{r_a}{r_n} \right) \qquad (2)$$

The Hamming Distance was not used as a similarity measure in these three runs as it is the expectation of the authors that real world agents will attempt to make use of what limited resources are available in pursuit of their own goals.

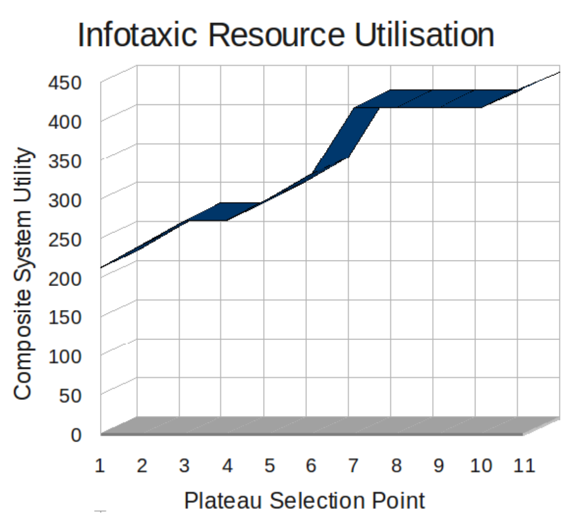

Figure 3. Archetypical run of prototype system illustrating the punctuated equilibira of stable resource utilisation.

VII. CONCLUSION

The paper presented a biologically derived exploration of obsolescence management in the context of mobile information and computation seeking agents operating in resource constrained environments. The unification of obsolescence management and mobility proposed by the paper ensures that agents whose goals do not align with those of the system were placed under selection pressure to either curtail their own resource usage or face removal from the system or failing that a lack of mobility and or resource access.

A service oriented prototype system was developed which recast distributed resource allocation in mobile multi-agent systems as a variation of the clonal expansion immune algorithm. The introduction of a unified resource allocation, mobility and obsolescence management service is a novel contribution of the work and may be applied using metaphors other than the mammalian adaptive immune system.

The work assumed that each agent in the system had a desire for mobility that was solely motivated by the resources available at the destination node and that no other exigencies where in effect. In a real world ubiquitous network the power reserves of a computational node may also be a factor and unexpected drains on such a resource may force migration in spite of otherwise plentiful resources.

This work relates to generic software life cycle management [11] and due to its metaphorical basis in the adaptive immune system coupled with the management of deployed systems shows that the immune system serves well as a source of inspiration to tasks within the management of deployed software systems. This is believed to be the first time that these technology sub-branches have been combined for this purpose.

ACKNOWLEDGMENT

Public domain graphical elements were used in the creation of Figs 1 and 2. The authors express their gratitude to the JADE community for making an open source FIPA compliant agent library freely available.

REFERENCES

[1] F. L. Bellifemine, G. Caire, and D. Greenwood. “Developing Multi-Agent Systems with JADE” (Wiley Series in Agent Technology). Wiley, 2007.

[2] D. Coulter and E. Ehlers. “Federated patent harmonization: An evovable agent-wise approach”. Proc. Tools and Methods of Competitive Engineering 2008,. Delft University of Technology, Izmir Turkey, 2008, vol 2, pp 1391-1393.

[3] D. Coulter and E. Ehlers. “Immunity amplified stochastic diffusion search” in Advances in Practical Multi-Agent Systems. Studies in Computational Intelligence Vol 325, pp. 84-89, 2010.

[4] D. Coulter and E. Ehlers. “Collaboration coordination via cellular automata” in Tools and Methods of Competitive Engineering April 2010,Ancona Italy, Delft University of Technology, Vol 1, pp 867-878.

[5] L. N. de Castro and J. Timmis. “Artificial Immune Systems: A New Computational Intelligence Approach”. Springer, London, 2002.

[6] S. J. Gould and N. Eldredge. “Punctuated equilibrium comes of age” in Nature, vol 36, 223-237, November 1993.

[7] P. T. Kenneth M. Murphy and M. Walport. “Janeway's Immunobiology”, Seventh Edition. Garland Science, 2007.

[8] A. Lakshman and P. Malik. “Cassandra – a decentralized structured storage system” in Proc, LADIS 2009. ACM, 2009.

[9] G. Moore. “Cramming more components onto integrated circuits” in Electronics, 38(8), April 1965.

[10] J. R. Searle. “Speech Acts”. Cambridge University Press,1969.

[11] I. Sommerville. Software Engineering (9th Edition). Addison Wesley, 2010.

[12] B. A. O. Thomas J. Kindt and R. A. Goldsby. “Kuby Immunology”. W. H. Freeman, 2006.