Integration of System Dynamics and Rule Based Reasoning Mechanism

Yi-Ming Tu
Naeyi Shiao
Institute of Information Management
National Sun Yat-Sen University
Taiwan, R.O.C.

Abstract

After providing a framework for integration of System Dynamics and Expert Systems, this paper builds theoretical bases to integrate three main features of rule based reasoning mechanism into conventional System Dynamics models. Then we start to modify the System Dynamics modeling tools to adopt the integrated features. To illustrate, we demonstrate a prototype for integrated theories above.

Introduction

System Dynamics (SD) has been successfully applied to solve many problems for years. However, the methodology itself seems to have less progress than its applications. This paper tries to broaden the modeling capability of SD through rule based reasoning mechanism in Expert Systems (ES).

First, we depict the framework of ES aid to SD ; then integrate rule based reasoning mechanism into the SD model, modify SD modeling process. To illustrate, we provide a prototype demonstrated for integrated features.

Expert Systems Aid to System Dynamics

In SD modeling process, we usually follow the four steps below:

1. Conceptualization : focusing on impotrtant elements of the real world problems, utilizing causal diagram to describe the causal feedback relationship among elements.
2. Formulation : partitioning causal diagram, determining levels, rates, auxiliary variables, and parameters by SD diagram, then preparing to compile and execute the model through SD simulation software, such as STELLA, DYNAMO.
3. Testing : Verifying and Validating the SD model.
4. Implementation : Analyzing the model according to decision scenarios.

Conventionally, the causal feedback relationship among elements must be represented in mathematical form, such as C=A+B. In addition, such relationship should be completely certain. That is, while we say C=A+B, B is always equal to A+C exactly at all cases.

Necessity for Integration

However, when the real world problems occur the following three conditions, the conventional SD methodology may not be met.

1. Some elements can not be quantified by numeric index, for example, "industry level". Usually, we deal with such question by using artificial index to quantify and thus represent this sort of elements. For example, we define closed interval [0, 1] to represent the industry level, 0 lowest and 1 highest. From that definition, however, the operation should be done with much care and the interpretation of simulation outcome is more unnatural relative to the real world problems. Therefore, we should allow the linguistic representation for such elements, like "low", "medium", and "high" for industry level.
2. The relationship among elements sometimes, but practically, may not be all certain. For instance, perhaps some cases hold, B=A+C, but in some other conditions, B=D+C. Although some SD software, like STELLA, DYNAMO, provide such IF..THEN feature, more sophisticted IF..THEN relationship can not be represented yet.

3. Furthermore, we are not often sure about the relationship among elements. Perhaps we have only 80% certain about B+A+C, and 60% about B=D+C. It should be noted that B now has two values with different certainty at the mean time. SD does not serve that also.

To provide the preceeding three features, we utilize the Expert Systems aid developed in detail at the following two sections. As a problem solving tool, ES provides flexible knowledge representation, including numeric and linguistic forms, as well as IF..THEN structure specified in rule based ES. Moreover, the reasoning mechanism of ES allows the inexact conditions, as known as inexact reasoning. Therefore, we first introduce the integration framework, and then describe how to integrate SD and ES in detail.

Framework and References of Expert Systems Aid to System Dynamics

From Figure 1 to Figure 4, we show the integration framework and relative references of ES aid to SD. Note that the framework is based on SD modeling process introduced above. Our efforts will focus on the Fig.2, formulation stage in the following sections. In that, 2, 3, 4, 5 items will be discussed more detailly.

Conceptualization	
Expert Systems Aids	Relative References
1. Extract knowledge about problems	(Klenhans,& ....,1986, 1989) (Metern Peter,1986) (Camara, & ...., 1986) (Young, 1985)
2. Natural language interface	
3. Provide models to users after request	(Banerjee, & ....,1990)
4. Use expertise to analyze patterns	(Merten, 1989)

Fig. 1 Integrated reference of conceptualization

Formulation	
Expert Systems Aids	Relative References
1. Convert causal diagram to SD flow chart	(Gonzalez,&....,1986) (Uhran, &.... , 1988) (Pe, &...., 1990)
2. Represent relations in the rule form	(Metern, 1986,87,89)
3. Use fuzzy theory	(Mlling, 1988)
4. Multidimensional simulation	(Camara, &...., 1990)
5. Allow uncertainty	(Milling, 1988)

