A Survey on the Use of Cultural Algorithms in Multi-Agent Systems

Shiven Sharma

TABLE OF CONTENTS

Abstract

Cultural Algorithms simulate cultural evolution, bringing about a more comprehensive learning and evolution than simple biological evolution. They can be used in both static and dynamic environments, and in complex multi-agent systems to provide effective simulations of learning procedures. This survey discusses the use of Cultural Algorithms in multi-agent systems at both academic and industrial levels. This survey also discusses the swarming behaviour at the cultural level and the interaction between various knowledge sources, which provides the theoretical basis for the working of Cultural Algorithms.

1. Introduction

Cultural Algorithms were introduced by (Reynolds, 1994) as a means of simulating cultural evolution. He argued that adaptation and evolution amongst individuals of the population takes place faster using cultural evolution as compared to using biological evolution. Evolution takes place at both the *cultural level* (the belief space) and the *population level* (for each individual). The belief space is the knowledge that is shared amongst the agents in the population. This model of dual-inheritance is the key feature of Cultural Algorithms, as it allows for a two-way system of learning and adaptation to take place. In a dual-inheritance system, the fit population members, as selected by an *acceptance* function on the basis of a fitness value, add their knowledge and experience to the belief space, thereby sharing it with all other agents in the environment. The belief space knowledge in turn helps guide the agents in the population from one generation to another. Other evolutionary approaches, such as Genetic Algorithms, allow for evolution to take place only at the individual (or population) level, i.e., they do not support a dual-inheritance system.

(Reynolds and Saleem, 2003) showed that cultural learning takes place using three distinct phases of problem solving. They defined them as coarse-grained, fine-grained and backtracking. They also discovered five types of knowledge, namely normative (ranges of acceptable values), topographic (representing spatial patterns), situational (successful and unsuccessful instances), historic (temporal patterns) and domain (relationships and interactions between domain objects) knowledge. Each of the phases has a dominance of one type of knowledge over the other.

The purpose of this survey is to examine the usage of cultural algorithms in complex multi-agent systems. In chapter 2, the existence of cultural swarms is discussed, which demonstrates the underlying learning and interacting mechanisms between the various knowledge types in the belief space. Recent work done by (Reynolds, Kobti, Kohler, 2003) uses cultural algorithms to simulate the early Anasazi settlements and answer questions regarding their disappearance from the Mesa Verde region, and is discussed in chapter 3. This work is an example of the use of Cultural Learning in a complex multi-agent system, in which many different factors affect the manner in which the population evolves. Cultural algorithms have also been used in large-scale multi-agent systems in industry, as shown in (Rychtyckyj, Ostrowski, Schleis, Reynolds, 2003) and (Kobti, Rahaman, Kent, Snowdonw, Dunlop, 2005). These, and other advances, are discussed in chapter 4. In chapter 5, a summary of the work and concluding remarks are provided.

2. Cultural Swarms: Swarming Behaviour in the Belief Space

In this chapter the existence of swarming behaviour between the various knowledge types in the belief space is discussed. The hypothesis was proposed by (Reynolds, Peng and Brewster, 2003) when (Reynolds and Saleem, 2003) discovered problem solving phases emerge in the belief space. (Iacoban, Reynolds and Brewster, 2003) prove the existence of cultural swarms in the belief space by performing experiments using a population represented by a Canonical Particle Swarm model.

2.1 Swarm Emergence

(Reynolds and Saleem, 2003) discovered the emergence of three phases of problem solving while using an Evolutionary Programming model to represent the population in a Cultural Algorithm framework. Each of the phases is dominated by a knowledge source which is most successful in generating good solutions. They observed that during the start, the coarse grain phase is the most dominant, especially when the problem landscape alters dynamically. The authors observe that topographic knowledge is the most successful, generating the best solutions 50% of the time, followed by situational knowledge. The fine-grained phase is dominated by situational knowledge, whereas in the backtracking phase, all knowledge sources contribute equally in producing the best solutions, the exception being historic knowledge in static environments because temporal patterns remain the same.

(Reynolds, Peng and Brewster, 2003) discuss whether the emergence of the above phases is a result of the operation of the Evolutionary Programming model to represent the population, or whether it is a direct consequence of the problem solving process undertaken by the agents. They also observe that certain sequences of operator usage emerge, which lead them to believe that there might be a higher level interaction going on between the various knowledge sources in the belief space which are guiding these usage of operators. These observations lead to the authors to contemplate the existence of a Cultural Swarm, i.e., swarming behaviour observed at the cultural level. They believe that various knowledge sources are interacting at the cultural level, with each knowledge source guiding problem solving decisions, the results of which can be exploited by other knowledge sources.

In the following section, experiments performed to test and verify the existence of Cultural Swarms are presented.

2.2 Experiments for Swarm Existence

In order to test the swarm hypothesis, (Iacoban, Reynolds and Brewster, 2003) use a Canonical Particle Swarm Model given by (Eberhart and Kennedy, 1995) to represent the population model, instead of an Evolutionary Programming model. The Particle Swarm Optimiser (PSO) algorithm works by initially distributing the particles (individual agents) with a random location and velocity over the function space. The position of the particles is updated at each time step by using its velocity, knowledge and a direction. Over time, the particles move to the best areas in the search space by using the knowledge of the most successful particles. The landscape to be searched is generated by using a random problem generator given by (Morrison and DeJong, 1999), and consists of cones of random heights.

Situational knowledge in the belief space is a list of exemplar individuals, each having the parameter values for the function and a fitness value. The knowledge is updated by either adding the best individual's knowledge if it is fitter than the current best, or by reinitialising it upon changes in the environment. The domain knowledge is represented by the domain ranges of all parameters and the best instances from the population. The update mechanism for this is similar to that for situational knowledge, except that the knowledge is not reinitialised after every environment change. The change in the fitness value (i.e., the difference between the value of the current best and the best found so far) is used to generate a mutation step size relative to the change magnitude, which is then mapped to the variable range. Normative knowledge is represented by a set of intervals that correspond to the range which is believed to be the best for each parameter. Each variable has an upper and lower bound for its values and its performance values. Historic knowledge consists of sequences of changes in the environment regarding shifts in the distance and direction of the optimum. It contains the average of these distances and directions, and a list of changing events over the sliding events window. To record environment change, the current best solution is recorded with the difference of the direction shift between it and the previous best solution. This can take the values of 1, -1 or 0, depending on an increase, decrease or no change in the values of the parameter. Topographic knowledge is represented as a multi-dimensional grid, with each cell described in terms of the number of dimensions and its size. The knowledge is created by sampling a solution with each cell, and then creating a list of the best cells as returned by the sampling. Updating is done by dividing a cell into smaller cells when the fitness value of an individual is better than that of the one in the previous generation. The sampling takes place again as described above, and the list is updated. Influence functions are created in such a way that each type of knowledge has a corresponding influence function. A roulette wheel mechanism is used for selecting influence functions, with the likelihood for selection dependent on the area of that function under the wheel.

With the above setup, (Iacoban, Reynolds and Brewster, 2003) have tested the swarm hypothesis by running the model over 10 different random configurations of the conic landscape, and notice identical phases of problem solving emerge, as discussed in (Saleem and Reynolds, 2003) and (Saleem, 2001). They observe that topographic knowledge first identifies the most promising regions in the search landscape and then propagates this knowledge to other sources so as to guide them. Situational knowledge then fine tunes the regions, and then topographic knowledge subdivides that region further into smaller regions. This early-on cycle of shifting back and forth serves to guide other knowledge sources into finding the optimum value in the landscape. Therefore, it is seen that topographic knowledge takes the lead towards the optimum, and the other sources follow it, guided by the knowledge propagated and shared between them all. This leads to the emergence of the three problem solving phases as discussed previously in the belief space. These experimental results lead them to believe in the existence of Cultural Swarms in the belief space, thus supporting the Swarm hypothesis.

2.3 Discussion

From the work done by (Iacoban, Reynolds and Brewster, 2003), the researchers believe that swarming behaviour in the belief space leads to the emergence of problem solving phases, with information being shared and propagated between all the knowledge sources. Which source takes the lead is dependent on the problem being solved, but the overall conclusion made by the authors is that swarming does take place. The importance of this discovery lies in that fact that new formulae and rules for the meta-level knowledge can be formulated with these observations that can be considered as extensions to the Schema Theorem given by (Holland, 1975). Work done by (Reynolds, Peng and Brewster, 2003) also showed evidence of emergence of a best-first search of the problem state space, giving rise to the conclusion that a virtual algorithm emerges as a result of knowledge interaction. Work is now being done by the authors to explore this property, answering questions such as what sort of preconditions for the population and belief space to give rise to such an algorithmic search for solutions, and the effect of state knowledge in the flow of the search.

