Technical Report:	*CSM-395*

PhD in Open Constraint Satisfaction

Technical Report 4:

Implementing Scenario One Strategies And The Use Of Evolutionary Computation

By

Timothy Gosling

Contents

1.1 Introduction

At present various electronic market places, auctions and negotiation systems exist, in the near future full electronic supply chains will be possible and indeed desirable to improve efficiency [4].
This situation, however, presents a problem. While humans are good at negotiation and situation analysis there are less able to handle large volumes of information and numbers of transactions. What is need is a computer-based strategy for handling these situations. The strategy does not need to be the perfect negotiation, although it must be competent, but it must be able to deal with more negotiations more rapidly than a human operator could. A core objective of this work is to develop strategies that are able to make a profit in a situation where customers are continually requesting bundles of products and may need to be negotiated with, suppliers must be negotiated with and there is a limit to both the communication capacity available and the amount of information about the market place.

The simple supply chain model (SSCM) was developed to allow the description of simple market scenarios with the aim of developing strategies to tackle such scenarios [5]. The SSCM however, while simple, is not trivial to tackle. As a result various scenarios have been developed to further restrict the use of the SSCM and so provide an incremental approach to tackling the problem [6]. Having defined the scenarios to be used to begin tackling the SSCM the next step was the development of a set of strategies to tackle the first of these, Scenario One [7]. The most important aspect of the developed strategies is that of the middleman strategy. This document describes a way that the strategies may be tested and the middleman strategy optimised.

The document is split into two sections. The first describes the ideas behind the testing of strategies in a simulation system. The second describes the way in which evolutionary computation (via population based incremental learning (PBIL)) may be used to optimise parameters for the middlemen allowing them to perform better in diverse environments.

1.2 Testing the SSCM Scenario Strategies

1.2.1 Introduction

The SSCM provides a way of completely describing a trading situation and the interaction protocols that may be used by the various participants. The SSCM Scenarios further define a given situation by placing constraints on the behaviour of participants and on the use of the SSCM. Having provided these constraints it is then possible to develop strategies for each of the participants in an attempt to tackle the defined problem.

To test a Scenario strategy it would first be necessary to create some form of simulation software. This software would take the full SSCM definition of the problem and then allow the participants to follow there given strategies time-step at a time for the full period defined within the SSCM. Within this system it would be possible to evaluate a strategy based on various criteria including the total funds available to each participant at the end, the number of customers satisfied and so on.

The middleman strategy in Scenario One has a considerable number of parameters to be set for any given experiment, future strategies for the middlemen or perhaps other participant will also have a large number of parameters that need to be set. These parameters will have a bearing on performance within the simulation system described. Since the number of parameters is large and the possible set of problems encountered also quite wide (even under Scenario One) it is reasonable to consider that some form of parameter set optimisation system would be useful if not critical for obtaining the most out of any given strategy. A simulation system that will allow this must be considered.

1.2.2 Optimisation and Environments

In considering the optimisation of parameters for an SSCM strategy it is first necessary to consider that a wide range of possible environments are available and it is probable that parameter sets will perform differently according to the environment they find themselves being used in. To this end it is unlikely that globally optimal parameter set could ever be found rather, a near optimal parameter set for a specific environment could be found, this set would hopefully outperform other parameter sets developed for other environments. How specific an environment is becomes an issue at this point. An environment could consist of the complete SSCM definition and, undoubtedly, there would be optimal settings for parameter sets under these conditions. However, this definition is not useful. In any realistic application you are highly unlikely to encounter exactly the same situation repeatedly. What is important is to optimise parameter sets for a general category of environments. In one sense the SSCM Scenarios provide an environment definition but the granularity of such environments is relatively poor, a more specific environment is needed.

With respect to Scenario One an environment could be viewed as the number of middlemen and the number of customers each middleman must deal with. Given the simplicity of the strategies suggested for the customers and suppliers under Scenario One it is reasonable to consider that the configuring parameters of these entities also constitutes the environment with the middleman parameter set being the only set of parameters considered for optimisation.