Fig.2 Formulation

Testing	
Expert Systems Aids	Relative References
1. Use expertise to validation	(Sue Shau-Yi, 1987)

Fig.3 Testing

Implementation	
Expert Systems Aids	Relative References
1. Use expertise to explain outcomes	(Sue Shau-Yi, 1987)

Fig.4 Implrmentation

Integration of System Dynamics and Rule Based Reasoning Mechanism

Rule based reasoning mechanism plays an important role in rule based ES. It represents knowledge in rule form, such as IF..THEN, and deals with uncertainty through certainty factor or fuzzy reasoning features. Section 3.1 to 3.3 will prove how to integrate three properties of rule based reasoning into SD models.

Rule Based Reasoning

In addition to conventional SD knowledge representation, direct mathematical form, we now add the rule form to represent the relationship of elements. In perspective of Programming Language, we just utilize the logic operator AND, OR, NOT, and decisive construct IF..THEN .. ELSE, and nested-if features.

In fact, DYNAMO has already provided such IF..THEN feature through its CLIP function. For example, A=CLIP(P, 1.5P, R, S) means IF R>=S THEN A=P, ELSE A=1,5P. Such CLIP function has been used in [Metern Peter, 1986] to describe the structural change in SD models. However, the conventional CLIP function can not describe the following sophisticated nested-ifs.

```
IF R>=S THEN A=P,
ELSE IF U>V THEN A=P+Q,
            ELSE A=P+R.
```

Note that A=B+C is just another form of IF *<true>* THEN A=B+C, which means the generality of rule form representation.

Multidimensional Simulation

Multidimentional simulation [Camara, A., Antunes, P., Pinheiro, M. and Seixas, M., 1987] means in the simulation process, the number of elements' data type is more than only one, numeric type. That is, in addition to conventional numeric data type, it is allowed to have linguistic (or even others) data type. In this paper, for those elements difficult to be quantified in nature, we can represent and measure them in linguistic data type.

In SD models, we say "human's memory is effected by his age". Owing to the difficulty and unnature to quantify the "memory" and "age", we let memory be represented by "good", "medium", and "poor", and also quantify age by " young", "medium", and "old". Naturally, we say that the larger the age, the worse the memory. Thus coupling with the rule form, we say

```
IF age="old" THEN memory="poor",
ELSE IF age="medium" THEN memory="medium",
ELSE IF age="young" THEN memory="good".
```

That is so intutive that we do not need to quantify "age" and "memory" with care any more.

Dealing with Uncertatinty

ES usually utilizes certainty factor or fuzzy reasoning techniques to solve uncertainty. Here we choose certainty factor [Guru Reference Manual, 1985] as our integration solution into SD models.

First, we define [0, 1] closed interval as our certainty boundary. When we say A=2B+C CF 80 means that we have 80% certainty about A=2B+C. "CF" is abbreviation of "Certainty Factor."

Combining with IF..THEN structure, we say

```
IF A>B THEN C=2P+Q  CF  80,
       ELSE C=2P+0.5Q  CF  60.
```

Notice that the sum of these two certainty factors is not equal to 100. That means certainty factor is a kind of possibility, not probabilty.

As above example, C has only one value 2P+Q or 2P+0.5Q. Another practical example, however, is as follows.

```
IF A> B THEN C=2P+Q  CF  80,
          ELSE C=2P+0.5Q  CF  80,
               C=2P+0.5R  CF  60.
```

Here, if A<B, C gets two values, 2P+0.5Q and 2P+0,5R, at a time. Such variable C is known as a fuzzy variable.

If there is a fuzzy variable in a SD model, it will be propagated to other variables through operation, and thus slow down the simulation performance. Therefore, while practical implementation, we need an effective mechanism to control the expansion of fuzziness. [Negoita, Constantin V., 1985] [Guru Reference Manual, 1985]

Feature Summary after Integration

As far, we integrate three properties of rule based reasoning mechanism into SD models.

1. Represent the relationship among elements by IF..THEN..ELSE structure.
2. Add linguistic data type in addition to conventional numeric form.
3. Utilize certainty factor to deal with uncertainty, and thus allow the usage of fuzzy variable.