Year	Paper Title and Authors	Major Contribution
2003	The Impact of Environmental Dynamics on Cultural Emergence, *R. Reynolds, S. Saleem*	Investigates the problem solving phases and interactions between the various knowledge types in the belief space.
2003	Cultural swarms: knowledge-driven problem solving in social systems, *R. Reynolds, B. Peng, J. Brewster*	Investigates the Cultural Swarm hypothesis by describing a population as a Particle Swarm, and examining the interactions between the knowledge.
2003	Cultural Swarms: Assessing the impact of Culture on Social Interaction and Problem Solving, *R. Iacoban*, *R. Reynolds*, *J. Brewster*	Experiments are performed to prove the existence of swarming behaviour between the various knowledge types in the belief space.
2003	Cultural swarms II: virtual algorithm emergence, *B. Peng*, *R. Reynolds*, *J. Brewster*	Discusses the emergence of virtual best first search of the problem space as a consequence of interactions between the various knowledge types in the belief space.

3. Historical Reconstruction using Cultural Algorithms

In this chapter, the use of cultural algorithms in simulating a large-scale multi-agent model of the pre-Hispanic population in the Mesa Verde region is discussed. The model is built upon the formulation given by (Kohler, 2000). The aim is to answer questions regarding their lifestyles and their eventual disappearance from that region. As stated by (Kobti, Reynolds and Kohler, 2003), using a cultural algorithm framework allows for the knowledge of each individual agent community to be shared by the overall population, and also allow for the global knowledge to determine the evolution of that population. Such a formulation allows for the addition of various social, economic and biological factors into the population and observe and track the eventual evolution as a result of these additions, as shown in (Kobti, Reynolds and Kohler, 2004) by adding economic exchanges based on relationship between agents, and in (Kobti and Reynolds, 2005) by adding animal food resources into the environment.

3.1 The Problem Description and Formulation

(Kohler, 2000) had designed a model to simulate the lifestyle of the early Anasazi settlement in the Mesa Verde Prehispanic Pueblo region. The aim of that model was to understand the reasons which led to their disappearance from that region. (Kobti, Reynolds and Kohler, 2003) transform this model into a multi-agent simulation embedded in a Cultural Algorithm framework, by using a population of social agents to represent the settlement population. The purpose of this is to embed social interaction and cultural learning into the system, and see how various socio-economic factors affect the population.

In the multi-agent simulation, the agents consist of a single household, which in turn is comprised of its members, each having an age and a gender. Over time, each household in the social network develops its own culture based upon its situation knowledge and behavioural characteristics that enhances its survival in the environment. The belief space is shared by all households, and communicates the knowledge and experiences of each household across the network. In addition to the global knowledge, each household also has a local knowledge. This is comprised of a move radius which determines the maximum distance a household can travel when searching for a new location to reside in, a move frequency which determines how often an agent can move in a single year, and a move trigger which is used to decide whether or not a household should move.

Kinship ties, which were missing in Kohler's model, are maintained in the multi-agent simulation. Rules regarding marriages are controlled, which lead to formation of new households. New households resulting from marriages are initially located in the same location as one of the parent household. Each household thus comprises of a husband, wife and children, all of whom are identified by unique tags, with links relating households formed as a result of marriages. Concepts such as neighbours and friends are also maintained using similar links.

3.2 Adding cooperation amongst agents

With the formulation ready as described above, several socio-economic factors are added to see their effects on the population. In (Kobti, Reynolds and Kohler, 2004), reciprocal exchange strategies are explicitly embedded into the simulation. The strategies are based upon the exchange of goods and services (food in this case) between various agents. The four strategies used are:

1. No exchange of food.
2. Food given to an agent when in need (shortage of food).
3. Food given by an agent to those in need (excess food).
4. Both 2 and 3 enabled together.

A finite state machine is used to model the states a cooperating agent goes through. The various states are:

1. Satisfied: The agent has enough food resources to meet the needs of the entire household.
2. Philanthropic: The agent has a surplus in food resources.
3. Hungry: A state just before starvation. Triggered when agent is on its last ration of food.
4. Critical: Agent has insufficient food, and must ask for food.
5. Death: An agent unable to meet the food needs dies, and is removed from the system.

(Kobti, Reynolds and Kohler, 2003) allowed agents to randomly select other agents to cooperate with. In (Kobti, Reynolds and Kohler, 2004), agents are allowed to select those with whom they wish to cooperate, thus forming generalised beliefs in the belief space regarding their preferences for cooperation. Initially, an agent has equal probability to select all other agents for cooperation, but as time progresses, it keeps track of those with whom it has successful interactions. This biases the selection of agents for cooperation in the future. This is especially essential when an agent is in critical stage, and faces starvation and/or death, and the only thing that can save it is selecting the right agent who is willing to lend it food.

Learning is done by using roulette wheel selection. Agents are assigned an area in the wheel which represents the likelihood of them being selected for cooperation. Successful interactions increase this likelihood. This learning takes place at the individual level. At the cultural level (the belief space), normative and situational knowledge are adjusted dynamically. Situational knowledge corresponds to individual exemplars which have been most successful in requests for cooperation. The normative knowledge corresponds to the frequency pertaining to the kinship types that have been most successful in cooperation. Both of these collectively work to guide the selection of agents for cooperation. At the individual level, an agent also maintains in its memory the last agent with whom it had successful cooperation. Thus, knowledge at both the global and individual level guide the selection of agents for cooperation. Experiments done on this show that addition of learning makes the system more resilient than when random cooperation strategies were used. However, if the agents are located close to each other, movement by one can cause the others to follow, thereby creating system stress. This, as suggested, can be avoided by having the agents spread out more across the landscape.

In the following section, addition of food resources in the form of deer, rabbits and hares, and their effect on the population is examined.

3.3 Adding food resources in the population

(Kobti and Reynolds, 2005) further update the system developed in (Kobti, Reynolds and Kohler, 2003) by adding food resources such as deer, rabbits and hares, thereby giving the agents an opportunity to hunt and share their resources. They aim to examine the effects the wildlife (representing the food source) had on the populations in the Mesa Verde Region, an whether it was a significant factor in their disappearance from that region. The ability of agents to hunt the resources is based on Charnov's Marginal Value Theorem, given in (Charnov, 1976). The kinship networks as introduced in (Kobti, Reynolds and Kohler, 2004) are further examined in detail. The overall social network is subdivided into three types of networks, each representing a specific social model. The kinship network represents the interactions between agents in a household and neighbours. The exchange type dominant in this network is that of Generalised Reciprocal Exchange, which involves exchanges without the need to reciprocate. This allows for the population to move to areas where productivity is high, which is a direct consequence of relocating closer to those agents who are productive (as they are able to provide goods). This network is also termed as a Generalised Reciprocal Exchange Network (GRN).

Archaeological evidence showed a large amount of pottery, tools and wooden goods in the settlement areas, leading to the belief that these were traded between individuals. This is simulated in the multi-agent model as an economic network. To accomplish this, each household agent maintains a list of agents that it considers to be trading partners, all of whom are independent from the kinship network described above. The agents keep a track of the experiences of the transactions between each other in order to generalise about their trading partners. Their ability to pay up their debts to other agents reflects their reliability in trading. This is maintained as a households reputation; a natural consequence of this is that a household with an excellent reputation is one which has no debts. A cultural algorithm is used with this network as it helps guide agents to make decisions regarding their trading partners, thereby increasing their chances for survival. This network is also called the Balanced Reciprocal Exchange Network (BRN). Hub networks are formed by those households that contribute to both kinship and economic networks. In other words, these networks consist of households that have a number of links to other households. The importance of households is calculated using a probabilistic discrete Poisson distribution based on the number of links it has with other nodes. Based on this, households can be promoted and demoted to and from the hub network.

Both the GRN and BRN can be used by households to help them meet their caloric needs. For instance, when a household enters a critical stage and is in need for resources, the agent tries to call in debts from the agents in its neighbourhood. This is done through the BRN. If this approach is unsuccessful, it tries to ask its relatives for help. This is done through the GRN, as this request is a one-way request in which the agent will not return anything back. In the eventuality that this also fails, the agent keeps searching its economic network till it is able to satisfy its need (or till it expires).

In order to hunt resources, the agent households store knowledge regarding the yield (the amount of resources) present in the various location (cells) in the landscape. Cells which have a high yield are arranged in a best first manner in a list. These lists are the hunting plans, and are processed using the Marginal Value Theorem developed by (Charnov, 1976) and (Stephens and Krebbs, 1986). Based on this approach, if a cell has a yield which is lower than what is expected by the agent, it is removed from the list and replaced by a cell which is within a certain distance from that cell. This distance can either be a fixed parameter, or can be learnt over time by the population. Experiments run on the simulation shows that the population is highly sensitive to the availability of food resources. Agent households that are unable to meet the food requirements on their own use the GRN and BRN networks to initiate exchanges. It is also observed that environmental stresses on the system can cause the population to move away from the region, as these cause them to be unable to sustain their food requirements.

