How to Explore and Manage the Future? Formal Model Analysis for Complex Issues under Deep Uncertainty

Bas M.J. Keijser, Jan H. Kwakkel & Erik Pruyt

Faculty of Technology, Policy and Management; Delft University of Technology
P.O. Box 5015, 2600 GA Delft, The Netherlands – E-mail: b.m.j.keijser@student.tudelft.nl

July 31th, 2012

Abstract

Formal Model Analysis (FMA) covers a group of methods and techniques to study structure-behaviour relations in best-estimate models. That is, FMA aims to identify the structural causes for the particular dynamics of a single best-estimate model. Under deep uncertainty, the notion of a best-estimate model is however troublesome. Then Exploratory Modelling and Analysis (EMA) can be used to handle deep uncertainty. Through EMA, an ensemble of models is created and analysed. This paper argues that FMA can strengthen EMA by assisting in generating plausible dynamics, exploring and analysing future worlds, identifying plausible policy levers and exploring and comparing various policy options under deep uncertainty. The generation of plausible dynamics is facilitated by deliberately changing model structures that have a large effect on the model behaviour. The other three ways of strengthening EMA rely on identifying the model structure that determines desired or non-desired model behaviours. We illustrate the combination of EMA and FMA using Ford's Loop Deactivation Method to explore and analyse the behaviours generated by a generic model of the scarcity of minerals and metals.

Keywords
Formal Model Analysis – Exploratory Modelling and Analysis – Loop Deactivation Method

1. Introduction

1.1 Exploratory modelling for complex and deeply uncertain problems

Formal Model Analysis (FMA) is used to investigate the link between model structure and model behaviour that lies at the basis of System Dynamics. It does this by attributing certain parts of the behaviour, or changes in this behaviour, to model structure. FMA is normally used in "mainstream System Dynamics", thus in *dynamically complex* issues under *shallow* or *medium uncertainty*. *Dynamically complex* situations are situations 'where cause and effect are subtle, and where the effects over time of interventions are not obvious' (Senge, 1990). These situations are caused by interactions between 'systems, markets, institutions, products, regulators, (groups of) actors, and policies/regulations' (Pruyt, 2010a). *Shallow* or *medium uncertainty* plays a role when an analyst is able to 'enumerate multiple alternatives and provide probabilities' or 'rank order the alternatives in terms of perceived likelihood' (Kwakkel et al., 2010).

When, however, *deep uncertainty* starts to play a role in the investigated issue, normal System Dynamics cannot fulfil the needs of a thorough problem exploration and analysis. *Deep uncertainty* is defined as situations 'where analysts do not know, or the parties to a decision cannot agree on: *(i)* the appropriate conceptual models that describe the relationships among the key driving forces that will shape the long-term future, *(ii)* the probability distributions used to represent uncertainty about key variables and parameters in the mathematical representation of these conceptual models, and/or *(iii)* how to value the desirability of alternative outcomes' (Lempert et al., 2003). In this kind of situations System Dynamics is replaced by Exploratory System Dynamics Modelling and Analysis (ESDMA, Pruyt, 2007, Pruyt, 2010b). FMA is also extremely useful in exploring and analysing complex and deeply uncertain problems.

1.2 Uses of Formal Model Analysis in exploratory modelling

ESDMA consists more precisely of *(i)* developing an ensemble of "exploratory" models of the issue of interest, *(ii)* then generating thousands to millions of scenarios (called an "ensemble of plausible futures") by exploring the ensemble of models, *(iii a)* followed by analysing the dynamic behaviours, bifurcations, et cetera and/or *(iii b)* specifying a variety of policy options, and simulating, calculating, and comparing the various options across the ensemble of models (Pruyt et al., 2011).

Uses of this kind of exploratory modelling are numerous. FMA can assist in all these uses by relating model structure and model behaviour. This structure-behaviour link lies at the basis of System Dynamics (Forrester, 1968). Uses of exploratory modelling are as follows. *(i)* Exploratory modelling can be used to generate large numbers of future developments. So it can help to give an idea of the widest possible scenarios that can unfold. *(ii)* Future worlds can be explored and analysed. *(iii)* Policy effectiveness over the complete scenario space can be tested. And finally, *(iv)* exploratory modelling can be used to evaluate and compare policies over the entire uncertainty space. Decent analysis of dynamic behaviours and bifurcations in that behaviour is needed. Investigation of conditions in model structure that lead to the different modes of behaviour could be very useful in each and every of the uses of exploratory modelling as specified. These can be compared to the uncertainty conditions identified by the Patient Rule Induction Method (Friedman & Fisher, 1999).

Using FMA can contribute strongly to the exploration and analysis of dynamically complex and deeply uncertain problems. The rest of this paper will give a first exploration and illustration of using FMA for ESDMA.

1.3 The content of this paper

This paper has the following structure. In section 2, preliminary goals and the enormous potential of connecting Formal Model Analysis and Exploratory SD are stated in detail. Section 3 contains an in depth discussion of EMA and ESDMA, clustering of behaviours, structural analysis of SD models and the Loop Deactivation Method. Section 4 applies the method to an illustrative case after which Section 5 shows the results of the illustration. Section 6 contains the conclusions and a discussion of further research into the joint uses of FMA and ESDMA.

2. Purpose of the combined use of Formal Model Analysis and Exploratory System Dynamics Modelling and Analysis

As further discussed in this paper, FMA can assist ESDMA in all its uses, namely *(i)* to generate large numbers of future developments, *(ii)* to explore and analyse plausible futures, *(iii)* to test policy effectiveness over the complete scenario space, and finally, *(iv)* to evaluate and compare policies over all plausible scenarios. Now the uses of connecting the two methods will be enlightened by looking at its purpose in every use of exploratory modelling.

2.1 To assist generation of plausible future developments

Lempert et al. (2003:31) stress the advantages of thinking in scenarios, but they also name the disadvantage that choosing a small number of scenarios from a complex future is highly arbitrary. Far more than a small number of futures should be considered. Deep uncertainty is thus introduced to the models constructed. This gives the possibility of considering a far larger number of plausible futures.

Uncertainties that are introduced are: *(i)* technical uncertainties about model quantities, *(ii)* methodological uncertainties about model structure, *(iii)* epistemological uncertainties about model completeness/adequacy, and *(iv)* model operation uncertainties like bugs (Pruyt, 2007). Incorporating these uncertainties in SD models is done by sampling preference structures, random patterns, varying lookups, sampling delay orders and times, varying scenarios and using switches to de/activate surprises (Pruyt et al., 2011).

Nonlinearity and the plethora of uncertainties in complex simulation models used, cause the models to be extremely sensitive to changes (Miller, 1998). This sensitivity is used as the way of taking into account deep uncertainty in the system behaviour. It is precisely used to explore "all" plausible futures. *Weaknesses* of a model are seen as possible problems in formulating valid models of a problem (Miller, 1998). In other words: differing model assumptions give differing model behaviours on these points. If these weaknesses can be found, they can be exploited to construct even more plausible behaviours. Formal Model Analysis methods can exactly be used to find parts of the model structure, which cause large changes in the behaviour. These structures can then be changed, i.e. loops can be systematically and exhaustively turned on/off, which results in lots of additional plausible behaviours.

A further use of combining Formal Model Analysis and Exploratory System Dynamics to find weaknesses is to assist model breaking and validation (Miller, 1998). Then it is used to explore structural conditions under which a model breaks down by giving non-valid results (for instance negative values). This can assist in validating models used for Exploratory System Dynamics. Non-valid behaviours are namely automatically non-plausible. This analysis thus gives an idea of which structures cannot be varied, because non-plausible behaviours would then result.

2.2 To assist exploring and analysing future worlds

After a large number of future worlds have been generated, the behaviour patterns can be clustered, explored and analysed. In doing this, the general relation between the model structure and its changes due to uncertainties, and the model's behaviour can be investigated with FMA-methods. When for instance a future development is found that shows an interesting (desirable or non-desirable) behaviour in a certain time period, the responsible feedback loops for that behaviour can be found by means of FMA-methods.