After such integration, we can make some breakthrough to solve quantification and uncertainty problems in conventional SD models.

Modification of System Dynamics Modeling Tools

Building theoretical bases as above, we now start to modify the modeling tools of conventional SD in order to adopt the intregration features into the whole modeling process.

Modifying Causal Diagram

According to the following Fig.5, there are six catagories to modify the causal diagram.

		IF .. THEN		
		No	Yes	
			structure changed	structure not changed
Fuzzy	Yes	1	2	3
	No	4	5	6

Fig.5 Modified classification of cansal diagram

1. If there is no IF..THEN and fuzzy variable, every symbol remains the same as the conventional causal diagram.
2. If there occurs IF..THEN, but model structure is not changed, and no fuzzy variable, as below,

IF C<U THEN A=2D+E,

ELSE A=2D+1.5E.

We may represent it by the following symbols in Fig.6.

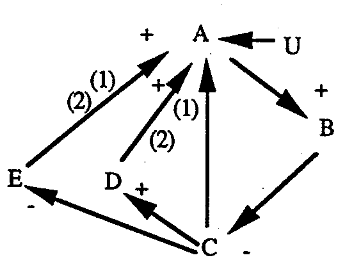

Fig.6

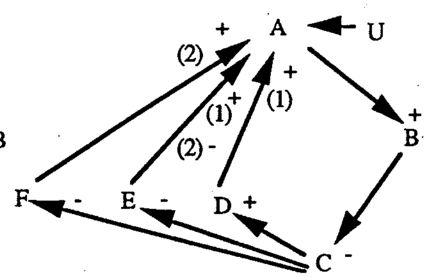

Fig.7

It means that A has two possible values under two different conditions which are both positively effected by D and E. Since C and U occur in IF part, there are no positive or negative symbols to A. The representation is analogous when the number of possible conditions is more than two.

3. If IF..THEN occurs, and structure is changed, but no fuzzy variable, for instance,

IF U>D THEN A=2D+E,

ELSE A=2F-E.

Symbols will be as Fig.7 above.

That means in one case, A is positively effected by D and E, but in the other, A is positively effected by F and negatively by E. The representation is analogous when the number of possible conditions is more than two.

4. There is no IF.. THEN, but fuzzy variable exists. For example,

A=2D+E CF 60
A=2D+1.5E CF 80
can be represented by the following Fig.8.

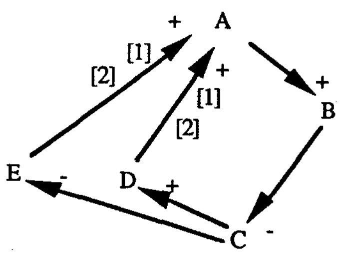

Fig.8

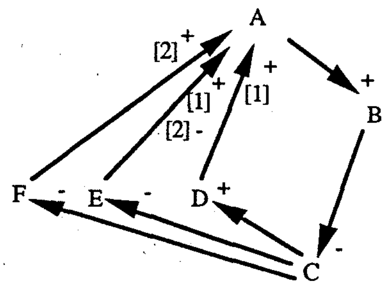

Fig.9

Here, A has two values meanwhile, both positively effected by D and E. Or as below,
A=2D+E CF 60
A=2F-E CF 80
can be depicted as Fig.9.
Here A also has two values meanwhile, but one positively effected by D and E, the other positively effected by F and negatively by E. The representation is analogous when the number of possible conditions is more than two.
5. IF..THEN and fuzzy variables occur, but no structural change, as the following.
IF B>U THEN A=2D+E,

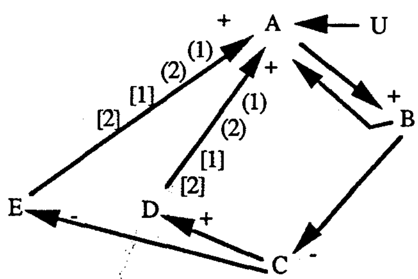

Fig.10

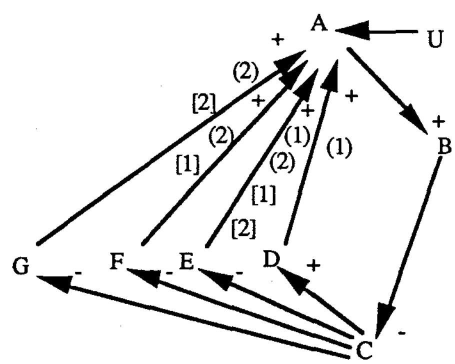

Fig.11

ELSE A=2D+1.5E CF 60,
A=2D+1.7E CF 80.
It can be pictured as Fig.10 below.