The environment under Scenario One can therefore be defined as the number of middlemen, the number of customer assigned to each middleman and the parameters of the customers and suppliers. This definition is still a little too specific as it ties the customers down exactly. What is needed is a mechanism for expressing a general likelihood that customers will take on certain parameters. This likelihood may then be used as part of the environment setting rather than the specific customer parameters that would force each simulation run to be identical in the same way that a complete SSCM definition would be.

Finally the environment for Scenario one can be defined as:

- The number of middlemen
- The number of customers assigned to each middleman
- The supplier parameters and strategy parameters

- A range of possible requirements a customer may take on

While this environment is fine for Scenario One it is likely to prove an ineffective definition for future scenarios.
The problem still remains of exactly how to define the range of requirements a customer is able to take on. To deal with this it is first necessary to review the customer parameters from within the SSCM.

An SSCM Customer starts with the following information:

- A maximum budget
- A trip duration
- The earliest start time for the trip
- The latest end time for the trip
- The minimum amount of time before the start of the trip that negotiations must be completed
- The set of travel preference, i.e. what entertainments are required.
- The communication availability
- The known middlemen

The last two of these can be handled simply. For communication availability this would be a fixed value for a given environment as it is fundamental to how the strategies perform and any change within an environment might lead to considerably different results. The known middleman is simple to resolve for Scenario One as each customer is only allowed to know about one middleman. Whether the distribution of customer is even or not amongst the middlemen is part of the environment.

Most of the remaining parameters (budget, duration, start time, end time and delay) can also be fairly simply treated. In each case a maximum and minimum range of acceptability should be considered. So for instance in the case of the customers budget a requirement might take on any value between 500 and 1000. For the latest end time these upper and lower limits would act as an increment beyond the earliest start time plus the selected duration. The set of travel preferences represents more of a problem.

The travel preferences parameter includes inbound and outbound flights, accommodation and entertainment. The flights and accommodation can be considered implicit in any definition and would be determined precisely by the other information. The entertainment requirement is thus the remaining element to define. For this it is suggested that a probabilistic approach be taken. For each entertainment item there would be a probability of it being required, this could stretch from 0 (not required ever) to 1 (required in all instances). A further minimum and maximum requirement value would be supplied for each item. The actual total amount required would then be selected from within this range. This mechanism increases the flexibility of the definition further allowing not only a probabilistic presence but also the possibility of variable amounts.

The Scenario One environment is thus fully defined.

1.2.3 The optimisation process

Having defined the environment for Scenario One it is now possible to reconsider the simulation system. While in its basic mode of operation the simulation system must be capable of simply running a fully described SSCM description and its participants, the system would benefit from the ability to optimise parameter sets as has been stated. To this end we have defined that for, Scenario One at least, this means optimising the middleman strategy parameter set.

The basic simulation software must contain elements to represent each of the participants, handle message passing and maintain a universal clock for all participants. This could be achieved in a number of ways from a single configurable software entity to multiple, interacting software agents. The approach taken to realisation is not important at this stage. The more complex simulation system must include the optimisation mechanism – approaches to this mechanism are discussed now.

There are many me methods available for the optimisation or parameters each of which is more or less suitable according to the particular situation. Determining the mechanism to use can be thought of in terms of defining the particular environment in which we are interested. In this case we have a large number of variables to optimise. The effectiveness of a given instantiation of the variables can only be found by running a market simulation – no other mechanism will work. Further since random effects within a simulation may provide bias against an otherwise good parameter set it is necessary to run multiple simulations. The evaluation process is thus slow. Assuming the parameter set of each middleman is identical is potentially dangerous as it fails to take into account how the differences between middleman strategies may effect the simulation out come, a middleman operating differently may have some form of advantage. A result of this is that evaluation within a market simulation where different configurations are being used may provide a less biased view.