With this exploratory knowledge plausible developments can be better understood in terms of what causes behaviour and bifurcations in that behaviour. Knowledge of which structure turns behaviour into (non-)desirable modes can help in understanding the real-world system structure and the causal effects that play a role over the complete uncertainty space. When the method is fully automated, Loop Deactivation and other FMA-methods can be seen as some of several instruments that have been/can be developed to analyse the causes of behaviour (changes).

2.3 To assist research of policy effectiveness in the complete scenario space

Structural policies that change a loop contribution can be constructed. FMA-methods can assist in investigating the loop change effects on the behaviour. But first the structure of the uncertainty space must be explored. The Patient Rule Induction Method (Friedman & Fisher, 1999) can be used to find scenario concentrations with undesirable outcomes in terms of which uncertainties are important. Also, Time Series Clustering (Liao, 2005) can be used to identify clusters of non-desirable behaviours. When the desirability results have been derived, policy effectiveness can be tested. This is done in terms of how effective a policy can change non-desirable into desirable modes.

FMA can assist in assessing the importance of the feedback loop that is influenced by a policy. If the influenced loop is very important in determining the outcome variables' behaviour, the policy effectiveness is large. In other words, if the dominance contribution of the loop is large, the policy's leverage will also be large. The other way around, FMA can help to identify loops that would help in designing influential policy measures. The way to do this is to search for the loops that cause non-desirable behaviour modes; these loops should subsequently be influenced so that desirable modes of behaviour are found.

2.4 To assist evaluation/comparison of different policies over all plausible scenarios

FMA-methods can also assist in the evaluation and comparison of different policies over all plausible scenarios. There is a possibility that different policies have been constructed that influence different parts of the model structure. A comparison can then be made on grounds of the importance of the influenced loops. The policy that influences the most important loop has the largest leverage over the model's behaviour. There is of course still a need of being able to classify behaviour modes into desirable and non-desirable modes.

Besides comparing different policies, combination policies can be constructed that influence completely different parts of the structure. The reason for this can be to minimise non-desirable interactions between the policies. These interactions can have non-desirable effects on the amount of means necessary to "use policies". FMA can then be used to investigate the effect of changing the contribution of multiple loops at once, and herewith derive results on the effectiveness of multiple measures being taken at once.

In connecting the virtues of FMA with the uses of ESDMA, we have found several reasons to want to use Formal Model Analysis methods to assist exploratory modelling. These reasons are that FMA can assist in: generating large numbers of plausible dynamics, exploring and analysing future developments, investigating policy effectiveness under deep uncertainty and evaluating/comparing different policies over the complete uncertainty space. The connection between FMA and ESDMA will thus be mended in the rest of this paper.

3. Method

In this section the different parts of the method used in this paper will be introduced. We will start by introducing Exploratory System Dynamics Modelling and Analysis in some more depth. Then behaviour pattern clustering and its uses in exploratory modelling will be explained, after which the structural analysis of SD models is enlightened. Finally, an introduction to the Loop Deactivation Method, the Formal Model Analysis method that will be used in the illustration, is given.

3.1 ESDMA

In the first part of the introduction the need for replacing mainstream System Dynamics under *dynamic complexity* and *deep uncertainty* has been explained. System Dynamics is in such situations substituted by Exploratory System Dynamics Modelling and Analysis. This is a combination method that combines Exploratory Modelling and Analysis (EMA) with System Dynamics as scenario generator. The different parts of this method will now be introduced in a more detailed way.

Mainstream System Dynamics is used as a method to investigate *dynamically complex* problems. *Dynamically complex* situations are situations 'where cause and effect are subtle, and where the effects over time of interventions are not obvious' (Senge, 1990). These situations are caused by interactions between 'systems, markets, institutions, products, regulators, (groups of) actors, and policies/regulations' (Pruyt, 2010a). When *deep uncertainty* however starts to play a role, System Dynamics cannot be used any more. Thus System Dynamics, which is appropriate for *dynamic complexity*, is combined with EMA, which is appropriate for *deep uncertainty*.

EMA is a multi-method for dealing with uncertainties. Exploratory modelling is said to be 'the use of series of […] computational experiments to explore the implications of varying assumptions and hypotheses' (Bankes, 1993:435). This type of exploratory modelling is specifically intended to deal with *deep uncertainties,* these uncertainties are defined as situations 'where analysts do not know, or the parties of a decision cannot agree on:

- the appropriate conceptual models that describe the relationships among the key driving forces that will shape the long-term future [e.g. different drivers and underlying structures than today],
- the probability distributions used to represent uncertainty about key variables and parameters in the mathematical representations of these conceptual models, and/or

- how to value the desirability of alternative outcomes' (Lempert et al., 2003)

Due to this kind of uncertainties, 'there are many models that might plausibly represent the system of interest' (Bankes, 1993:441). Lempert et al. (2003:46) speak of assembling 'futures as a challenge set against which to test the robustness of alternative strategies'.

When using the SD method in an EMA study, the following steps are taken:

- Conceptualise the key uncertainties associated with the problem under study;
- Develop one or more SD models that can be used to explore the key uncertainties;
- Explore the dynamics of the ensemble of models over the specified uncertainties;
- Define (a) decision rule(s) on desirability, e.g. minimum regret for relative robustness;
- Define and test different policies;
- Explore the acceptability or robustness of the policies (adapted from Pruyt, 2007:15).

Advantages of ESDMA are according to Pruyt (2007:19) amongst else that the analysis leads to the *systematic* search for robust policies, that very broad uncertainty boundaries can be explored, that multiple different world views and assumptions can be tested at once and that robust compromises can be searched for in accordance to different stakeholder views and preference profiles of a problem.

3.2 Time series clustering

Exploring the uncertainties results in the generation of a large ensemble of transient scenarios. The next step is to analyse the dynamic behaviour and possible bifurcations in this ensemble. This is needed because of the impossibility of overseeing a number of behaviour patterns that runs into the tens or hundreds of thousands, or even more. To do this, time series clustering is applied.

The goal of clustering is to organize an unlabeled data set into homogenous groups where the similarity within the group is minimized and the dissimilarity between groups is maximized (Liao, 2005, Theodoridis, 2003). In general, time series clustering approaches try to modify existing clustering approaches for static data so that they can cope with time series data. Either the algorithm is modified to deal with the raw time series data, or the time series are processed in such a way that static clustering methods can be used directly. A substantial portion of the research on time series clustering focuses on modifying the similarity measure used in a clustering method to handle time series data (Keogh & Kasetty, 2003).

A possible way is by seeing the behaviour as a concatenation of Atomic Behaviour Patterns (ABP, Ford, 1999). By combining ABPs with the sign of the first order derivative, six different behaviours are possible (i.e. positive logarithmic, negative logarithmic, positive exponential, negative exponential, positive linear, negative linear). Each time series can then be converted into a concatenation of these patterns, and is thus clustered (Yücel, 2012).

We combined the behavioural distance metric with a bottom up hierarchical clustering procedure, part of the SciPy library for scientific computing (SciPy Reference, 2012). In order to perform the hierarchical clustering, the Farthest Point or Voor Hees algorithm was used, which means that the inter cluster-distance equals the maximum distance between cases in cluster *i* and *j*. Next, we used the hierarchical clusters, to form a fixed number of flat clusters.

3.3 Structural analysis of System Dynamics models

Model structure analysis is a method that centres on the structural complexity of SD models to make sense of it (Oliva, 2004). It borrows heavily from properties of mathematical graph theory. It sees a SD model as a directed graph. Quantification of the relations is not possible in the graph shape; it only looks at structural relations between the variables. Feedback loops present in the model thus do show up in the graph. Phaff (2006, paragraph 4.1) expands on the graph theory view of SD models.