(1) and (2) say that there are two conditions by IF..THEN, and [1] and [2] represent there are two values at a time in both (1) and (2) conditions. The representation is analogous when the number of possible conditions is more than two.
6. IF..THEN, fuzzy variables, and structural change all occur. For example,
IF U="medium" THEN A=2D+E,
ELSE A=2F+E CF 60,
A=2G+E CF 80.
may be Fig.11.

It means that in the first IF..THEN case, A is positively effected by D and E, but in the other, A has two values at a time, one value positively effected by D and E, another also positively effected by E and G. Note that U is a linguistic variable, and is located in IF part, so there are no positive or negative signs from U to A. The representation is analogous when the number of possible conditions is more than two.

Adding Element Specification Table

After integrating rule based reasoning mechanism, the causal diagram contains so much modeling information that we need another tool, as data dictionary in information systems analysis stage, to carry information about each element in a SD model.

Therefore, we add a tool named "element specification table" to describe the followings of each element.

1. Identity number : specification of each element.
2. Name : full name and abbreviation used in the causal diagram.
3. Range : data type and its range for possible values if it has.
4. Description : about element's relationship relative to others in natural language or standrized pseudocode
5. Minimal confidence degree : degree to represent the minimal acceptable certainty factor respective to each element. That is, if the element's certainty factor is smaller than this minimal confidence degree, we cancel the value in such fuzzy variable. This item is available to control the expansion of fuzzy variables.
6. Note : other information available to simulation.

The format of element specification table could be as Fig.12.

ID.	
Name	
Range	
Description	
Minimal Confidence Degree	
Note	

Fig.12 Format of Element Specification Table

Modifying System Dynamics Diagram

In SD Diagram, we choose levels, rates, physical flows, information flows, sources, sinks, auxiliary variables, parameters, and necessary delays in the model.

As our modification in causal diagram, the same will be used on SD diagram.

Considerations of Coding Tools

It's time for us to consider the coding tools to implement SD models. Here, we discuss three classes for integrated SD models theoretically.

1. Based on SD software, we may couple with other software tools to fulfill rule based reasoning features. The followings are four coupling choices.

-- No coupling : for example, we use DYNAMO to complete the whole integrated features. [Metern Peter, P., 1986] utilizes CLIP and SWITCH function to implement simple IF..THEN feature, but loss of uncertainty.
-- Coupling with 3GL : to illustarte, we may implement rule based reasoning in FORTRAN, as well as DYNAMO interface funtions.
-- Coupling with ES language : using PROLOG or LISP for rule based reasoning, we make DYNAMO as interface with them.
-- Coupling with package software : for instance, use LOTUS 1-2-3 and DYNAMO for integration.

2. Based on ES shell or ES language, we may implement rule based reasoning easily. However, many SD features may not be achieved by ES tools. This class of technique may take much effort. There are two possible couplings.
-- Implement only by ES shell or ES language, for example, GURU [Uhran, J., Ghiaseddin, N. and Bualuan, R., 1988], and PROLOG [Camara, A., Antunes, P., Pinheiro, M. and Seixas, M., 1987].
-- Coupling with 3GL : for example, C and PROLOG for SD features and rule based reasoning respectively.

3. New integrated software, we suggest, for implementation. We can fulfill the preceeding integrated features only by another new software tool. Nevertheless, it is unnecessary to design in the whole new source codes. Perhaps we may utilize the framework of SD software, say STELLA, and then just add integrated features and accompanied user interfaces.

Prototype for Integration

In this section, we demonstarte an example based on the theories in the preceeding two sections. This example origins from the chapter 9 model of [Peterson, S., Richmond, B., and Vescuso, P., 1987]. we modify it to show the integrated features described above. The followings are consistent with the four-stage SD modeling process. The causal diagram shows as the following Fig.13.