3.4 Discussion

The Cultural Algorithm framework here works well with the simulation, as it provides for collective sharing and distribution of knowledge between the households. The work done in (Reynolds, Kobti and Kohler, 2003) introduces the concept of using cultural algorithms to simulate the village environment, which lays the foundation for significant work in answering the historical questions posed by (Kohler, 2002). The framework allows for constructing a small world network, in which ties between households are maintained and remembered, and the GRN and BRN allow the agents to learn with whom they should interact and make transactions. The major contribution of the researchers work is that it allows for a realistic simulation of a real world environment by using the dual-inheritance system of Cultural Algorithms. Using such a simulation, they are able to understand the lifestyles of ancient societies, and understand how social, economic and natural factors affected their daily lives. Future work by (Reynolds, Kobti, Kohler and Yap, 2005) involves adding various environmental factors into the system, such as fluctuations in the flora and fauna and erosion, and observing their effects on the population and thereby find answers to questions regarding the disappearance of the actual population in the Mesa Verde region.

Year	Paper Title and Authors	Major Contribution
2003	A Multi-Agent Simulation using Cultural Algorithms: The Effect of Culture on the Resilience of Social Systems, *Z, Kobti*, *R. Reynolds*, *T. Kohler*, 2003	A multi-agent cultural framework is proposed to describe the population of pre-Hispanic Pueblo Indians in the Mesa Verde region and their corresponding belief space.
2005	Modelling Protein Exchange across the Social Network in the village Multi-Agent Simulation, *Z. Kobti, R. Reynolds*	Functions are added into the multi-agent framework to facilitate economic exchanges between various agent households based upon relationships between them.
2005	The Effect of Kinship Cooperation Learning Strategy and Culture on the Resilience of Social Systems in the Village Multi-Agent Simulation, *Z. Kobti*, *R. Reynolds*, *T, Kohler*	Co-operation strategies and animal food resources are added into the multi-agent framework in order to see the behaviour of the population and their subsequent evolution.
2005	Unraveling Ancient Mysteries: Reimagining the Past Using Evolutionary Computation in a Complex Gaming Environment, *R. Reynolds, Z. Kobti, T.A. Kohler, L. Y. L. Yap*	A "Trust in Networks" gaming paradigm is added into the system to examine the levels and extent of co-operation between individuals in a GRN and a BRN.

4. Industrial Multi-Agent Systems and Cultural Algorithms

Evolutionary computation has been used extensively in solving real-world problems, specifically in industry, as given in (Dasgupta and Michalewicz, 97). (Rychtyckyj, Ostrowski, Schleis and Reynolds, 2003) outline areas in which Cultural Algorithms have been used. One approach they discuss is in knowledge base re-engineering. Cultural Algorithms have been used for re-engineering decision tree-based knowledge discovery system, as given by (Al-Sheher, 1997) and commercial rule based systems, as given by (Sternberg and Reynolds, 1997).

(Ostrowski, Tassier, Everson and Reynolds, 2002) discuss the usage of Cultural Algorithms to evolve pricing strategies for vehicle sales. The evolution of an optimal pricing strategy dependent on the economic environment will allow for the greatest profit to be generated for the company by ensuring maximum sales. They develop a multi-agent system to model a real-world consumer market, called the Original Equipment Manufacturer (OEM). They design a Dual Cultural Algorithm with Genetic Programming (DCAGP) which integrates both the software engineering techniques of black and white box testing. For the first Cultural Algorithm, which models the white box phase, the initial population consists of a number of programmes which are determined from the problem specification and the identified constraints. The belief space is generated using the structural components that are defined in the initial population. With the population and the belief space ready, the first cultural algorithm for white box testing is started. The population members are evaluated using a performance function, and the top quartile are applied to the belief space, which in turn influences the next generation. This process continues till the performance stabilises. In the second Cultural Algorithm, modelling the black box phase, the population is generated using the implicit constraints given in the white box belief space. The belief space is then initialised by taking the third quartile of the population as ordered by their performance. The reason for this choice is that these individuals are those which show potential to be the best solutions, but which also posses flaws that need to be identified and fixed. Variation operators are applied to the individuals, and the algorithm runs till the performance stabilises. Upon termination, the black box population is applied to the white box belief space, and, if the termination condition has not been met, the process repeats itself.

To test their theories, the researchers use MarketScape, a Java-based multi-agent framework to model economic environments. Four object types are represented in it, namely consumers, vehicles, manufacturers and dealers. A memory-based strategy is used, called "postpone strategy" that relates to consumer expectations on price falls. Normative knowledge is used in the white box belief space to represent intervals for the values in the MarketScape equations. Normative knowledge is also used in the black box belief space, and represents changes in the step directions. It is observed that using the DCAGP algorithm, the highest optimal strategy is generated, and is done so in a significantly less time than when using a Genetic Programming approach. (Ostrowski and Reynolds, 2004) also use the DCAGP system in a model using economic recession, and observe similar results.

(Rychtyckyj, Ostrowski, Schleis and Reynolds, 2003) use the Direct Labour Management System (DLMS), used by Ford for managing vehicle assembly , for re-engineering using Cultural Algorithms. Re-engineering done manually takes a lot of time and resources, and the authors aim to simplify and automate the process in an intelligent manner by employing Cultural Algorithms. They use both a top-down and a bottom-up approach to attack the problem. They re-engineer the semantic network to make it less complex and more efficient so that the efficiency of the subsumption algorithm increases. Firstly, they apply Cultural Algorithms for re-engineering the DLMS semantic network using a top-down approach. The search space used by the algorithm consists of sub-trees representing sub-structures in the semantic network, each of whom are evaluated using a performance function that assigns them a reward. This reward takes into consideration how well the sub-trees are able to preserve the structure of the network in terms of knowledge retrieval, but at the same time how much decrease they cause in the cost of subsumption. Fit individuals in the population influence the belief space with their characteristics. The belief space then uses these characteristics and guides the learning from one generation to the next by passing them on so that better solutions can be evolved. The results showed that the resulting network is as good, and in some cases better, than those which were created by human developers. A top down approach for re-engineering the DLMS system using Cultural Algorithms has also been discussed in (Rychtyckyj and Reynolds, 2002). Bottom-up approach for re-engineering the DLMS system have been discussed in (Rychtyckyj and Reynolds, 2001). The Cultural Algorithm classifies input concepts into clusters which are considered most efficient for subsumption and classification. The advantage to this approach is that the new semantic network is constructed without relying on previous design information. The results of using the bottom-up approach are compared to that of using human developers and it is observed that using the former causes a greater reduction in costs of subsumption. In comparing the bottom-up approach to the top-down approach, it is observed that the former causes about five times more reduction in the cost of subsumption, however, it also causes some loss of useful knowledge. The researchers conclude that using both approaches to re-engineer the semantic network is the best approach and causes significant reduction in costs while preserving the integrity of the network.

(Kobti, Snowdon, Rahaman, Dunlop and Kent, 2006) use Cultural Algorithms to train driver agents in learning the correct usage of vehicle safety restraints for children. They aim to understand the behaviours of drivers in selecting and applying the correct constraint types for children. According to the authors, understanding such behaviours will not only reduce the number of accidents, but also decrease insurance costs resulting from claims. The belief space is modelled to consist of situational knowledge consisting of the best exemplars. A correct knowledge source is maintained which represents the actual knowledge that needs to be learnt, and is obtained from surveys done by researchers. This source is used along with the knowledge in the belief space to calculate the quality of the exemplars in the belief space. The researchers use an Average Matching Score (AMS) measure to do this. Knowledge in both the belief space and the correct knowledge source is represented as a binary string, and a matching score is calculated by matching the exemplar strings with the correct knowledge source strings. The AMS is then calculated by getting the average of all the matching scores. The knowledge for each individual agent (the driver) encapsulates the drivers knowledge regarding the age, height and weight of a child. Each of these parameters has an associated set of seat location and types.

The social network under which the agents operates consists of a kinship network and a local neighbour network, the former consisting of family connections of the agent and the latter consisting of drivers residing within a specific distance from the agent. The driver agent uses its knowledge in deciding an appropriate seat assignment. An intervention framework is also implemented to increase the drivers knowledge about using safety restraints. This injects knowledge into the driver population, which is then used by the driver agents. The performance of the driver is measured as the ability of it to correctly select a seat restraint type for the child and minimise injury. The experiments showed that learning from an expert source alone without propagating knowledge through the system via the belief space yields the best results for minimising injury, and adding a cultural component makes the entire system more resilient to change. In the future, the researchers plan on adding other knowledge types, such as historical and domain, to the belief space, and also building different intervention strategies which will be able to influence the culture more easily instead of being held back by the resiliency of the system.

Year	Paper Title and Authors	Major Contribution
2002	Using Cultural Algorithms to Evolve Strategies in Agent-based Models, *T. Tassier, D. Ostrowski, M. Everson*, *R. Reynolds*	Optimal pricing strategies are evolved using Cultural Algorithms in a durable goods market, with the population being supported by Genetic Algorithms.
2003	Using Cultural Algorithms in Industry, *N. Rychtyckyj, D. Ostrowski, G. Schleis*, *R.Reynolds*	Applications of Cultural Algorithms in knowledge-base re-engineering and evolution of pricing strategies are discussed.
2004	Using Cultural Algorithms to Evolve Strategies for Recessionary Markets, *D. Ostrowski, R. Reynolds*	A Cultural Algorithm framework is developed to model and evolve pricing strategies during periods of economic recession.
2003	A Cultural Algorithm to Guide Driver Learning in Applying Child Vehicle Safety Restraint, *Z. Kobti, A. Snowdon, S. Rahaman*, *T. Dunlop*, *R. D. Kent*,	A Cultural Algorithm is used to understand the selection and usage of child vehicle safety restraints by drivers.