Graphs can be visualised in a number of ways. One of this ways is by drawing the graphs nodes and subsequently connect them whenever an edge exists between two nodes. There are also formal ways that make use of matrices. The adjacency and reachability matrix are two oft-used matrices that show some of the graphs structure. The first matrix is a square matrix with as size the number of nodes in the graph; each row and column represents a node. When there is a connection between node 1 (source) and node 2 (destination), the entry on the first row and in the second column is one. The reachability matrix has the same size and works with the same basic principles as the other matrix. Now an entry is made one whenever a node is reachable starting from a given node and walking through the directed edges of the graph. In other words, whenever a node is 'antecedent to' the destination node (Oliva, 2004).

Besides looking at structural complexity, model structure analysis can be used to analyse the feedback complexity and the causal structure of the model. The next important tool in this respect is the use of partitions, which is dividing a graph into different subsections based on its structure. Two important partitions used are level and cycle partitions.

A level partition looks at variables' dependency on other variables. 'The variables in the first level are those that do not have any successors outside of their predecessor set' (Oliva, 2004:319). These are the outcome variables. The variables in this level are then deleted from the reachability matrix, after which the same is tested. Cycle partitions further divide a level partition into clusters of variables that 'share the same predecessor and successor set in a reachability matrix' (Oliva, 2004:320). All elements of a cycle partition are thus reachable from all other elements. Cycle partitions are strongly connected graphs (Balakrishnan, 1997). Algorithms to construct the partitions are found in Oliva (2004).

The reason for explaining all these structural analysis concepts in some detail here is that the last step of a typical SD model structural analysis is constructing the (Shortest) Independent Loop Set. This loop set is representative for all feedback complexity of a SD model (Kampmann, 1996). It can thus be used in further analysis of the structure-behaviour link.

3.4 The Loop Deactivation Method

In the first two sections we have shown the possible purposes of connecting Formal Model Analysis and Exploratory System Dynamics. But we still need to make a choice as to what Formal Model Analysis method to use in the first illustration. Formal Model Analysis methods have been developed to research the link between structure and behaviour. Mainstream SD modellers often see these formal methods as too complex or as having a too mathematical nature to apply in everyday simulation. Phaff (2006) gives an extensive overview of the most important methods, namely: Eigenvalue Elasticity Analysis (EEA), of which the idea goes back to Forrester (1983), Pathway Participation Metrics (Mojtahedzadeh et al., 2004) and the Loop Deactivation Method (Ford, 1999). This paper will illustrate the integration of Formal Model Analysis and Exploratory System Dynamics Modelling and Analysis by using the Loop Deactivation Method. The reason for this is the relative ease of implementation when compared to EEA. Pathway Participation Metrics were not considered. Phaff (2006: 50-52) gives reasons for this omission.

Ford (1999) describes a behavioural approach to feedback loop dominance analysis. This approach, here called the Loop Deactivation Method, links the structure of the model in terms of feedback loops to its behaviour in certain time periods of the simulated behaviour. A good illustration of the method is found in Phaff et al. (2006). The behavioural method follows a number of steps, which are (adapted from Ford, 1999: 10-11): *(i)* a variable of interest (VoI) is identified, *(ii)* then a time interval during which the VoI displays behaviour in a certain Atomic Behaviour Pattern (ABP) is chosen. *(iii)* The responsible structure for the behaviour of the VoI is found for every period the ABP changes by looking for changing ABPs if loops are deactivated. *(iv)* This is done by *deactivating* the Independent Loops one-by-one. From this description a number of concepts will first be explained more thoroughly.

The first such concept is the Atomic Behaviour Pattern. Ford (1999) defines this as follows.

linear atomic behavior pattern	$\partial(\lvert(\partial x/\partial t)\rvert)/\partial t = 0$
exponential atomic behavior pattern	$\partial(\lvert(\partial x/\partial t)\rvert)/\partial t > 0$
logarithmic atomic behavior pattern	$\partial(\lvert(\partial x/\partial t)\rvert)/\partial t < 0$

With 'x' the VoI. For every point on a researched behaviour this ABP can thus be calculated. Then the behaviour of the VoI is divided into phases on the basis of changes in this ABP. In each of these phases a separate analysis of loop dominance is executed. This analysis of loop dominance is performed by constructing control variables with which all (analysed) loops can be switched off independently. A loop is said to be dominant in a certain phase if when this loop is switched off during that time period the ABP of the VoI changes. It is also possible that no single loop is dominant in a certain phase. Then combinations of loops are switched off at once, in this case a shadow loop also plays a role. A deeper analysis with more explanation of the procedure can be found in Ford (1999) and Phaff et al. (2006).

In earlier work some problems with the method have been identified (Phaff, 2008). Firstly, the method as introduced by Ford, starts from the assumption that interesting feedback loops to analyse are known beforehand. Phaff (2008) proposes to use the Shortest Independent Loop Set as developed by Oliva (2004) to overcome computational problems when using all the loops in the model. One of the other points where the method can be strengthened is in formulating how to switch off a loop. Phaff is not conclusive in which method is the best to switch off loops. Huang et al. (2010) then pick up where Phaff has stopped.

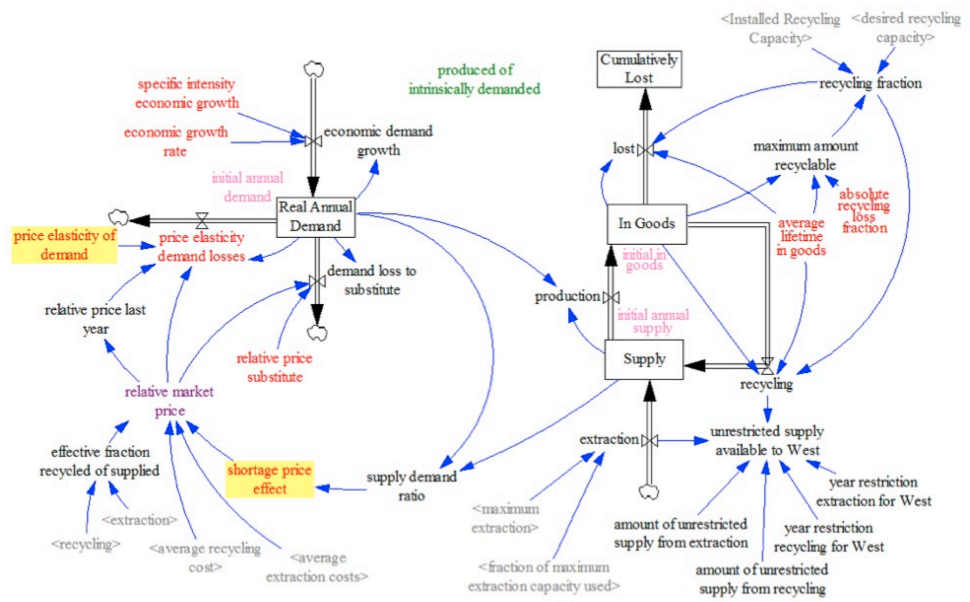
4. Illustration of the method

In this section we will give the first illustration of the connection between Formal Model Analysis and Exploratory System Dynamics. For this we will repeat part of the analysis shown in other work and then apply loop deactivation to certain behaviour patterns.