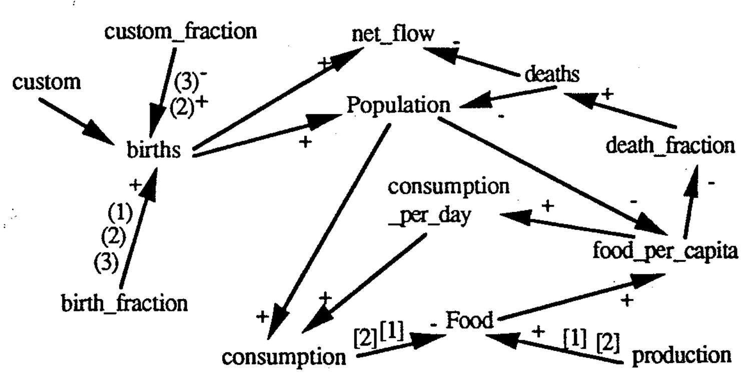

Fig.13 Intergrate causal diagram of prototype

Note that "births" is determined under three different conditions. One is positively effected by "birth_fraction"; another is positively effected by "birth_fraction" and "custom_fraction"; the other is positively effected by "birth_fraction" and negatively effected by "custom_fraction". We may see "custom" appears at IF part. Moreover, "Food" is here a fuzzy variable with two values meanwhile. Both are positively effected by "production" and negatively by "consumption".

After the causal diagram, we should complete element specification table for each element. Here we provide two instances, "Food" - Fig. 14, "births" - Fig. 15. Then the SD diagram is as Fig. 16. "Custom" is a linguistic variable. We now represent the relationship more naturally through IF..THEN mechanism. Fuzzy variable "Food" carries two values with 60 and 80 certainty factor respectively.

Coded and compiled by TURBO C version 2.0, we show the simulation outcomes as Fig. 17.

ID.	1
Name	Food
Range	Numerical type, Fuzzy variable
Description	Food=Food+DT*(production-consumption)*0.95 CF 80 Food=Food+DT*(production-consumption) CF 80
Minimal Confidence Degree	50
Note	

Fig.14

ID.	3
Name	births (birth rate)
Range	Numerical type
Description	IF custom="normal birth rate" THEN births = Population*birth_fraction ELSE IF custom="high birth rate" THEN births = Population*(birth_fraction+custom_fraction) ELSE IF custom="low birth rate" THEN births = Population*(birth_fraction-custom_fraction)
Minimal Confidence Degree	50
Note	

Fig.15

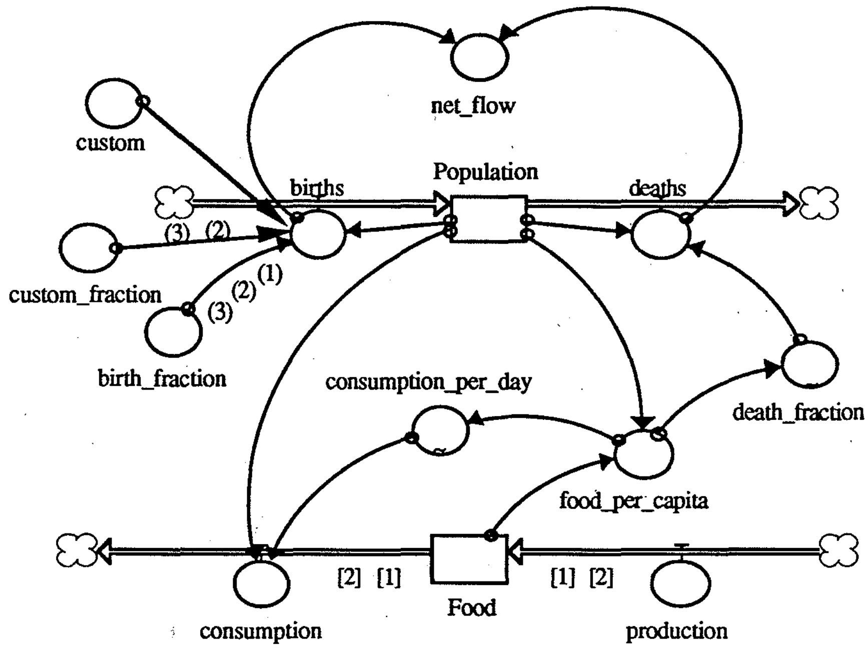

Fig.16 System Dynamics Diagram of Prototyping Integrated

Note that in "food_per_capita", the value of fuzzy variable "Food" is the sum average of multiplication of its values and respective certainty factor weight. This is an effective method to control the expansion of fuzzy variables while simulation.