5. Concluding Remarks

The aim of this survey was to present work done in the training and learning of complex multi-agent systems by using cultural evolutionary techniques. Since their introduction in (Reynolds, 1994), they have been used in place of biological evolutionary methods such as Genetic Algorithms in many areas as described here. Their ability to allow for construction of social agent communities makes them ideal for work in systems where propagation of knowledge is either required, or makes the learning much more efficient.

Work in the multi-agent village simulation of the Mesa Verde region continues, with researchers adding more factors into the belief space. As (Reynolds, Kobti, Kohler and Yap, 2006) comment, "Environmental constraints such as erosion, depletion of firewood, and reduction in plant and animal density due to increased human populations have yet to be added to the current model and will certainly serve to reduce population levels more than observed here. It will be of interest to observe how the agents adapt their networks and cooperation strategies in response to these density- and time-dependent factors." A gaming environment is being created by them to not only to simulate the environment, but also to assist in educational areas, such as making students learn about the history of the area, and see how many different factors influence communities. Industry related research is also increasing, as documented by (Reynolds and Jin, 2002) , (Rychtyckyj, Osrtowski, Schleis and Reynolds, 2003). (Rychtyckyj, Ostrowski, Schleis and Reynolds, 2003) add "Our experience with Cultural Algorithms have been very positive and we look forward to applying this technology in the future here at Ford Motor Company.". Furthermore, recent advances in the field sees increasing use of Cultural Algorithm alongside other algorithmic approaches for machine learning, as presented in (Reynolds and Peng, 2006).

APPENDIX I : ANNOTATIONS

*Papers marked with * are considered milestone papers in the field.*

[1]. **A Cultural Algorithm to Guide Driver Learning in Applying Child Vehicle Safety Restraint**, *Ziad Kobti, Anne W. Snowdon, Shamual Rahaman*, *Tina Dunlop, Robert D. Kent*, 2006, IEEE *

The authors design a multi-agent system using Cultural Algorithms to examine the interactions between the various factors and dimensions due to injuries caused by vehicle crashes. Their goal is to get a better understanding of how drivers decide on the usage and selection of constraints and location in the vehicle for children. They have previously designed an agent based prototype for simulating the effect of safety systems on the injury levels in children passengers.

For implementing their system, they design a belief space that is composed entirely of situational knowledge of the best examples. The knowledge structure for a driver is composed of what the driver knows about assigning an appropriate restraint for the child passenger, taking into account the height, age and weight. The correct knowledge source consists of the actual correct knowledge, to distinguish between the knowledge the driver knows. The overall social network consists of two sub-networks, one for kinship connections and the other for connections between drivers present within a specific distance from the current driver agent. The knowledge propagated impacts the other drivers in this network. An intervention framework is used to improve driver knowledge for selecting restraints. The performance is measure using a Driver Score, which measures the ability of the driver in selecting a correct child restraint.

Testing of the model is done under six different settings, with each setting dependent on the knowledge propagation method and the knowledge sources. The authors observe that individual experience based on expert knowledge gives the best result for reducing injury, and the social network has little impact on improving the correctness level. While measuring the quality of the best examples in the belief space, they observe that the social network causes the examples to exhibit the best performance. They conclude that the absence of cultural influence causes the most improvement in the drivers selection of child restraint, but at the same time it makes it more susceptible to change. In future work, they plan on adding other types of knowledge to the belief space and devising other intervention strategies and seeing their effect on driver learning.

---

[2]. **An Introduction to Cultural Algorithms**, *Robert G. Reynolds*, 1994 IEEE *

In this paper, the author introduces a new model for evolutionary computation based on cultural evolution. This framework views cultural evolution as a dual-inheritance system, supporting evolution at both the individual and population levels. The author bases his dual-inheritance model for cultural evolution on the THINK model developed by (Renfrew, 1994). Based upon the dual-inheritance system, the cultural algorithm supports transmission of behaviours and traits between individuals at the individual level, and the formation of generalised beliefs based on past experience of individuals at the population level. The algorithm works by first initialising the individuals in terms of traits or behaviours and a mappa (generalised description of their experience), and a belief network. Certain individuals are then selected based on an acceptance function which update the belief space, which then influences the selection of individuals in the next generation through an influence function. This process repeats itself till a termination condition is met. The author describes a specific implementation of the cultural algorithm on a sample population represented by a Genetic

Algorithm framework. The generalisations of the performance of the individuals is represented by Version Spaces The author concludes that, computationally, cultural evolution works faster than biological evolution because of the constraints placed on the performance of individuals and the maintenance of a performance history that is separate from the individuals.

Cited by: J. H. Kim, H. Myung, Evolutionary programming techniques for constrained optimisation problems, IEEE Transactions on Evolutionary Computation, 1997

---

[3]. **Cultural Algorithms in Dynamic Environments**, *S. Saleem, Robert G. Reynolds,* 2000, IEEE

The authors observe the tracking of change and prediction of an agents next move in a dynamic environment by observing both a self-adaptive and cultured Evolutionary Programming version of Cultural Algorithms in order to understand the reasoning of environment change in dynamic environments. The authors are inspired by previous work done by (Sternberg, 1997), in which cultural algorithms were used to study the reengineering of rule based Fraud Detection Expert Systems in dynamic environments. They aim to observe the behaviour of Cultural Algorithms in more generic settings. The dynamic landscape consists of multiple cones of different heights.
A dynamic influence function is introduced which introduces diversity in the population. The belief space knowledge is represented by normative and situational knowledge, along with information from previous environments and current environment characteristics. The population is modelled using an Evolutionary Programming model. Mutation by the influence function only occurs for location only as the maximum peak height is assumed to remain

constant. The acceptance function only considers the top 20% of the population. During testing, the performances of the implemented version of the Cultural Algorithm and a population-only version are compared. They authors observe that their version was less sensitive to changes and shift magnitudes than the population-only version, especially in situations where the changes and shifts are of a large nature. They conclude that Cultured systems such as the one implemented are very well suited in systems where system constraints change over time.

Cited by: K. Weicker, Problem Difficulty in Real-Valued Dynamic Problems, Proceedings of the International Conference, 7th Fuzzy Days, 2001

---

[4]. **Altruism, Selfishness, and Survival: An Agent-Based model of Sharing Behaviour**, *Steven J. Goodhall, Robert Whallon, Robert G. Reynolds*, 2002, IEEE

The authors investigate how the vector voting model exploits resources such as food in the environment in order to understand the influence of culture on the decision making behaviour of agents. They use distribution of food within the foraging group to test the outcomes. In their previous work, they used several different strategies for allocating resources to investigate the vector voting model and observed that it performed best when certain strategies such as Fixed Rank Order were used as compared to others such as Neediest First. In this paper, they use a new strategy which involves periodic reordering of precedence by simulating groups of foraging agents gathering and hunting in a physical environment.
Each band of hunter-gatherer agents are modelled to move within a specific space at the start of each day by selecting a direction. Directions are selected based on memories of past consumption of resources. This model is inspired

by the manner in which primate groups make decisions, as researched by (Kummer, 1968). The agents forage in that direction till their needs are met or till they have covered the maximum distance for the day. The consumption strategies are characterised as being either selfish or altruistic. The strategies are Fixed Order, Equal Shares, Round Robin, Neediest First and Satisfied First. Movements made in the foraging space are based on memories which are recorded at the end of each day. Fully nourished agents can reproduce. Food resources are distributed according to a patchwork distribution to emulate a semi-arid environment, and is available around landmarks marked in the landscape.

The model is implemented using Swarm and Objective C. The experiments conducted are used to compare specific alternatives for foraging and survival. The authors observe that decisions made based on memories of past successes performed better than random ones, and that selfish resource allocation strategies such as Fixed Order outperformed altruistic ones such as Equal Shares. For future work the authors plan on allowing individuals to pool their memories, thereby allowing decisions to be made that are more reliable than the ones made using simple vector voting.

---

[5]. **Using Cultural Algorithms to Evolve Strategies in Agent-based Models**, *Troy Tassier, David Ostrowski, Mark Everson, Robert G. Reynolds*, 2002, IEEE

The authors apply Cultural Algorithms to derive generalised belief sets for the belief space from successive populations of agents in a simulation of a durable goods market, using both white and black box testing techniques of software engineering within the framework. (Reynolds, Zannoni, 1996) had in a previous work used Cultural Algorithms to extract design knowledge from Genetic Programmes. In this paper they use Genetic Programming to support the population, using beliefs to represent programme segments. These beliefs are used to guide search for solutions for either similar or same problems and to reengineer existing solutions.