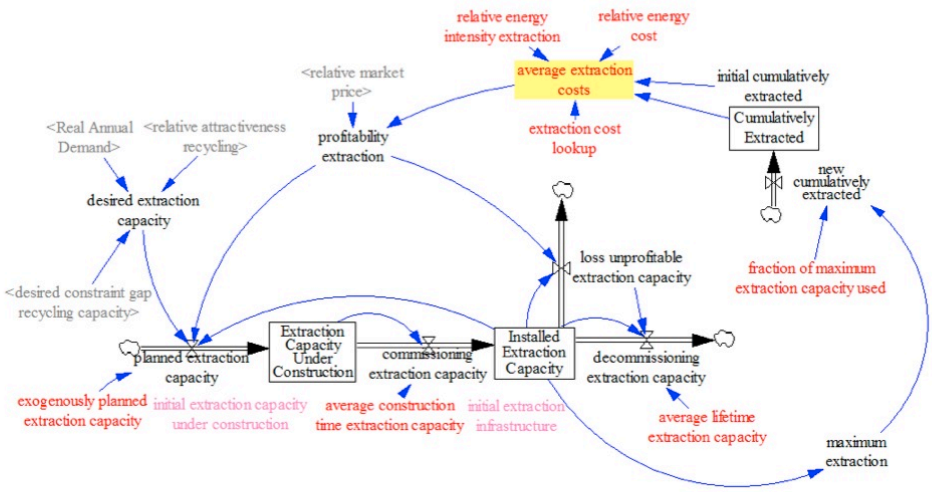
4.1 The model: metals & minerals

The model that is used is extensively described in Pruyt (2010b). It is a generic model for scarcity of minerals and metals. This model is chosen because it is of reasonable size to use as input for the explorative automated Loop Deactivation analysis that follows. Besides that there are a number of named loops (Pruyt, 2010b), so conceptually the model is very understandable. This aids in showing the uses of the proposed connection between different methods. Now some of the explanation about the model used will be given. Information about the issue in reaction to which the model was build can be found in Pruyt (2010b).

The model consists of three connected stock-flow diagram parts. These are 'demand and supply', 'extraction capacity', and 'recycling capacity'. These stock-flow diagrams can be found in the following figures, in respectively figure 1a, b, and c. The described model has earlier been used in an exploratory modelling study (Kwakkel & Pruyt, 2011).



(a)



(b)

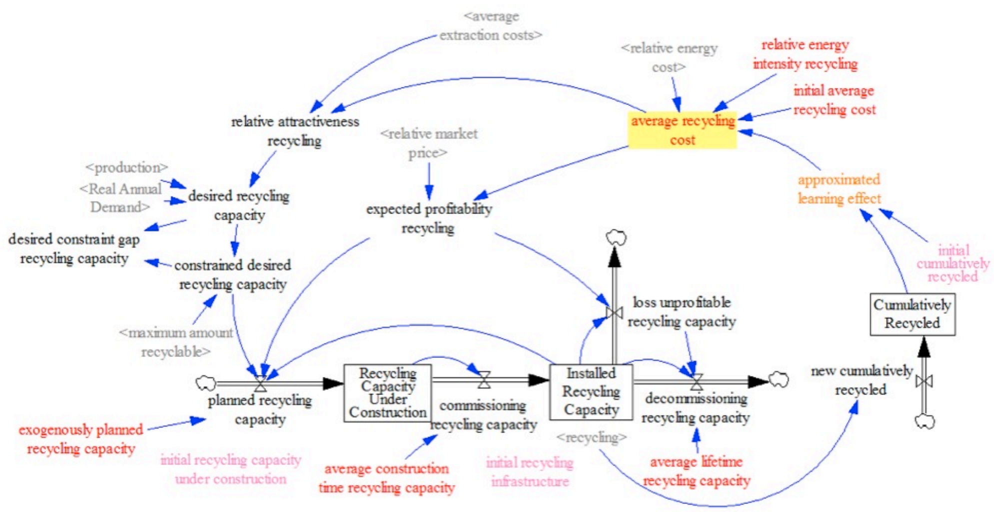


Figure 1 (c) – Stock flow diagrams of the different views of the model

To give a complete overview of the model used the complete causal diagram with the different views and most important feedback loops is repeated in figure 2. This causal diagram shows the interactions between demand/supply, extraction and recycling. Besides that delays, which play an important role in the systems behaviour, are diagrammed.

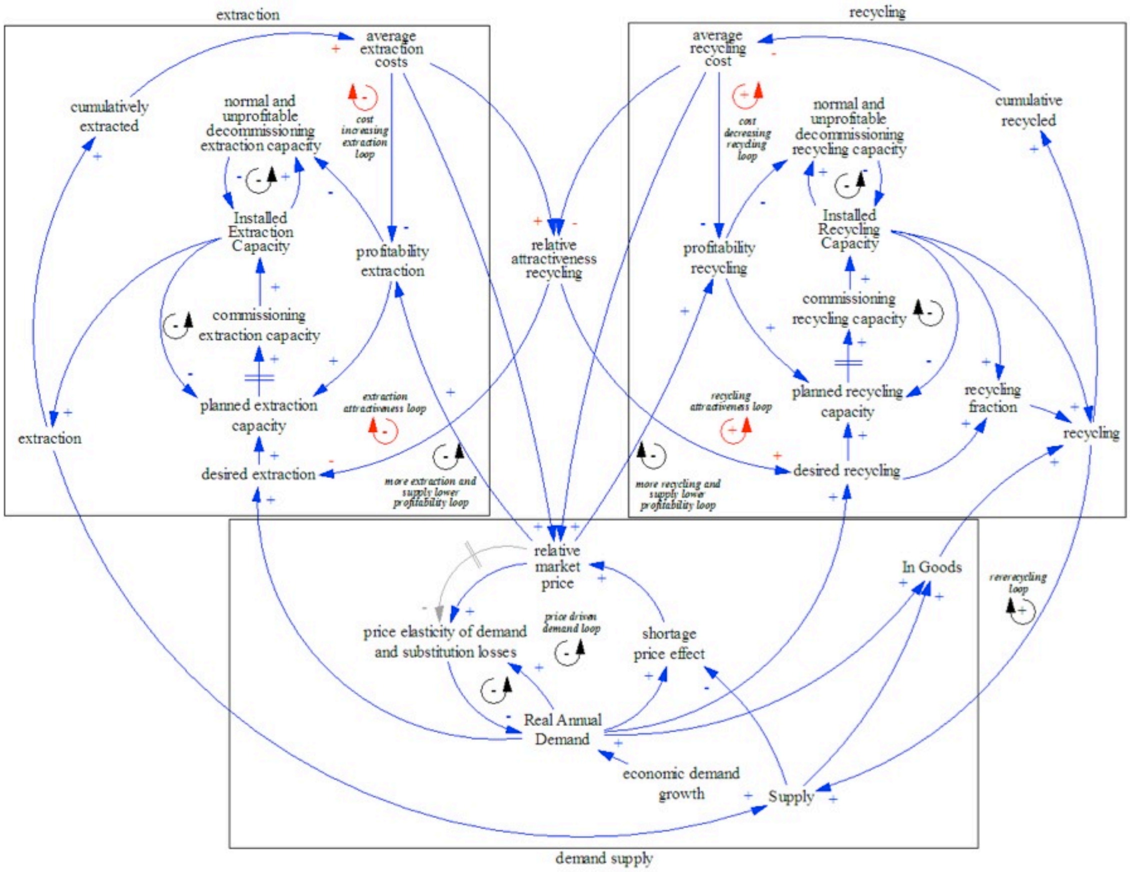


Figure 2 – Causal loop diagram of the complete model, different views are in different blocks

4.2 Introducing uncertainties

To start the exploratory modelling study a plethora of uncertainties is added to the model. This is justified by the fact that the future of extraction and recycling are intrinsically uncertain. However, plausible behaviours can be contemplated. The uncertainties are summarised in table 1. It can be observed that both parametric and structural uncertainties are added (Kwakkel & Pruyt, 2011).

