Modelling Environmental Impact with a Goal and Causal Relation Integrated Approach

He Zhang, Lin Liu†

School of Software, Tsinghua University, Beijing

Linliu@tsinghua.edu.cn

Abstract: Due to the complexity and cross-disciplinary nature of environmental and climate change related systems, collaborative modelling and cross-disciplinary knowledge sharing is unavoidable. Environmental impact modelling encompasses graphical and numerical representations, synergizes conceptual frameworks with quantitative annotation and computation. As a consequence, a collaborative knowledge sharing and acquisition platform is in dire need to support system analysis and holistic decision making in such a setting. This paper proposes modelling environmental impacts using a goal and causal relation integrated approach. In particular, top-level goals of stakeholders are modeled and refined into concrete parameters and causal relationships. To facilitate collaborative conceptual modelling and model reuse, the paper also provides a set of formalized modelling supports and a six-step collaborative modelling process. At the end of the paper, a running example modelling the relationships between the electronic vehicle market share and the regional GHG emissions is used to illustrate the proposed collaborative conceptual modelling approach.

1. INTRODUCTION

Nowadays, more and more enterprises and individuals begin to pay attention to the negative environmental impacts they have caused. Many of them tend to use models to understand the environmental outcome of their behavior and actions. People use different conceptual models to explain and analyze knowledge in various domains. Complicated social-environmental interactive systems usually cross several disciplines and are hard to model. Therefore, building cross-disciplinary models for understanding the correlations between different factors requires conceptual model integration and cross reference in various domains. To achieve cross-disciplinary model integration and collaborative modelling, there are two major challenges to tackle: One is the lack of integrated knowledge modelling language, and the other is the lack of a group model building and refinement process.

In this paper, we extend and combine the causal loop diagram in system dynamics modelling with the goal-oriented modelling in system requirements engineering as our

basic integrated knowledge modelling language. In social and environmental science, system dynamics modelling is used to understand the behavior of complex systems over time (Forrester 1961). Some of the cross-disciplinary models such as some energy-economy-climate models (Fiddaman 1997, C-Roads 2009) are built using this method. To illustrate the causal relationships and the feedback loops between the changes of variables, causal loop diagram is introduced as a major conceptual modelling method in system dynamics modelling (Sterman 2000). Although the diagram is capable of explaining the interactions between factors on the detail level, it usually cannot provide a clear holistic view of the target system. On the other hand, in system requirements engineering, goal-oriented modelling is a commonly used top-down approach to explain the system under development (Van Lamsweerde 2001). The goals used to describe the systems cover different types of concerns, and could be decomposed into sub-goals which represent the sub-systems of the original system. Therefore, goal-oriented modelling could help the user to obtain a top-down holistic view of the target system. Based on the integrated knowledge modelling language, we also provide three kinds of basic cross-disciplinary collaborative modelling support and a six-step group modelling process. At the end of the paper, a modelling example of the relationship between electric vehicle market share and regional GHG emissions is given to illustrate the proposed cross-disciplinary collaborative modelling method.

In the following sections: Section 2 describes the basic data structure and operations for causal relation conceptual models and model repository. Section 3 discusses the modelling supports provided by the proposed approach. Section 4 introduces a collaborative modelling process with a running example. Section 5 discusses related work and summarizes the paper.

2.   Goal and Causal Relation Integrated Conceptual Model

In this section, we first introduce a causal loop diagram structure used to represent the qualitative conceptual models. Beyond this causal conceptual model structure, a hierarchical domain goal refinement structure, called Domain-Model Tree, will be introduced for storing and indexing domain modelling knowledge. Finally, a goal driven Domain-Model Tree customization using the formal modelling task description will be discussed.

2.1   Conceptual Model Representation: Causal Loop Diagram

To build a model repository and to share ideas easily, it is important for the modellers to agree on a common vocabulary. In our approach, we use a wiki based system to manage the variables used in modelling.

As the variables being defined, a unified modelling language is needed to express the conceptual models. In order to focus on the holistic system behaviour and to make model integration easier, we adopt the basic concepts in Causal Loop Diagram (shown at the right

side of Figure 1) as the basic notation to represent the feedback relationships between variables in the System Dynamic Modelling practices. It can also be translated into stock and flow diagram for further quantitative modelling.
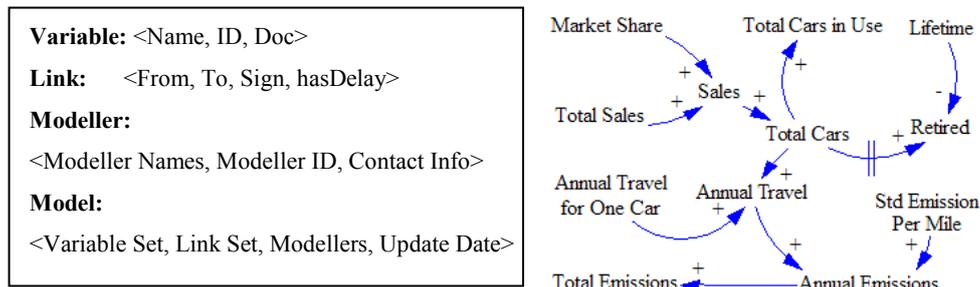
Figure 1. Causal Conceptual Model Structure