The authors implement a Dual Cultural Algorithm with Genetic Programming (DCAGP) in order to use both black and white box testing for generating new designs. They use a Java based multi-agent system called MarketScape to model economic scenarios, which uses four distinct objects, namely consumers, vehicles, manufacturers and dealers of used vehicles. Consumers make evaluations for new or used vehicles every period. They use a memory based strategy called postpone strategy which is related to consumer expectations regarding the fall in vehicle prices.

For testing, they use 200 agents in a 60-period timeframe in the MarketScape environment, using them to observe improvements in the postpone strategy to determine pricing of goods. For the white box phase, they use normative knowledge to acquire interval information for the coefficients in the MarketScape equations. They observe that a Cultural Algorithm employing white box testing generates in only half the number of runs the amount of profitability that the GP strategy generated in 100 runs. The DCAGP generates the highest amount of profitability, and in quicker time than using only traditional evolutionary techniques. They conclude that by using complementary software engineering techniques, near optimal strategies can be derived in multi-agent systems.

Cited by: P. Arpaia, G. Lucariello, A. Zanesco, Multi-Agent Remote Predictive Diagnosis of Dangerous Good Transports, Instrumentation and Measurement Technology Conference, IEEE, 2005

[6]. **Simulating the Evolution of Archaic States**, *Robert G. Reynolds*, *Alina Lazar*, 2002, IEEE

The authors of the paper attempt to make a multi-agent model using Cultural Algorithms of the ancient civilisation at the Oxaca Valley in Mexico in order to test the "evolution without stages" hypothesis of (Marcus, Flannery, 1996). The theory proposed by Marcus and Flannery suggested that archaic states evolved based upon changes in the social and political institutions, and not on the occupational stages that the states underwent with time. The authors aim to operationalise this model using Cultural Algorithms. They attempt to observe to what extent the natural and social environment of the valley affects the aggregation processes.
In order to model the environment, the authors make a grid of 1 by 1 km2 cells, some of which are occupied by an agent, which in turn represents the group residing at the cell. Occupied cells are represented by a set of environment variables, and those which are not occupied are represented by the class of agricultural lands contained in them. The social relationships are modelled using a linear model which determines the radius within which agents can maintain social relationships. The behaviours of the agents are considered to be static within each period, but can vary from period to period. Seven different categories are used to constrain the activities an agent can perform within a cell. Agent interaction with the environment is modelled using a production plan and a relational plan, each of which are represented as binary vectors. Relationships are established using a payoff matrix, which is equivalent to that of the prisoners dilemma if there are equal populations on both sides requesting a relationship, and tends to the Wardens dilemma when the payoff to the agents in one cell is greater than that to the agents in all the other cells. A Cultural algorithm is used to collect and process the information regarding social interactions and probabilistically guides the production and relation plans. In future work, the authors plan on investigating the extent to which the transformation from the prisoners

dilemma to the Wardens dilemma affects the rate of increase in the social complexity of the system.

Cited by: R. G Reynolds, B. Peng, J. J Brewster, Cultural swarms: knowledge-driven problem solving in social systems, Systems, Man and Cybernetics, IEEE, 2003

---

[7]. **A Multi-Agent Simulation using Cultural Algorithms: The Effect of Culture on the Resilience of Social Systems**, *Ziad Kobti*, *Robert G. Reynolds*, *Tim Kohler*, 2003, IEEE *

In this paper the authors develop a framework based on cultural algorithms to extend a multi-agent model developed by (Kohler, 2000) for the Mesa Verde Prehispanic Pueblo region. They aim to embed various social parameters into this framework and study the social networks resulting from various combinations of these social parameters. The model developed by Kohler aimed to understand the reasons as to why the people of the aforementioned region disappeared from there.

In the model developed here, each agent is represented by a household in the study area of the Mesa Verde region, each of which consists of individuals. The agents have a local knowledge set, whose parameters are influenced by the knowledge held by the population (beliefs), and vice versa. The agents gain knowledge through various social interactions. This knowledge in turn affects the belief space, which then guides future agent actions. The social network formed maintains kinship ties between the agents, which were absent in the original model by Kohler. The experiments were conducted in the Swarm simulation environment. Various social parameters are adjusted to examine the resulting social networks, notably the move and search radius for the agents. The authors conclude that the larger the move radius, the lesser the population size, and the resulting social network is sparsely populated, and vice versa with a smaller move radius.

Cited by: R. G. Reynolds, Z. Kobti, Learning in Dynamic Multi-layered Social Networks: A Mesa Verde Example, Proceedings of Geo-Computation 2005, Ann Arbor Michigan, August 1-3, 2005

---

[8]. **Cultural Swarms: Knowledge-driven Problem Solving in Social Systems**, *Bin Peng, Jon J. Brewster*, *Robert G. Reynolds*, 2003, IEEE

The authors test the validity of the Cultural Swarm hypothesis in order to explain the emergence of high-level patterns of interaction between knowledge sources in the belief space. In a previous work by (Saleem, Reynolds, 2003), they had identified five different knowledge sources, namely normative, historic, historic, situational and topographic, and three phases of problem solving in the belief space. In order to test the existence of Cultural Swarms, they use the Canonical Particle Swarm Model given by (Eberhart, Kennedy, 1995). They aim to examine how the knowledge in the belief space guides and creates groups of problem solving agents in the population.

In the belief space, at each time step a type of knowledge source is selected using roulette wheel selection. The probability of selection depends on the previous successes. The function landscape consists of cones of varying height, and the aim is to find the cone with the highest height. Ten different configurations of the landscape are used. The experiment runs for 100 generations or till the maximum value is found. During testing, the authors observe swarming activity in the belief space. During the runs, topographic knowledge leads the search for the optimum solution, followed by other knowledge sources. The authors conclude that by observing such swarming behaviour, meta-level rules can be formed for Cultural Evolution as an extension to the Schema Theorem given by Holland.

---

[9]. **Cultural Swarms**, *Bin Peng*, *Robert G. Reynolds*, *Jon Brewster*, 2003, IEEE *

In this paper the authors investigate whether the existence of swarming behaviour at the meta-level (belief space) causes the same to occur at the population level in an Evolutionary Programming model. Work done previously by (Reynolds, Saleem, 2003) had indicated the existence of Cultural Swarms in the belief space of the population while using a Particle Swarm model. Previous work done by the same had also indicated the existence of five types of knowledge, namely normative, situational, historic, topographic and domain, and also three phases of problem solving, namely coarse grained, fine grained and backtracking.
In this paper, the authors use an Evolutionary Programming model instead of a Particle Swarm model to represent the population. The problem being used is that of finding the global maximum value. Test problems are generated using a multi-modal problem generator developed by (Morrison, De Jong, 1999). The resulting landscape consists of cones of different heights, spread out at different locations in a 2-dimensional function space. The initial population is set to 100 agents, and terminates after 300 generations. During the course of experimentation, the authors observe that in landscapes having less than 25 cones, the Cultural Algorithm performs well and finds the global maximum value within the first few generations, whereas with landscapes having more than 25 cones, the time taken is significantly longer. The authors conclude that different knowledge sources are able to function as a

knowledge repository and improve the learning in multi-modal environments, thereby inducing a flocking behaviour (swarms) in the individuals at the population level and enabling them to converge to a solution quickly. They also remark that this enables identical agents to play different roles during the problem solving process.

Cited by: B. Kaewkamnerdpong, P.J. Bentley, Perceptive particle swarm optimisation: an investigation, Swarm Intelligence Symposium, 2005. SIS 2005. Proceedings 2005 IEEE

---

[10]. **Cultural Swarms: Assessing the impact of Culture on Social Interaction and Problem Solving**, *Radu Iacoban*, *Robert G. Reynolds*, *Jon Brewster*, 2003, IEEE

In this paper the authors test the swarm population model they developed in the preceding paper in order to investigate the role different types of knowledge play during the course of problem solving by a population of agents using Cultural Algorithms. They produce a landscape of a set of randomly distributed conic structures for a two-dimensional real valued function, using the number of cones, their location in the landscape and their height as variable experimental parameters. They run the Cultural Algorithm using the swarm population model to find the optimum over 10 different random configurations of cones.
In a previous work done by (Reynolds, Saleem, 2003), the authors had identified three problem solving phases, each using certain types of knowledge. In this paper, the authors observe these phases replicated in the course of their experiments. They observe that topographic knowledge finds the most promising region in the landscape, and then situation knowledge

fine tunes that region to narrow down on a potential solution. They observe this cycle of finding a region and then narrowing down repeat itself, thus concluding that topographic and situation knowledge are the most important types during the early stages. They authors conclude that during the course of their experiments, they observed three problem solving phases, namely coarse grained, fine tuning and backtracking, and see that each phase is dominated by a certain knowledge type. In future work, the authors plan to develop a GUI to assist in visualising meta-level patterns generated during problem solving, and investigate the roles of historic and situation knowledge in more detail.