Table 1 – Categorisation of uncertainties used

Name	Description	Ranges
Parametric uncertainties	A wide variety of parametric uncertainties are explored, including the lifetime of mines and recycling facilities, the initial values, and behavioural parameters such as price elasticity and desired profit margins.	Typically plus and minus 50% of the default value
Orders of time delays	There are various time delays, such as the building of new recycling capacity and mines.	First order, third order and tenth order, thousandth
Non-linear lookups	There are various non-linear relations, modelled with lookups. Examples include learning effects, the impact of shortage on price, and substitutions in case of shortages.	Start, end, slope

4.3 Exemplars from the behaviour

When uncertainties have been added to the model, plausible futures can be generated. For this illustration we have generated 1000 model runs. The variable of interest that is chosen is 'relative market price' of the metal or mineral to which the model is tailored. This is the most important variable to get an idea about the market situation. 'Relative market price' namely has an influence on, and is influenced by, the recycling and extraction profitability and on the supply/demand-dynamics. Figure 3 shows all simulated behaviours, with a kernel density plot of the end states at the right.

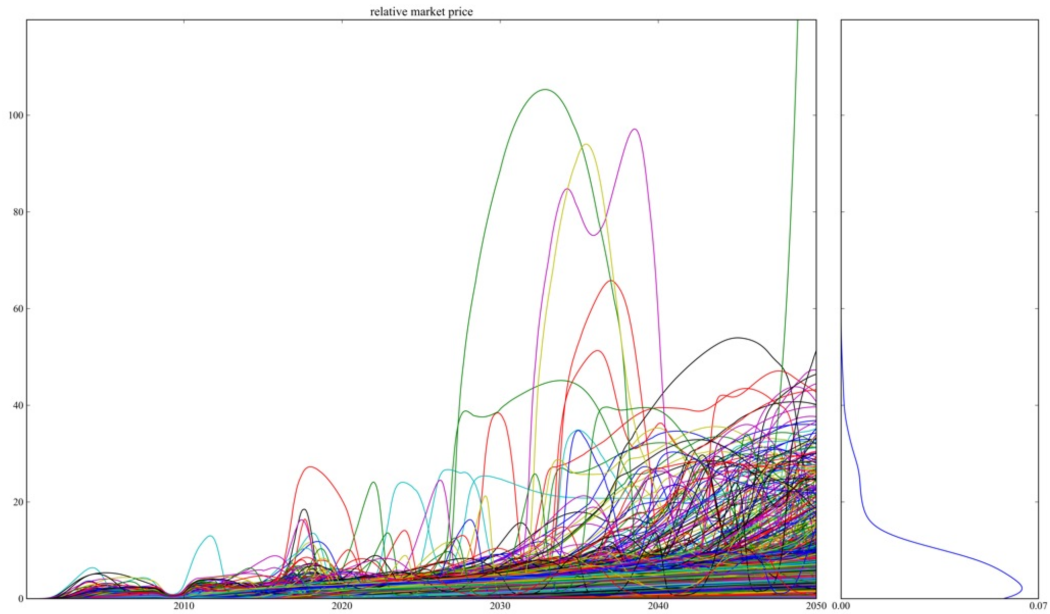


Figure 3 – Plausible future developments for relative market price, 1000 runs

To illustrate the proposed connection of methods we need to have a far smaller number of plausible behaviours. There is thus a need for time series clustering. This clustering is already explained in subsection 3.2. Clustering is used to find 20 behaviour clusters. From these we have chosen 10 exemplary behaviour patterns (see figure 4) on which we will apply FMA.

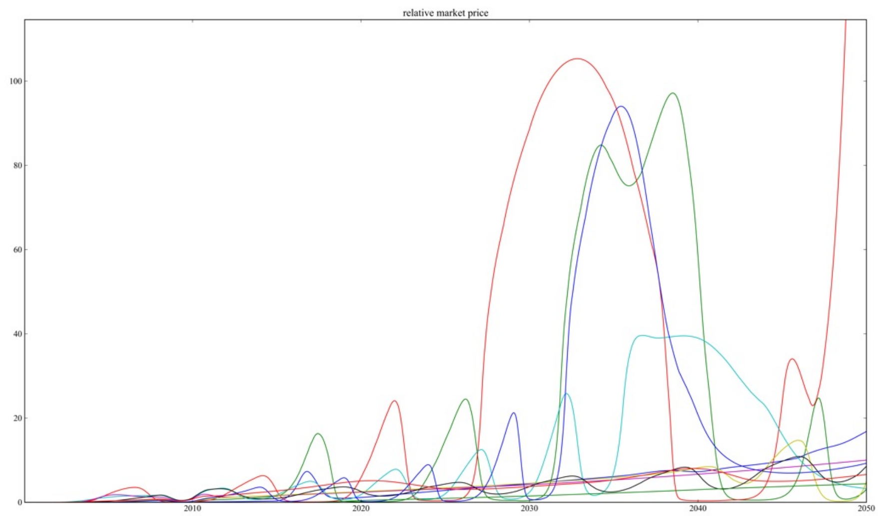


Figure 4 – Ten exemplary behaviour patterns for relative market price through time

4.4 The Shortest Independent Loop Set

In the analysis of the generic model for metals & minerals scarcity we will use the Shortest Independent Loop Set (SILS) as proposed by Oliva (2004). Phaff (2008) proposed to use this smaller loop set as input for the Loop Deactivation Method. To generate the Loop Set, we have written a number of scripts in the Python programming language (van Rossum, 1995). In writing this code we have used certain functions of the NumPy (NumPy Reference, 2012) and NetworkX-libraries (Hagberg et al., 2008) to implement algorithms set forth by Oliva (2004).

Appendix 1 contains the variables in the 47 loops found to be in the SILS(s) of the generic model for metals & minerals scarcity. In fact, the model contains two cycle partitions, one partition with two variables and only one loop. The other partition contains 46 loops in a SILS. We will only look at the structure and behaviour of the loops in the large cycle partition, the other partition being of minor interest due to its size.

The length of the loops found differs very much. There are, for instance, ten loops that contain only two variables. These are influences between stocks and their in- or out-flow(s). Furthermore, there are loops in the SILS of lengths of ten to fifteen variables. Now that we have found the SILS, we can start to apply the Loop Deactivation Method to investigate structure-behaviour relations.

4.5 Applying the Loop Deactivation Method

Finding loop switch locations
The application of the Loop Deactivation Method proceeds in a number of steps. With a subset of the total number of loops in a model (the SILS), loops must be made suitable to be switched off. Methods for switching off loops rely on unique parts of the loop's structure in terms of unique edges and/or nodes. Phaff (2008) states the first possible method to identify a loop switch method, looking for edges that occur in only one loop. He then formulates three different ways of switching the loops off (setting a link gain to zero, using a steady state value and using the value at the time of elimination).

Huang et al. (2010) subsequently proposed another method because in most cases not all loops in the SILS can be deactivated with the first method. They use a method that looks for two consecutive edges that uniquely identify a loop in the SILS. The structure is then altered such, that the loop studied does not exist anymore, but other loops in the SILS are not affected.

Both methods are implemented in Python, by summing adjacency-style matrices and checking which edges occur only once in all loops. When looking only at unique edges, the algorithm identifies 23 loops that can be switched off. For two unique consecutive edges per loop another 20 are switchable. The SILS of the large cycle partition contains 46 loops in total, so there are still 3 loops that cannot be switched off. We choose to only switch the first 43 loops and not switch multiple loops together, because of the fact that this paper is a first exploration of the possibilities of using the Loop Deactivation Method in Exploratory Modelling.

Specification of loop switches
Due to the impossibility of creating variables in the Vensim model with the used Python Workbench, the model structure must be altered manually. This is a point of attention for future application of automated Loop Deactivation. Both methods of switching loops off need a different specification of switches. The first method relies on cutting a unique edge in two parts and adding a variable that results in the structure of figure 5. The edge from 'Installed Recycling Capacity' to 'decommissioning recycling capacity' is cut in two. 'Switch loop 20' is multiplied by 'Installed Recycling Capacity' and then added to 'value loop 20'. This value is used to vary over the three elimination methods.