The process of finding good parameter sets for the middleman strategy is a matter of configuring the middlemen, running a simulation (or simulations), evaluating each middleman and determining how effective the configurations were. New configurations may be created ready for a new set of simulations. Over time good parameter sets should be found. This process leaves several questions unanswered. The first of these is what mechanism is used to form a parameter set in the first place. Following from this the next question logically is what evaluation metric is used. The final question is how, having evaluated each middleman, should the parameter generation mechanism be adjusted.

The answer to the first of these questions need not be difficult. It is reasonable to suppose that for each parameter in the strategy there are upper and lower bounds that appear sensible or at least would only ever be crossed in extreme circumstances. These bounds then, at the very least, provide a sensible starting point for creating new parameter sets. Each parameter value could be selected from within the range either by some probabilistic process or by common sense judgement of the experimenter.

To address the question of an evaluation metric there are many sources of inspiration and indeed the metric may largely depend on what in particular wishes to be focused on. For instance if the number communications being minimised was a priority then a metric that emphasised this would be appropriate. The current view is that purely the net worth of the middleman is really that which we are interested in. There are various ways in which this may be measured. The middleman’s total funds at the end of the simulation may provide one way.

Given the environment described and the questions outlined how might the parameter optimisation process be addressed? While various options are available evolutionary learning would seem to appropriate due to the natural generations delimited by the simulations being run. A problem exists however in that the population of the evolutionary system being used is effectively limited to the number of middlemen in the simulation. Since the number of middlemen cannot be too large, less the simulations become too time-consuming, this would appear to be a problem. Two possible solutions exist; these would be to use multiple populations along with some form of cross fertilisation between the populations or to use some other technique. Fortunately evolutionary techniques do exist that operate well with limited population sizes. Population Based Incremental Learning [3] is such a technique.

PBIL combines elements of reinforcement and evolutionary learning in a statistical method that is capable of representing a population of solutions without the need for maintaining the full set of solutions itself. PBIL replaces the population of solutions with a probability vector. The probability vector consists of the probabilities of each value for each variable in a possible solution. To handle continuous variables the range of the variable is divided up and a probability is assigned to each block. To generate a solution, a value for each variable is selected probabilistically. To allow the PBIL vector to learn a learning rate (between 0 and 1) is first selected. A good problem solution is then selected by some mechanism and is used to update the PBIL vector. The update is in the form of a probability increase for each value in the selected solution, the increase in probability is the learning rate with a corresponding proportional decrease in the remaining values for each variable. In the case of continuous variables the probability of the block associated with the value is increased and furthermore the learning rate proportion decreases the size of the block, the boundaries focusing in towards the value. By repeatedly selecting good solutions and updating the PBIL vector it will be increasingly probable that good solutions will be generated by the PBIL. To further enhance the PBIL systems operation it is possible to use negative reinforcement learning, that is, to reduce the probability of values associated with bad solutions.

Using the process outlined above in combination with PBIL it should be possible to build a simulation system that is capable of testing and optimising Scenario One strategies. Similar systems in the future should be capable of optimising strategies for the remaining Scenarios although the details of those systems will be left to a later document to describe.

1.3 References

[1] Simple Constrained Bargaining Game
Edward Tsang, Tim Gosling
[2] Determining Successful Negotiation Strategies: An Evolutionary Approach
Noyda Matos, Carles Sierra, Nick Jennings
[3] Population Based Incremental Learning – A Method for Integrating Genetic Search Based Function Optimisation and Competitive Learning
Shumeet Baluja
[4] On Agent-Mediated Electronic Commerce
Minghua He, et al, 2003
[5] Technical Report 1: The Simple Supply Chain Model (SSCM)
Tim Gosling and Edward Tsang
[6] Technical Report 2: SSCM Scenarios
Tim Gosling
[7] Techincal Report 3: SSCM Scenario One
Tim Gosling