Cited by: R.L. Becerra, C.A.C. Coello, Culturising differential evolution for constrained optimization, Proceedings of the Fifth Mexican International Conference in Computer Science, 2004

---

[11]. **Cultural swarms II: virtual algorithm emergence**, *Bin Peng*, *Robert G. Reynolds*, *Jon Brewster*, 2003, IEEE

The authors investigate how algorithmic interaction between knowledge sources at the meta-level occur in Cultural Algorithms. In previous papers, (Reynolds, Saleem, 2003), they had demonstrated the existence of swarming behaviour in the agents while various knowledge sources interacted. They observe the meta-level patterns that are produced as a result of this swarming behaviour. They use real world maximisation problems as a test bed for their experiments. The specific problem they use is one in which a random field of cones of varying heights are generated, which was also used in their previous work.

During testing, they observe the influence each knowledge source has on the population. For normative knowledge, they observe that it produces a state space search graph with respect to the changes in the patterns of influence during the search. Situation knowledge follows a more exploitative role, as it produces narrow and elongated directions to the population. Domain knowledge moves outwards and then slowly starts converging to the optimal solution. Historic knowledge summarises the intersection of the search taken by the other sources. Topographic knowledge generates the search space, and as a result influences the other knowledge sources. They observe that a virtual best first algorithm gets induced from the flow of the population as the various knowledge sources interact with each other. They conclude that the influence of the five knowledge sources causes a best first search to take place in the state space by creating virtual graphs that direct the flow of the population individuals.

Cited by: M. G. H Omran, Particle Swarm Optimization Methods for Pattern Recognition and Image Processing, 2004

---

[12]. **Cultural Swarms: Modelling the impact of Culture on Social Interaction and Problem Solving**, *Radu Iacoban*, *Robert G. Reynolds*, *Jon Brewster*, 2003, IEEE

In this paper the authors test the existence of a Cultural Swarm during different phases of solving solve real valued function optimisation problems, and observe how different knowledge sources guide the individuals of the swarm. A previous work by (Reynolds, Saleem, 2003) had identified five types of knowledge while using Cultural Algorithms to solve optimisation problems with the population represented by an Evolutionary Programming model, namely normative, topographic, historic, situational and historic knowledge. The same work had identified three phases of problem solving, namely coarse grained, fine grained and backtracking phases. The authors replace the Evolutionary Programming model with the Canonical Particle Swarm Model given by (Eberhart, Kennedy, 1995).

The authors describe the representation and updating details for each of the five knowledge types. Situation knowledge is represented by a set of exemplars from the population. Domain knowledge is represented by a set of domain ranges for the various parameters and the best examples from the population. Normative knowledge is represented by a set of intervals thought to be a good solution for each parameter. Historic knowledge is represented by sequences of changes in the population environment, in this case by shifts in the distances and direction of the optimum in the function landscape. The topographic knowledge is represented by a multi-dimensional grid, which is determined by the number of dimensions of the function. Five different influence functions are used, one for each knowledge type, and a roulette wheel mechanism is used to select them.

The authors observe that the particles in the swarm are able to gain knowledge which they use to modify their culture and social interactions. They also observe that the particles use the best solution they have and the best solution each of the five knowledge types have, and collectively use that to guide themselves to the optimum solution. In a companion paper, the authors investigate how the landscape features and the knowledge types are related to each other.

Cited by: R.L. Becerra, C.A.C. Coello, Culturising differential evolution for constrained optimization, Proceedings of the Fifth Mexican International Conference in Computer Science, 2004

---

[13]. **Using Cultural Algorithms in Industry**, *Nestor Rychtyckyj, David Ostrowski, George Schleis*, *Robert G. Reynolds*, 2003, IEEE *

In this paper, the authors explore some of the areas in industry where the concepts of Cultural Algorithms can be applied. They discuss the usage of Cultural Algorithms for knowledge base re-engineering. Cultural Algorithms are used using a top-down and bottom-up approach for solving the semantic network re-engineering problem in order to improve the efficiency of the subsumption algorithm. They test this approach on the Direct Labour Management System (DLMS) used by Ford Vehicle Operations. First they re-engineer its semantic network using Cultural Algorithms using a top-down approach. They also use a bottom-up approach to re-engineer it. The approaches are similar as far as the manner of solving go, but differ in the manner of problem representation. In another application, they use Cultural Algorithms for modelling pricing strategies for vehicle sales as a multi-agent modelling system. They use the Original Equipment Manufacturer (OEM) pricing model to the multi-agent system. Their framework allows for an easy demonstration of dynamic events in the market which would otherwise be hard to observe using standard economic theory. The authors plan on using Cultural Algorithms in other areas including pricing strategies during recessionary periods, text understanding, classification and analysis, data mining and knowledge discovery and evolution of individual strategies in agents.

Cited by: R. G Reynolds, B. Peng, Cultural algorithms: modeling of how cultures learn to solve problems, Tools with Artificial Intelligence, ICTAI 2004

[14]. **Using Cultural Algorithms to Evolve Strategies for Recessionary Markets**, *David Ostrowski, Robert G. Reynolds*, 2004, IEEE

The authors develop a Cultural Algorithm framework to abstract coefficients of pricing strategies and simulate consumer behaviour during economic recession in order to observe the manners in which using evolutionary programming can work with software engineering techniques. They use both white and black box testing approaches used in software engineering.

The Cultural Algorithm they develop consists of two algorithms chained together, one for black and one for white box strategy, each using a belief space of its own to represent implicit and explicit constraints for developing programmes. This framework was proposed by the authors in a previous paper, and is called the Chained Cultural Algorithm (CCAGP). Ostrowski had developed MarketScape, a multi-agent system to simulate real-world consumer markets which uses Original Equipment Manufacturer (OEM) pricing model. The CCAGP framework is used to evolve a near optimal pricing strategy for maximising the profits in an OEM.

For testing, recession is modelled in MarketScape and applied to a 60 period scenario. The CCAGP framework is then applied to a population of 100 agents for a total of 300 generations. A second, longer recessionary period is tested as well. The authors observe that in the shorter period, the CCAGP derive a lower pricing strategy for making profits, whereas for the longer period, it derives a higher pricing strategy. They conclude that using both black and white box testing using Cultural Algorithms efficiently derives knowledge in complex multi-agent systems, such as those modelled in the paper.

Cited by: Robert G. Reynolds, Bin Peng, Cultural Algorithms: Computational modeling of how cultures learn to solve problems: An Engineering Example, Cybernetics and Systems, Vol. 36, Issue 8, 2005, pp 753-771

[15]. **Cultural Algorithms: Knowledge Learning in Dynamic Environments**, *Robert G. Reynolds*, *Bin Peng*, 2004 IEEE

In this paper the authors investigate the behaviour of cultural algorithms in a dynamic environment. Previous work (Reynolds, Saleem, 2003) demonstrated the emergence of three phases of knowledge integration while solving a problem. The authors attempt to study the effect of these in a dynamic environment. The dynamic environment is inspired by work done by (Lahann, 2003) on smart surfaces, where the surface changed from water-repelling to water-attracting by applying a weak electrical field.

The cultural algorithm uses five knowledge sources: Normative, Situational, Domain, Historic and Topographic. Any cultural knowledge can be expressed as a combination of the above types. The population is expressed as an Evolutionary Programming model, having parent individuals which reproduce to make children, which further mutate and get evaluated. The dynamic landscape is formed by a series of overlapping conic structures. The objective of the cultural algorithm is to find the global optimum in the dynamic landscape. The landscape, though being dynamic, repeats its pattern. During the training, the optimum was moved in a counter clockwise direction through each cone. In the testing done by the authors, during the course of the population learning, certain knowledge sources become more dominant than the others, and the search goes from being a coarse-grained search to a more specific and fine search. In other words, the cultural algorithm manages to adjust its knowledge sources and interactions and produce swarms that are able to predict the changes in the landscape parameters.

[16]. **The Effect of Kinship Cooperation Learning Strategy and Culture on the Resilience of Social Systems in the Village Multi-Agent Simulation**, *Ziad Kobti*, *Robert G. Reynolds*, *Tim Kohler*, 2005, IEEE *

In this paper, the authors explicitly embed kinship strategies in a previously develop cultural algorithm framework in order to see the effect of this on the agents in the Village Multi-Agent simulation. In their previous work (Kobti, Reynolds, 2003), the authors had included reciprocal exchange strategies in a manner such that they were carried out randomly, thereby resulting in the loss of information regarding who carried out the exchanges. Here, they attempt to explicitly add the reciprocal exchange, thus allowing the agents to remember with whom they carried out the exchanges, and also giving them the ability to learn over time with which agent an exchange is most likely to succeed.

In their previous work, the authors used random and roulette wheel selection to determine which agents would interact and perform exchanges. A further constraint on this was that the agents had to be directly linked, and within a specific distance from each other. Whether or not the actual exchange takes place was dependent on the agent's state, and its probability of cooperation. In this paper, they allow the agents to plan out their exchanges and determine which agents they prefer to interact with, and also produce generalisations in the belief space regarding the type of individuals they are most likely to perform exchanges with. To facilitate this, all agents are initialised with an equal probability for interaction with its kin, and over time they learn to bias their selection towards specific agents. Learning takes place at the individual level for interacting with specific agents, and at the cultural level (the belief space) for generalising about the type of agents to interact with. The selection strategy is represented as a vector of probabilities, each of which is selected using roulette wheel selection. Over time, successful interactions with an agent increase the probability for that interaction, and unsuccessful ones decrease it. Knowledge at the belief space is expressed using Normative knowledge, and is used to influence the local strategy for identifying kin agents most likely to give a positive response to requests for exchange.