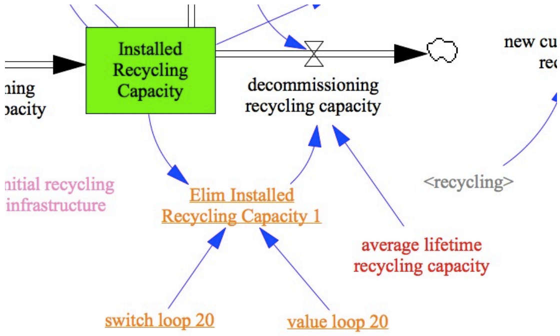


Figure 5 - Model snapshot that exemplifies the first elimination method

The second method, used when a loop contains two consecutive unique edges, uses another way of loop elimination (Huang et al., 2010). Figure 6 shows an example of how a loop with two consecutive unique edges is turned off. The consecutive two edges in this case, were the edges between the variables 'returns to scale', 'average recycling cost' and 'relative attractiveness recycling'. To switch the loop off, a representing variable for 'average recycling cost' has been constructed. This extra variable is calculated in almost the exact same way as 'average recycling cost', except for the fact that 'returns to scale' is substituted with a constant value. The last step is to add a switch that makes sure that only when the loop is switched off the representing variable's value determines the value of 'relative attractiveness recycling'.

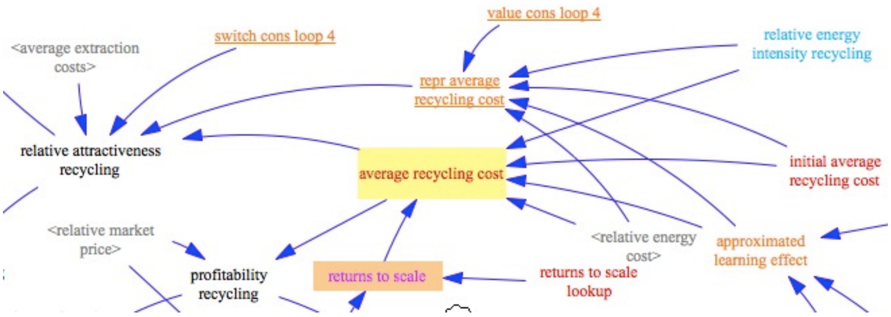


Figure 6 – Model snapshot that exemplifies the second elimination method

Switching loops off

Switching loops off can be done in three ways (Phaff, 2008). These are: setting a link to zero gain, to steady state gain, or to the value it has at the time of elimination. All three methods can be implemented with the proposed model structures of the preceding paragraph. The switch and constant value in every method are given in Table 2. We use the third method as default, setting the constant value to the source variable's value at the time of elimination. A choice for an elimination method may also be given in by the purpose of the model study done as identified in section 2. It is possible that when generating larger and larger numbers of plausible behaviours, a different elimination method is used because larger changes in the variable of interest are caused.

Table 2 – Switch and constant value for different methods of eliminating source to destination-link

For both elimination methods	Switch	Constant value
Zero gain	0	0
Steady state gain	0	$\text{source}_{\text{steady}}$
Time of elimination gain	0	$\text{source}_{\text{elimination time}}$

Determination of time intervals

After the loop switches have been constructed the Variable of Interest's behaviour must be cut into time intervals that show a single Atomic Behaviour Pattern, namely linear, exponential or logarithmic behaviour. However, the model behaviour exhibits such jumpy behaviour that cutting it into ABPs results in tens to hundreds of intervals in only eight hundred time steps. Furthermore, we have applied a filtering of the behaviour pattern to bring the number of behaviour patterns down. An illustration of this filtering is given in figure 7. The light and darker grey separate consecutive ABP-intervals. When an interval of one time step is found, that interval is not coloured.

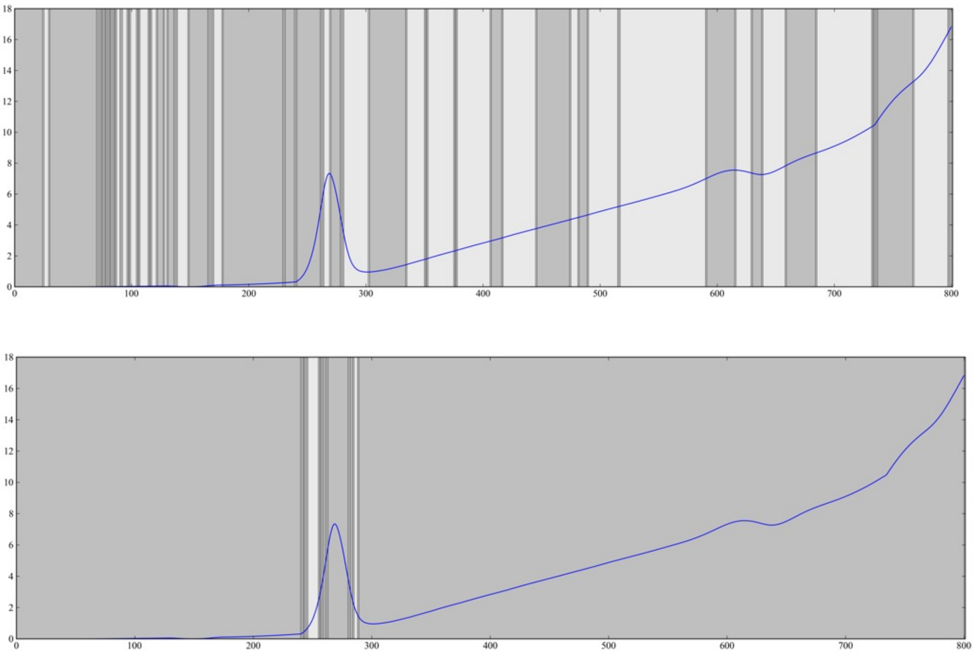


Figure 7 – Illustration of behaviour pattern filtering to determine Atomic Behaviour Pattern Intervals

After filtering the behaviour pattern a far smaller number of intervals is identified. As can be seen in the right-hand part of the behaviour there are some pattern changes the methods filters out, while some of these changes subjectively seem to classify as ABP-changes. It is possible to change the threshold value on the basis of which the data series are filtered; this would give slightly other sized intervals. There is a trade-off between detail of interval identification and number of intervals.

We have not found a good solution for this trade-off yet. In this analysis only time intervals of a considerable number of time steps length will be looked at. We have chosen to look at the longest ABP-interval for each unfiltered behaviour.

Changing model structure in every interval

In every interval of interest the model structure will subsequently be changed so to switch off loops. For this purpose we have again written a number of Python scripts implemented within the Python Workbench used. Firstly, the model uncertainties are set according to one of the behaviour patterns identified. In other words: the same experiment is instantiated again. Then, the model is run until the start of the time interval we are interested in, and the switch and constant value are set to eliminate a certain loop (we use the third elimination method). After which the simulation is continued to the end of the interval of interest. Then we switch the loop on again. Lastly, the model is run until final simulation time.

Which loop(s) are dominant per time interval?

According to Ford (1999) a loop is dominant in a certain time interval when the Atomic Behaviour Pattern in the time interval changes compared to the reference behaviour pattern identified earlier. In a large model as analysed in this paper it is conceivable that multiple loops share dominance in certain time intervals. Furthermore, in assessing loop dominance the same problem with the jumpiness of the behaviour becomes apparent. Drawing conclusions on dominance based on the ABP might cause a far too large number of loops to be called dominant.

We propose looking at loop dominance, or at least contribution to loop dominance, by means of the same distance metrics that are used by time series clustering algorithms. The loop that has the largest distance to the original behaviour pattern is most dominant. In this way it is also possible to speak of relative contribution to dominance for every loop switched off.

