Phase II: Optimisation of system dynamics models

Patrick E. McSharry [a,b,c,*],

[a] *Department of Engineering Science, University of Oxford, Oxford, OX1 3PJ, UK*

[b] *Mathematical Institute, University of Oxford, Oxford OX1 3LB, UK*

[c] *Commissioned by The World Bank, 1818 H Street, NW, Washington, D.C. 20433*

Abstract

The initial project entitled "Optimisation of system dynamics models using Genetic Algorithms (GAs)" proved that the GA algorithm has the potential to efficiently provide accurate and robust solutions to relatively complex policy models. A *Matlab* program is capable of translating the raw equations from an *i-think* model into *Matlab* code. A GA algorithm written in *Matlab* is then able to optimise the model and to pass the optimal solution back to *i-think* for further analysis. Following this procedure, it was possible to optimise the Bolivian malaria model, which has 17 variables, each corresponding to a separate investment target. The GA solution was shown to outperform both the policy that is currently being employed and that obtained using human optimisation (the tedious task of attempting to find an optimal solution through trial and error).

The proposed phase two of the project aims to extend this initial research by (i) making it more user-friendly, (ii) adapting the GA algorithm to directly incorporate a budget constraint and (iii) developing the concept of time-path optimisation.

---

* Department of Engineering Science, University of Oxford, Oxford, OX1 3PJ, UK
Email address: patrick@mcsharry.net (Patrick E. McSharry).

1 Objectives

The objectives and results described in this report are an extension to the work carried out in phase one of the project entitled "Optimisation of system dynamics models using Genetic Algorithms (GAs)" [1].

The main aim of this project is that it should be easy to use by all policy-makers. With this in mind, the procedure of translating and optimising the model and importing the optimal solution back into *i-think* should be simple and well documented in an "operations manual".

The budget constraint was originally maintained by adding a penalty to the cost function. While using this approach, the GA algorithm proved to be able to optimise the systems dynamics model. The GA algorithm was, however, found to utilise a lot of its resources on maintaining the budget constraint. A more efficient approach is to build this constraint directly into the GA algorithm. In this way, only solutions that satisfy the constraint are considered throughout the entire optimisation process.

The final part of this project is to consider the optimisation as a process that gives results over a time-path. While the overall objective may be to reduce a particular index over the course of five years, it is also important to view the time-path of this index over the interim years.

The evaluation of different policy strategies can be greatly facilitated by assessing their ability to provide results that have similar time-paths to the ideal policy. This approach will have the advantage of being able to identify policies that can fulfill the required objective both in a shorter amount of time and for a smaller overall cost.

2 Results

2.1 Webpage and operations manual

A webpage `www.mcsharry.net/ga` has been developed for the project. This will provide all the required *Matlab* files so that different users can implement and test the optimisation product using their own system dynamics models. Over time, the feedback and suggestions for improvement will be incorporated into the webpage and the operations manual. Eventually it is hoped that this webpage will provide a *stand-alone* utility for those that want to optimise system dynamics models using the Genetic Algorithm.

2.2 The new GA code with budget constraint

Strategy planning within the field of development typically requires the most effective policy and solution given a fixed amount of investment, US\$ c. Suppose that there are m possible targets for investment such that the optimisation takes place in an m-dimensional parameter space. Each of the parameters x_i corresponds to an expenditure and the budget constraint may be viewed as requiring the solution to satisfy

$$\sum_{i=1}^{m} x_i = c.$$

The latest version of the GA, called `gabudget.m`, takes the budget constraint, c as an input value and forces the evolving population of solutions to satisfy this constraint throughout the entire optimisation process. This modification of the algorithm provides a much more effective means of obtaining an optimal solution.

In addition, the operators in the GA code for (i) selection, (ii) crossover and (iii) mutation have all been fine-tuned using simple test problems with multiple equilibria. These operators each have a single parameter to control the probability of their execution during the optimisation process:

- α - governs the probability of choosing the best solution when selecting which solutions are kept for the next generation
- β - crossover rate controlling the probability that selected solutions will be combined to give new offspring
- γ - mutation rate gives the probability that mutations will take place

Finally, the new GA code is capable of accepting a *best-guess* starting solution, x_{start}. This will be particularly useful in situations where the purpose of carrying out the optimisation is to test whether a given policy can be outperformed. If the GA returns the same optimal solution, x_{opt} as the best-guess starting solution, then the policy-makers can rest assured that their current strategy is optimal. On the other hand, if the GA returns a different optimal solution, this could then be implemented in order to improve the policy.