The authors test the architecture by first allowing only random selection of agents for exchange, then allowing learning at the individual level for modification of the probabilities, and finally allowing the global knowledge to be changed to guide agent selection. All of these strategies are tested with variable move radii, where a move radius determines the extent to which the agents can move about and relocate to search for resources. Using cooperation at the individual level allows for a larger and more complex network to be sustained. Adding belief space knowledge also increases the network volume, but decreases the maximum size of the hub nodes, which are nodes with the highest number of links. Upon increasing the search radius, they notice a decline in the number of successful requests for exchange, as the population is less dense. The authors conclude that adding cooperation increases the network size and its resilience to drought conditions, and also allows it to recover better when these conditions subside. The downside is that there is more dependence on the hub nodes to hold the network together. They observe that if individuals were more spread out then that would reduce stress on the system. In future work, they will examine the effects of resettlement of hub nodes within the landscape of the network.

Cited by: R. G. Reynolds, Z. Kobti, T. A. Kohler, L. Y. LYap, Unraveling ancient mysteries: reimagining the past using evolutionary computation in a complex gaming environment, IEEE Transactions on Evolutionary Computation, Dec. 2005

[17]. **Modelling Protein Exchange across the Social Network in the village Multi-Agent Simulation**, *Ziad Kobti*, *Robert G. Reynolds*, 2005, IEEE *

In this paper the authors attempt to introduce protein resources in the form of deer, rabbits and hares in order to study their effects in the population of the village agent simulation. The hunting plans for the agents are based on Charnov's Marginal Value Theorem (Charnov, 1976). This work extends the previous architecture developed by (Kobti, Reynolds, Kohler, 2003). The social network constructed in this paper consists of kinship, economic and hub networks. Kinship network is based on the interactions between agents in a household; the economic network models the economic exchanges between agents; the hub network is composed of successful agents from both these networks. Two types of exchange, generalised reciprocal exchange and balanced reciprocal exchange are used, the former based on one-way exchanges and the latter based on two-way exchanges. The networks using these exchange systems are the generalised reciprocal network (GRN) and the balanced reciprocal network (BRN). The hunting plans for agents are processed using an approach based on Charnov's Marginal Value Theorem.
During testing, the initial agent population is set to 154 agents. The system is a closed system, with no external factors influencing the population. The authors note that the system is able to survive several drought periods, with drops in the population in each period. They notice that the BRN is first to feel the drought, followed by the GRN and then the Hub, which is the hardest hit. The authors conclude that the population is susceptible to the fluctuations in the animal resources due to environmental constraints. They deduce that such pressures may cause the population to leave the region due to the inability to sustain their protein requirements. The authors plan on extending this work to include other paleoproductivity measures.

[18]. **Unraveling Ancient Mysteries: Reimagining the Past Using Evolutionary Computation in a Complex Gaming Environment**, *Robert G. Reynolds, Ziad Kobti, Timothy A. Kohler, and Lorene Y. L. Yap*, 2005, IEEE *

In this paper the authors develop a framework using Cultural Algorithms to investigate the disappearance of pre-Hispanic Indians from their homelands. Their aim is to examine the extent to which cooperation and competition in agent households affect population structure and spatial distribution, with respect to the actual archaeological discoveries. In previous work done by (Kobti, Reynolds, 2003) and (Kobti, Reynolds, Kohler, 2003), they developed simulations which introduce kinship and reciprocal exchanges between agents in the population. They use a "Trust in Network" game paradigm in their current simulation, with includes three types of strategies, namely trust, inspect and defect.

Each agent in their framework represents a household, each of which is composed over several individuals. Two social networks are used to model the social relationships of the household, a generalised reciprocal exchange network (GRN), and a balanced reciprocal exchange network (BRN). Learning amongst agents takes place at two levels, one at the individual level which depends on their past experiences, and the other at the cultural level which guides individuals to make plans for gathering resources. During testing, they first use just a single "trust" strategy where all agents cooperate with each other. Then they add GRN to the simulation, followed by both GRN and BRN. The latter two are further tested by adding defection. They observe that there is no cooperation between agents, the population starts small, and remains small throughout the testing. Adding GRN causes a slight increase in the population size. When both GRN and BRN are used, the population size increases to a much larger value, about 10 times as much as with GRN alone.

Adding defection causes the population to stabilise to values supported by archaeological records. The authors conclude that in the real world both GRN and BRN would have to have existed, with some individuals choosing to defect within them.

---

[19]. **Agent Based Modelling of Early Cultural Evolution**, *Robert G. Reynolds, Robert Wahllon, Mostafa Z. Ali, Behnooshi M. Zadegan*, 2006, IEEE

In this paper, the authors use a previously developed multi-agent model of hunter-gatherer search methods and perform experiments on it in order to examine the impact of decision-making and resource-sharing methods on the survival of the agent population. The model developed previously by (Reynolds, Kobti, Kohler, 2005) was used in the simulation of the village multi-agent simulation, which tested the effect different sharing strategies had on the survival of the agent population. In this paper, the authors increase the potential for agent interaction and add more knowledge and test the impact of this on the decision making strategies of the population, and its survivability. In the model proposed here, agents cast votes to choose a direction to move or to gather resources. Once votes are cast, an algorithm is invoked to tabulate the votes cast and get the direction to move in and its magnitude. The agents then move about and forage for food, which is either the maximum they can collect in a day or is dependent on the maximum cells they can visit in a day. After the gathering, the individuals of the group assemble at a new location. The ability of the agent group to decide on a direction for foraging depends on their ability to gather information, and is independent of the group size.

In the experimental setup, the landscape is represented as a 128 by 128 grid of cells, with each region of 12 by 12 cells representing a food patch. Each patch has a food value between 0 and a maximum value. The amount of food an agent requires is user-defined, set as 1500 in the experiment. Six strategies are used for deciding on the sharing of food resources between the individuals of a group, namely Fixed Order, Equal Portions, Neediest First, Satisfied by Decision, Round Robin and Reverse Fixed Order. Three types of simulations are used, namely Homogenous simulations that use only one strategy, Homogenous simulation/Heterogeneous band that ensures individuals in the same band share the same strategy and Heterogeneous simulation/Heterogeneous band in which individuals in the same band can use a different strategy. The authors observe that in the homogeneous experiments, as the richness of the environment increases, the population size and the number of bands increases as well. In the heterogeneous experiments, the Fixed Order strategy dominates and causes the populations and number of bands to increase, while the others cause them to decline. The authors conclude that Fixed Order sharing strategies are able to efficiently cover larger search radii than egalitarian strategies such as Equal Sharing. The authors also observe that using sharing strategies that indirectly preserve information leads to increased survivability of the individuals.

---

[20]. **Cultural Evolution of Ensemble Learning for Problem Solving**, *Bin Peng*, *Robert G. Reynolds*, *R. S. Alomari*, 2006, IEEE

The authors introduce a Cultural Algorithm that is integrated with ensemble learning, which was introduced by (Diettreich, 2002). They use the five knowledge types of the belief space as an ensemble, with each interacting with the other to control exploration of the environment. They are motivated by the fact that successful ensemble learning depends on the diversity and accuracy of its members, with each knowledge source is viewed as an ensemble serving as a decision marker. They use a problem which involves minimising the weight of a tension/compression spring, which is subject to constraints on minimum deflection, shear stress, surge frequency, limits on outside diameter and design variables. During experimentation, the population size is set to 200, and a maximum of 5000 generations are allowed. Five subcultures are evolving within the population, each corresponding to an ensemble. The future generations are guided by decision markers evolved by sub-cultures in the previous generations. The test results are plotted to observe trends in the mean coil diameter, wire diameter and the number of active coils. They observe that the convergence to the optimal value occurs very quickly because of the guidance by the ensembles. The authors conclude that ensemble learning fits naturally into the Cultural Algorithm framework as the knowledge sources can be viewed as ensembles.