5. Results of Loop Deactivation

After the lengthy explanation of section 4 that goes into the different parts of applying Ford's Loop Deactivation Method in an exploratory modelling study, we now show the first results of the combination. A summary of what will be shown is the following. We have chosen 10 runs (partially) subjectively from the 1000 different behaviour patterns from a generic metal/minerals scarcity model. These 10 runs show a number of different types of behaviour. For each of these behaviours we identify the longest Atomic Behaviour Pattern-interval. In these intervals of interest we (try to) switch off all loops in the Shortest Independent Loop Set and look at behaviour deviations that are caused.

Figure 8 shows two behaviour patterns compared to the behaviours when all 43 loops in the model that can be switched off are switched off.

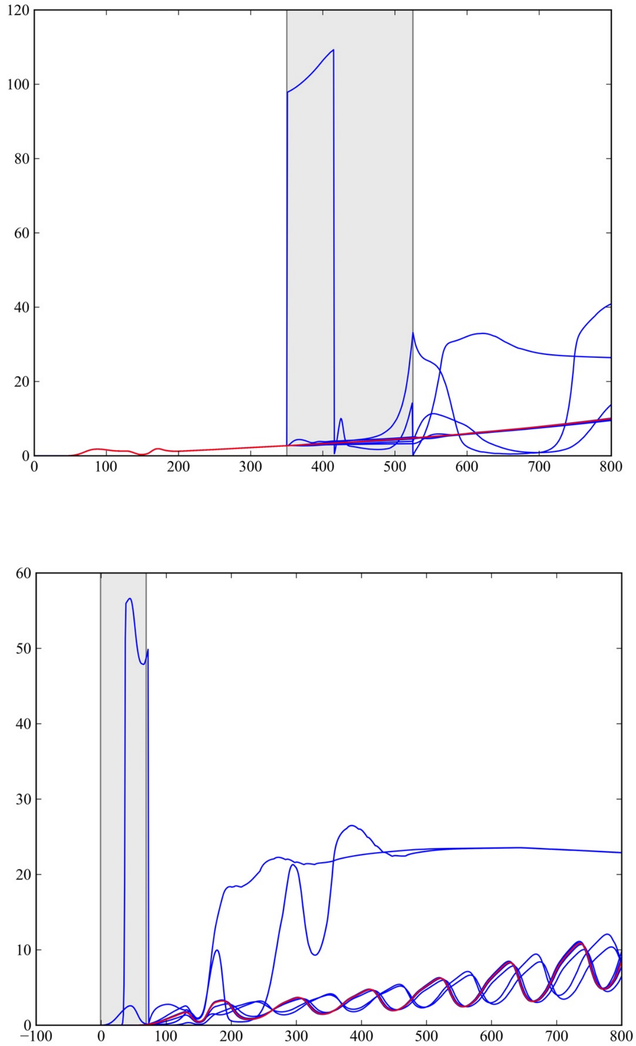


Figure 8 – Switching all loops off for two of the selected behaviour patterns. In grey the interval of interest. The red pattern is the original experiment, blue are the behaviour with loops off. Relative market price is on the y-axis, time on the x-axis.

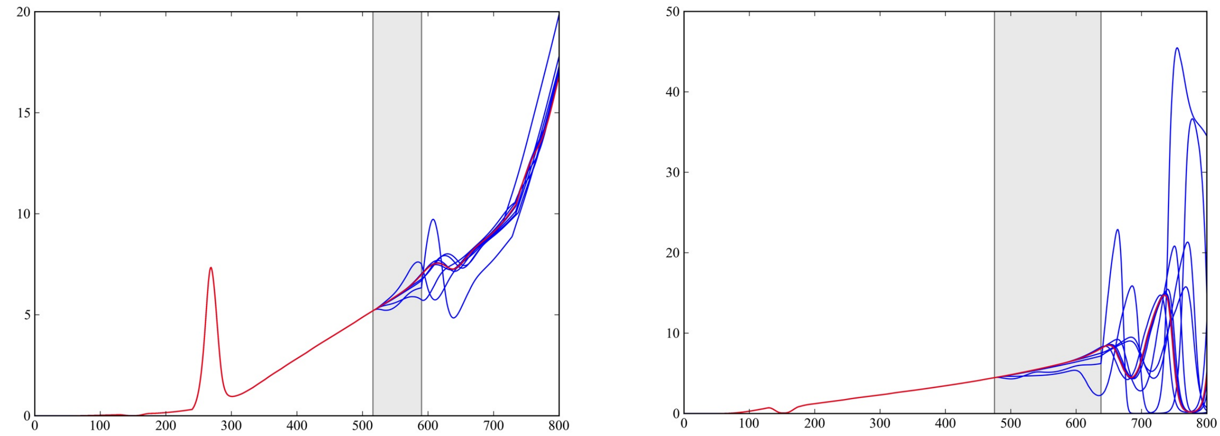


Figure 9 – Switching only those loops off that are switched with the unique edge elimination method, for two other behaviour patterns.

Figure 9 shows two other behaviour patterns that are stopped at the start of their time interval of interest, after which all by the unique edges elimination method-switchable loops are switched off. At the end of the time interval the loops are switched on again and the changed behaviour pattern is plotted. Comparing this graph with the graph in which also consecutive edge-switchable loops are switched off, shows that this last method has some problems. We have either now and then implemented it wrongly or the method is not perfect as it was proposed by Huang et al. (2010).

In these graphs the lines belonging to changed experiment settings that induce floating-point errors have been left out. This is an important point for further research. The reason for these floating-point errors is mostly a division by zero in the extra model structure build to implement the unique consecutive edge method. This problem can be alleviated but needs extra research.

From the results we can conclude that in most intervals of interest a small number of loops has an important contribution in determining the behaviour type. Switching the loops off thus causes the behaviour to change (even after the interval, when the loop has already been switched on again). Speaking of one dominant loop per interval is thus not possible.

6. Concluding remarks and future work

6.1 Concluding remarks on the connection

In this paper we have researched the possibility of using FMA under *dynamic complexity* and *deep uncertainty,* thus combined with ESDMA. It is argued that the virtues of the different ways of finding relations between model structure and model behaviour, can complement every use of exploratory modelling. FMA can namely assist in: generating large numbers of plausible dynamics, exploring and analysing future developments, investigating policy effectiveness under deep uncertainty and evaluating/comparing different policies over the complete uncertainty space. The reason for this is in the importance of being able to attribute a variables' behaviour to a certain model structure responsible for that behaviour. Behaviour changes, bifurcations and drivers for (non-)desirability of behaviours can thus be investigated. Furthermore, SD models can be forced to output even more plausible futures than by only sampling over very wide uncertainty ranges.

The applicability of Ford's Loop Deactivation Method to medium to large models as analysed in this paper is severely limited by a number of computational problems. Conceptually, the method is reasonably well developed. Computation problems however play a very large role. These include the fact that model structure to switch off loops is not so easy to find, and cannot be built automatically, it must be done by hand. Besides that dividing behaviour patterns into time intervals is far more difficult than in small models with only a few stock variables.

6.2 Discussion of method deficiencies

In applying and automating Ford's Deactivation Method a number of deficiencies have been found. Some of these points of attention are just inherent to the method used, but doing more research could close other gaps.

The first problem that needs attention is the fact that not all loops could be switched off. Not all loops in the SILS have a unique structure from which a switch construction can be derived. This fact is of course only a potential problem when it is necessary to exhaustively switch off all loops, for instance when drawing conclusions about loop dominance. However, when the Loop Deactivation Method is only used to assist in the generation of plausible futures, it is not more than helpful when more and more loops can be switched off. A smaller problem that is loosely connected to this model structure problem is the non-unicity of the SILS.

Secondly, loops further away seen from the variable of interest naturally give less variance in that variable when switched off. This problem occurs more and more when analysing models of increasing size. This is also more a problem when direct conclusions are drawn on structure-behaviour relations. Furthermore, the specification of switches may be influenced by the usage of discrete elements like if-then-else and minimum/maximum functions. This could be the reason of the sudden behaviour changes observed in the graphs for the unique consecutive edge-methods.