2.3 Time-path optimisation

While the primary objective of policy-making may be to optimise the level of a particular indicator at some time in the future, it is also important to consider the time-path of the indicator during the intervening time. The importance of

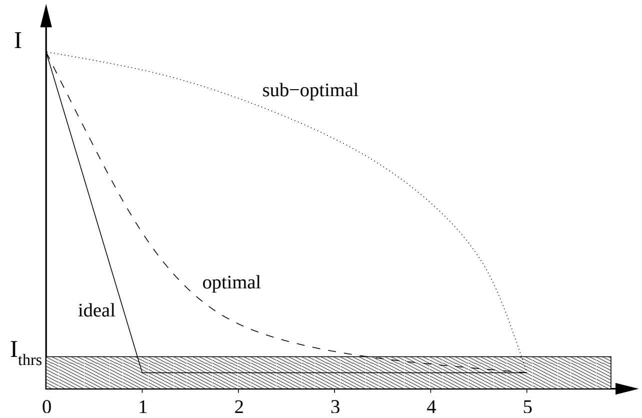

Fig. 1. *Optimisaton process showing the time-paths of ideal (solid), optimal (dashed) and sub-optimal (dotted scenarios where the objective is to reduce the index I to I_{thrs} in five years or less.*

the time-path of the index may be seen by studying the hypothetical situation illustrated in Fig. 1. Consider an index I that can be used to quantify the impact of implementing a particular policy. Suppose that the objective is to reduce this index to less than I_{rmthrs} as shown by the grey area in Fig. 1. If the index I is measured at the end of each year, then the ideal scenario (solid line) would be to decrease I below I_{thrs} in just one year and to maintain this level from then onwards. In reality, an entire family of scenarios are likely to exist that lie between the optimal (dashed) and sub-optimal (dotted) policies. While both the optimal and sub-optimal policies have time-paths that are capable of achieving their objective in five years, their temporal characteristics are extremely different and the policy-maker would ideally like to select the optimal policy. By simply optimising the index obtained at the end of five years, the optimisation technique cannot distinguish between the optimal and sub-optimal scenarios. For this reason, it is also necessary to consider the entire time-path during the optimisation process. This can be achieved by minimising the distance between the time-path of the ideal scenario (solid) and that of each of the candidate scenarios being tested during the optimisation process.

For an indicator (such as the annual parasitic index (API) in the case of the malaria model), which is always positive, the quality of a given time-path scenario can be quantified by simply summing up the values for each time step. Obviously ,the closer the indicator is to zero, the better the solution.

The time-path optimisation was applied to the Malaria model, but was found to give the same optimal solution implying that the investment strategy giving the largest decrease in the API over one year also gives the largest decrease throughout the entire time path over four years.

3 Future work

The ability to use the optimisation procedure described in this report depends on the availability of an accurate model. The accuracy of a model can be evaluated by assessing its ability to reproduce historical outcomes given suitable initial conditions and parameter values. All models are inadequate and the policy-maker should be aware of the exact limitations of the model. By testing the consistency of the model with the historical observed time series, it is possible to identify model error [2]. An analysis of this kind is particularly important when attempting to determine the effect of omitting certain factors from the model, such as the influence of weather and climate change in the case of the malaria model [3].

References

[1] P. E. McSharry. *Optimisation of system dynamics models using Genetic Algorithms (GAs)*. World Bank, La Paz, Bolivia, May 2004.

[2] P. E. McSharry and L. A. Smith. Consistent nonlinear dynamics: identifying model inadequacy. *Physica D*, 192:1–22, 2004.

[3] J. Newman, Velasco M. A., L. Martin, and A.-M Fantini. A system dynamics approach to monitoring and evaluation at the country level: an application to the evaluation of malaria-control programs in bolivia. In *Proceedings of the Fifth Biennial World Bank Conference on Evaluation and Development, "Evaluating Development Effectiveness: Challenges and the Way Forward*, Washington, D.C., 2003. Operations Evaluation Department.