APPENDIX II : Bibliography

Al-Shehri, H. (1997), Evolution-based Decision Tree Optimisation Using Cultural Algorithms, Ph.D. Dissertation, Wayne State University

Angeline, P. A (1995), Adaptive and Self-Adaptive Evolutionary Computation, In Communication Intelligent, pp152-163

Charnov, E. L. (1976) Optimal Foraging Theory: The Marginal Value Theorem, Journal of Theoretical population Biology, Vol. 9, pp129-136

Chung, C., Knowledge based approaches to self-adaptation in cultural algorithms, Ph.D. dissertation, Wayne State University

Dasgupta, D., Michalewicz, Z. (1997), Evolutionary Algorithms in Engineering Applications, Springer-Verlag

Diettrich, T. (2002), Ensemble Learning: The Handbook of Brain Theory and Neural Networks, 2nd Edition, pp 405-408

Fogel, D. B. (1993), Evolving behaviours in the iterated prisoners dilemma, Evol. Comput., Vol. 1., No. 1, pp77-97

Holland, J. H. (1975), Adaptation in Natural and Artificial Systems, Ann Arbor, University of Michigan Press

Kennedy, J., Eberhart, R. C. (1995), Particle Swarm Optimisation, The IEEE International Conference on Neural Networks, pp12-13

Kobti, Z, Snowdon, A. W., Rahaman, S., Dunlop, T., Kent, R. D. (2006), A Cultural Algorithm to Guide Driver Learning in Applying Child Vehicle Safety Restraint, 2006 IEEE Congress on Evolutionary Computation, pp 1111-1118.

Kohler, T. (2000), The Final 400 years of pre-Hispanic Agricultural Society in the Mesa Verde Region, Kiva, V. 22, pp191-264

Kummer, H (1968) Primate Societies, New York, Aldine Press

Lahann, J. (2003), A Reversibly Switching Surface, Science Vol. 299, pp371-374

Marcus, J., Flannery, K. V. (1996), Zapotec Civilisation: How Urban Society Evolved in Mexico's Oxaca Valley, Thames and Hudson Press Ltd. London

Morrison, R., De Jong, K. (1999), A Test Problem Generator for Non-Stationary Environments, Proceedings of Congress on Evolutionary Computation, pp2047-2053

Renfrew, A. C. (1994) "Dynamic Modelling in Archaeology: What, When and Where?" Dynamic Modelling and the Study of Change in Archaeology, S. E. van der Leeuw, ed., Edinburgh University Press

Reynolds, R. G. (1979), An adaptive computer model of the evolution of agriculture for hunter-gatherers in the Valley of Oaxaca, Ph.D. dissertation, University of Michigan, Ann Arbor

Reynolds, R. G. (1999), An overview of Cultural Algorithms, Advances in Evolutionary Computation, D. Corne, M. Dorigo and F. Glover, Eds. New Yourk: McGraw-Hill, pp367-378

Reynolds, R. G. (1994), An Introduction to Cultural Algorithms, Proceedings of the Third Annual Conference on Evolutionary Programming San Diego, California, pp. 131-139

Reynolds, R. G., Saleem, S. (2000), Cultural Algorithms in Dynamic Environments, Proceedings of the 2000 Congress on Evolutionary Computation, pp1513-1520

Reynolds, R.G., Goodhall, S. J., Whallon, R. (2002), Altruism, selfishness, and survival: an agent-based model of sharing behaviour, Proceedings of the 2002 Congress on Evolutionary Computation, pp 867 – 871.

Reynolds, R. G., Jin, X. (2002), Data Mining Using Cultural Algorithms and Regional Schemata, Proceedings of the 14th International Conference on Tools with Artificial Intelligence, pp33-40.

Reynolds, R.G., Ostrowski, D.A., Tassier, T., Everson, M. (2002), Using cultural algorithms to evolve strategies in agent-based models, Proceedings of the Congress on Evolutionary Computation, 2002 Volume 1, pp741 – 746

Reynolds, R. G., Lazar, A. (2002), Simulating the evolution of archaic states, Proceedings of the 2002 Congress on Evolutionary Computation, pp861-866

Reynolds, R. G., Saleem, S. (2003), The Impact of Environmental Dynamics on Cultural Emergence, Oxford University Press, pp1-10

Reynolds, R. G., Kobti, Z., Kohler, T. (2003) Robustness in Coupled Human/Natural Systems in the Northern Prehispanic Southwest, Santa Fe, NM, Oxford Studies in the Sciences of Complexity

Reynolds, R. G., Kobti, Z., Kohler, T. A. (2003), A Multi-Agent Simulation Using Cultural Algorithms, Proc. of Congress on Evolutionary Computation, pp1988-1995

Reynolds, R., Kobti, Z. (2003), A Multi-Agent Simulation Using Cultural Algorithms: The Effect of Culture on the Resilience of Social Systems, Santa Fe Institute Workshop on Human Environmental Interaction, pp1988 – 1995.

Reynolds, R. G., Kobti, Z. (2003), The effect of environment variability on the resilience of social networks: An example using the Mesa Verde Pueblo Culture, presented at the Society for American Archaeology 68th Annual Meeting, 2003

Reynolds, R. G., Kobti, Z., Kohler, T. (2003), Robustness in Coupled Human/Natural Systems in the Northern Prehispanic Southwest, Santa Fe, NM, Oxford Studies in the Sciences of Complexity, Santa Fe Inst. 2003

Reynolds, R. G., Peng, B., Brewster, J. J. (2003), Cultural swarms: knowledge-driven problem solving in social systems, IEEE International Conference on Systems, Man and Cybernetics, Volume 4, pp3589 - 3594

Reynolds, R.G., Peng, B., Brewster, J. (2003), Cultural Swarms, The Congress on Evolutionary Computation, Volume 3, pp1965 – 1971

Reynolds, R.G., Jacoban, R., Brewster, J. (2003), Cultural swarms: assessing the impact of culture on social interaction and problem solving, Proceedings of the 2003 IEEE Swarm Intelligence Symposium, pp212-219

Reynolds, R.G., Peng, B., Brewster, J. (2003), Cultural swarms II: virtual algorithm emergence, The 2003 Congress on Evolutionary Computation, pp1972-1979

Reynolds, R.G., Iacoban, R., Brewster, J. (2003), Cultural swarms: modeling the impact of culture on social interaction and problem solving, Proceedings of the 2003 IEEE Swarm Intelligence Symposium, pp205 – 211

Reynolds, R. G., Rychtyckyj, N. Ostrowski, D. Schleis, G. (2003), Using Cultural Algorithms in Industry, Proceedings of the Swarm Intelligence Symposium, IEEE, pp187 – 192

Reynolds, R. G., Kobti, Z., Kohler, T. (2004), Agent-Based Modeling of Cultural Change in Swarm Using Cultural Algorithms, Proceedings of SWARMFEST 2004, University of Michigan, Ann Arbor

Reynolds, R. G., Kobti, Z., Kohler, T. (2004) The effects of generalised reciprocal exchange on the resilience of social networks: An example from the preshispanic Mesa Verde region, J.Comput.Math.Org.Theory, pp227-254

Reynolds, R.G., Ostrowski, D.A. (2004), Using Cultural Algorithms to Evolve Strategies for Recessionary Markets, Congress on Evolutionary Computation, Volume 2, pp1780 – 1785

Reynolds, R. G., Peng, B. (2004), Cultural Algorithms: Knowledge Learning in Dynamic Environments, Congress on Evolutionary Computation, Volume 2, pp1751 – 1758

Reynolds, R. G., Kobti, Z., Kohler, T. (2004), The Effect of Kinship Cooperation Learning Strategy and Culture on the Resilience of Social Systems in the Village Multi-Agent Simulation, Congress on Evolutionary Computation, Volume 2, pp1743 – 1750

Reynolds, R. G., Kobti, Z. (2005), Modeling Protein Exchange across the Social Network in the Village Multi-Agent Simulation, IEEE International Conference on Systems, Man and Cybernetics, Volume 4, pp3197 – 3203

Reynolds, R. G., Kobti, Z., Kohler, T. A., Yap, L. Y. L. (2005), Unraveling Ancient Mysteries: Reimagining the Past Using Evolutionary Computation in a Complex Gaming Environment, IEEE Transactions on Evolutionary Computation, Volume 9, pp707 – 720

Reynolds, R. G., Whallon, R., Mostafa Z. A., Zadegan, B. M. (2006), Agent-Based Modeling of Early Cultural Evolution, IEEE Congress on Evolutionary Computation, pp1135- 1142

Reynolds, R.G., Peng, B., Alomari, R.S. (2006), Cultural Evolution of Ensemble Learning for Problem Solving, IEEE Congress on Evolutionary Computation, pp1119-1126

Rychtyckyj, N., Reynolds, R. G. (2001), Using Cultural Algorithms to Improve Knowledge Base Maintainability, Proceedings of the 2001 Genetic and Evolutionary Computation Conference, pp1405-1412

Rychtyckyj, N., Reynolds, R. G. (2002), Knowledge Base Maintenance Using Cultural Algorithms: Applications to the DLMS Manufacturing Process Planning System at Ford Motor Company, Proceedings of the 2002 Congress on Evolutionary Computation, pp855-860

Stephens, D. W, Krebbs, J. R. (1986), Foraging Theory, Princeton University Press, New Jersey

Sternberg, M. (1997), Using Cultural Algorithms to Support Reengineering of Rule Based Expert Systems in Dynamic Performance Environments: A Case Study in Fraud Detection, IEEE Transactions on Evolutionary Computation, 1997, pp225 – 243.

Zannoni, E. (1996), Cultural Algorithms with Genetic Programming: Learning to Control the Programme Evolution Process, PhD. Thesis, Computer Science, Wayne State University, Detroit.