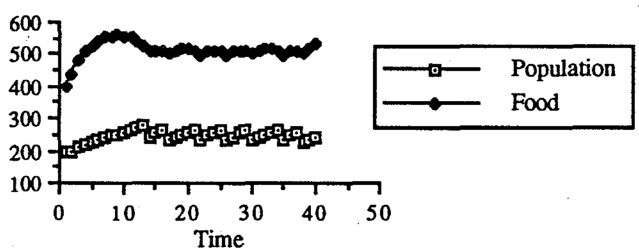

Fig.17 Integrated Implementation of Prototyping

Conclusion

Integrating IF..THEN representation, linguistic data type, and certainty factor into conventional SD models, we make SD more powerful to solve practical problems.

As future research, we will first accent the development of the new integration software. In addition, various and wide applications of integrated features and detailed discussion about implemention techniques, such as inexact reasoning in SD environment, will be as much our research interest.

References

Banerjee, Snehamay, and Basu, Amit. 1990. A Knowledge-Based Framework for Selecting Management Science Models. IEEE Transactions on Computers: 484-493.

Camara, Antonio, and Antunes, Paula. 1986. A New Approach to Environmental Impact Assessment. The Conference of The International System Dynamics Society: 1007-1018.

Camara, A., Antunes, P., Pinheiro, M. and Seixas, M. 1987. Linguistic Dynamic Simulation - A New Approach. Simulation, Vol. 49. 5, 208-212.

Camara, A., Ferreria, F. and Seixas, M. 1990. Exploring IDEAS : A Multidimensional Dynamic Simulation Approach. The Conference of The International System Dynamics Society: 181-189.

Gonzalez, J. and Fernandez, G. 1986. EASDM : An Expert Aid for System Dynamics Modeling. The Conference of The International System Dynamics Society: 1019-1027.

Guru Reference Manual. 1985. vol. 1 :MDBS, Inc.

Kleinhans, Andreas M. 1986. BAMBOO - A Behaviroal Analysis Expert System for System Dynamics Models. The Conference of The International System Dynamics Society: 1039-1047.

Kleinhans, Andreas M. 1989. Knowledge-Based Modeling. The Conference of The International System Dynamics Society: 527-534.

Metern Peter, P. 1986. Know-How Transfer by Multinational Corporation in Developing Countries - A System Dynamics Model with Spiral Loops. The Conference of The International System Dynamics Society: 795-849.

Merten, Peter P. 1987. Portfolio Simulation : A Tool to Support Strategic Management. System Dynamics Review, Vol. 3. 2, 81-101.

Merten, Peter P. 1989. Loop-Based Strategic Decision Support Systems - Theory and Application. The Conference of The International System Dynamics Society: 144-156.

Milling, Peter P. 1988. Subjective Knowledge Bases in Corporate Policy Making. The Conference of The International System Dynamics Society: 272-281.

Negoita, Constantin V. 1985. Expert Systems And Fuzzy Systems. The Menjamin / Cummings,Inc.

Pei, W., Liu, S. and Wang, D. 1990. A Decision Support System for System Dynamics Modeling. The Conference of The International System Dynamics Society: 841-851.

Peterson, S., Richmond, B., and Vescuso, P. 1987. An Academic User's Guide to STELLA. High Performance Systems, Inc.

Senn, James A. 1989. Analysis And Design of Information Systems. McGraw Hill, Inc.

Sue, Shau-Yi. 1987. The Support Tools for The Outcome of System Dynamics Models. The Master Dissertation of NCTU. Taiwan.

Uhran, J., Ghiaseddin, N. and Bualuan, R. 1988. Expert System Dynamics Modeling with GURU. The Conference of The International System Dynamics Society: 404-418.

Young, David F. 1985. KBSIM : A Knowledge-Based Tool And Its Use in Model Processing. The Conference of The International System Dynamics Society: 1070-1080.

Zimmermann, Hans J. 1987. Fuzzy Sets, Decision Making And Expert Systems, Kluwer Academic Publisher. Boston.