Then there are the problems with identifying a suitable number of time intervals of ABPs to which the Loop Deactivation Method can be applied. The kind of behaviour shown by larger models (11 stocks with discrete elements in this case) is such that the ABP changes far more than in small models with only two or three stocks. This problem could be alleviated by smoothing the behaviour, in other words only looking at *larger* changes in the first and second derivatives.

Lastly, the way in which dominance is defined can be strengthened, for instance by using distance metrics as also implemented in behaviour clustering. Another point is the fact that a larger number of loops is important in determining in the model behaviour in the interval of interest (and thereafter). So it is maybe better to speak of relative contribution of a certain loop in determining the model behaviour in a certain time interval.

It has been observed that large-model behaviour is far more jumpy than the nice and even exponential, linear and logarithmic behaviour observed in models where FMA-methods are mainly tested on. This results in a very large number of time intervals of which in each interval all loops have to be switched off. Whereas a number of intervals in the order of magnitude of a few up to ten per analysed behaviour pattern is feasible.

6.3 Directions for future work

Future research must strengthen the justification of connecting Formal Model Analysis and Exploratory System Dynamics to confront *dynamically complex* and *deeply uncertain* problems. This justification can come from showing that strong results on structure-behaviour relations can be found by applying the methods illustrated in this paper to more models and more problems. Further investigation should strengthen the purposes of connecting the methods and also find further uses.

Furthermore, Ford's Loop Deactivation Method must be altered in a number of ways to be able to deal with models of medium to large size. The deficiencies identified in the last subsection amount to a limited usability of the method when analysing a larger model.

Lastly, the other main stream of Formal Analysis techniques, Eigenvalue Elasticity Analysis, should also be incorporated into an exploratory modelling study. Implementing and automating this method is however more difficult than the implementation of Ford's behavioural method. For this method is less understandable for non-trained mathematicians and thus for mainstream System Dynamicists. The mathematical foundation of Eigenvalue-based methods is however much stronger, it could thus aide in connecting the field of System Dynamics with its own roots in differential equations and dynamical systems, a field of hundreds of years of mathematical development.

References

Balakrishnan, V.K. (1997). *Graph theory, Schaum's outlines.* McGraw-Hill: New York.

Bankes, S. (1993). Exploratory modeling for policy analysis. *Operations Research.* **41**(3), 435-449.

Ford, D. N. (1999). A behavioral approach to feedback loop dominance analysis. *System Dynamics Review.* **15** (1999), 3-36.

Forrester, J.W. (1968). *Principles of systems.* Second preliminary edition, fifth printing, October 1971. Wright-Allen Press, Inc.: Cambridge, MA, USA.

Forrester, N. (1983). Eigenvalue analysis of dominant feedback loops. *Proceedings of the 1st International Conference of the System Dynamics Society.* Chestnut Hill, MA, USA.

Friedman, J.H., N.I. Fisher (1999). Bump hunting in high-dimensional data. *Statistics and Computing.* **9** (1999), 123-143.

Hagberg, A.A., D.A. Schult, P.J. Swart (2008). Exploring network structure, dynamics, and function using NetworkX. *Proceedings of the 7th Python in Science Conference (SciPy2008).* G. Varoquaux, T. Vaught & J. Millman (eds.), Pasadena, CA, USA, August 2008. 11-15.

Huang, J., E. Howley, J. Duggan (2010). An extension of Loop Deactivation in the behavioural method. *Proceedings of the 28th International Conference of the System Dynamics Society*. Seoul, Korea.

Kampmann, C.E. (1996). Feedback loop gains and system behavior (unpublished manuscript). Summarized in *Proceedings of the 14th International Conference of the System Dynamics Society.* Cambridge, MA, USA.

Keogh, E., S. Kasetty (2003). On the need for time series data mining benchmarks: A survey and empirical demonstration. *Data Mining and Knowledge Discovery.* **7** (2003), 349-371.

Kwakkel, J. & E. Pruyt (2011). Exploratory modelling and analysis, An approach for model-based foresight under deep uncertainty. *Fourth International Seville Conference on Future-Oriented Technology Analysis.* Seville, Spain, 12-13th May, 2011.

Kwakkel, J.H., W.E. Walker, V.A.W.J. Marchau (2010). Classifying and communicating uncertainties in model-based policy analysis. *International Journal of Technology, Policy and Management.* **10**, 4 (2010), 299-315.

Lempert, R., S. Popper & S. Bankes (2003). *Shaping the next one hundred years: New methods for quantitative, long-term policy analysis.* RAND report MR-1626, The Rand Pardee Center, Santa Monica, CA, USA.

Liao, T.W. (2005). Clustering of time series – a survey. *Pattern Recognition.* **38** (2005), 1857-1874.

Miller, J.H. (1998). Active Nonlinear Tests (ANTs) of complex simulation models. *Management Science.* **44** (6), 820-830.

Mojtahedzadeh, M. T., Andersen, D., and Richardson, G. P. (2004). Using digest to implement the pathway participation method for detecting influential system structure. *System Dynamics Review,* **20** (2004), 1–20.

NumPy Reference (2012). *Numpy Reference. Numpy v1.7.dev-6fd716c Manual (DRAFT).* http://docs.scipy.org/doc/numpy/reference/. Accessed on March 19th, 2012.

Oliva, R. (2004). Model structure analysis through graph theory: partition heuristics and feedback structure decomposition. *System Dynamics Review.* **20** (2004), 313-336.

Phaff, W.G. (2006). *Investigating model analysis. A critical examination of current methods for model behavioural analysis in System Dynamics.* Master thesis, Faculty of Technology, Policy and Management, Delft University of Technology. June 2006.

Phaff, W.G. (2008). Generalised Loop Deactivation Method. An extension of Ford's behavioural approach to Loop Dominance Analysis and its applications. *Proceedings of the 26th International Conference of the System Dynamics Society.* Athens, Greece.

Phaff, W.G., J.H. Slinger, B. Güneralp, C.E. van Daalen (2006). Investigating model behavioural analysis. A critical examination of two methods. *Proceedings of the 24th International Conference of the System Dynamics Society.* Nijmegen, The Netherlands.

Pruyt, E. (2007). Dealing with uncertainties? Combining System Dynamics with Multiple Criteria Decision Analysis or with Exploratory Modelling. *Proceedings of the 25th International Conference of the System Dynamics Society*. Boston, MA, USA.

Pruyt, E. (2010a). Using small models for big issues: Exploratory System Dynamics Modelling and Analysis for insightful crisis management. *Proceedings of the 28th International Conference of the System Dynamics Society*. Seoul, Korea.

Pruyt, E. (2010b). Scarcity of minerals and metals: A generic Exploratory System Dynamics model*. Proceedings of the 28th International Conference of the System Dynamics Society*. Seoul, Korea.

Pruyt, E., J. Kwakkel, G. Yücel, C. Hamarat (2011). Energy transitions towards sustainability I: A staged exploration of complexity and deep uncertainty. *Proceedings of the 29th International Conference of the System Dynamics Society*. Washington DC, USA.

Rossum, G. van (1995). *Python reference manual.* Amsterdam, The Netherlands: CWI.

SciPy Reference (2012). *SciPy – Scipy v0.10 Reference Guide (DRAFT).* http://docs.scipy.org/doc/scipy-0.10.1/reference/. Accessed on March 19th, 2012.

Senge, P. (1990). *The fifth discipline: the art and practice of the learning organization.* London: Doubleday/Currency.

Theodoridis, S. (2003). *Pattern Recognition*. In Encyclopedia of Information Systems, pp. 459-479.

Yücel, G. (2012). A novel way to measure (dis)similarity between model behaviors based on dynamic pattern features. *Proceedings of the 30th International Conference of the System Dynamics Society.* St. Gallen, Switzerland.