Figure 1 shows the basic data structure used to express the causal conceptual models in the system. The Variables type Name and ID attributes for identification, and Doc attribute is the wiki entry for the explanation of variables. The Links refers to the causal relationships between the variables in the model: the "From" and "To" attributes represents the starting and ending of the causal relationship Variable; the Sign attributes could take the value of "+", "-", "?", representing the "From" Variable's impact towards the "To" Variable while change; and finally, hasDelay is a Boolean attribute explaining whether the change in the "From" Variable reflect in the "To" Variable simultaneously or there is a delay involved. The Model type is defined by the set of Variables, the set of Links that used in the model. And to make the model traceable, we also introduce the Modellers and Update Date attribute to the Model type.

2.2   Goal-Based Model Repository Structure and Customization
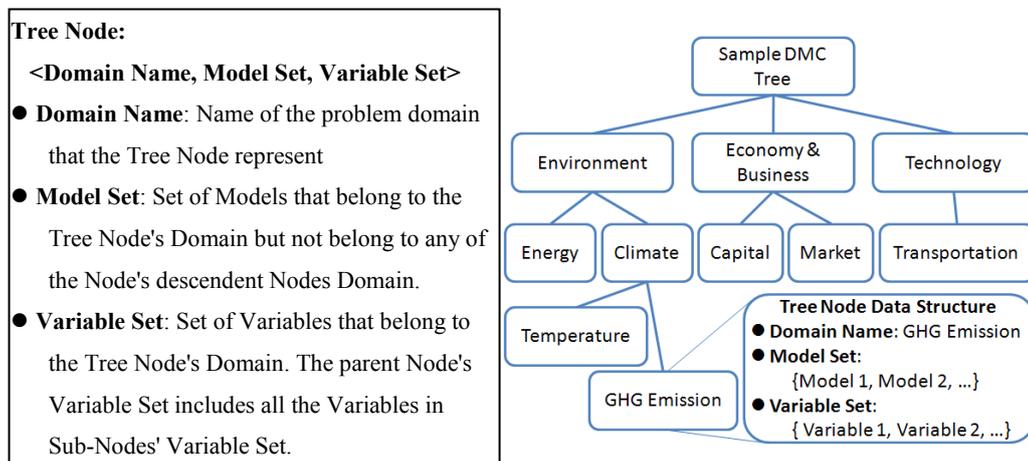
Figure 2. Domain-Model Tree Data Structure

To build the model repository, we use a hierarchical domain goal refinement structure: Domain-Model Tree (DMT), to manage the models. As Figure 2 shows, the models are being categorized according to their modelling goal and stored in the tree nodes of the DMT structure. The tree node of DMT stores not only the models addressing that problem and goal domain, but also a set of Variables within that domain. The global DMT structure will be maintained by appointed domain experts.

With the support of this global DMT, the modelling team could select a subset of problem domains in the DMT to form the system boundary for the model under development. To assist this selection operation, the system offers a DMT customization support based on their formal described Model Task.

Model Task<Input, Output, Important Causal Relations, Important Variables>

- **Input**: Set of **Variables** that serves as input in the Model to be built.

- **Output**: Set of **Variables** that serves as output in the Model to be built.

- **Important Causal Relations**: Set of important causal relations between Variables that the Modellers identified, serves as the initial causal relation set of the Model.
  The causal relations are in the form of: $A \rightarrow B(+)$; $B \rightarrow C(-, delay)$.

- **Important Variables**: Set of important **Variables** identified by the Modellers that should be considered in the Model to be built.

The customization of the DMT involves three steps:

1. The system forms initial Model Fragments using the information offered by Model Task: Input, Output and Important Variables will be treated as Variables, while the Important Causal Relations will be translated into Links in the newly formed model.

2. The system searches for the Relevant Node Set in the DMT Tree. The relevant nodes are the DMT tree nodes that share Variables in common with the initial Model Fragments.

3. Modelling team then makes modifications to the Relevant Node Set search result to form the system boundary for modelling. The customized DMT structure can be built using the selected DMT tree nodes.

Find the Relevant Tree Nodes in the DMT Tree:

Relevant Node Set =

$$\{Node\ i \mid Variable\ Set_{Domain\ Model(Node\ i)} \cap Variable\ Set_{Merge(Model\ Fragments)} \neq \emptyset\}$$

The implementation of finding Relevant Node Set can be done with a backtracking search for DMT. Because the parent tree node's Variable Set includes all the Variables in sub-nodes' Variable Set, the system could search the Tree Nodes from the root recursively: if the current tree node is relevant, then check its instant child nodes; if the current tree node is not relevant then stop checking its descendent Nodes and go back to its parent node.
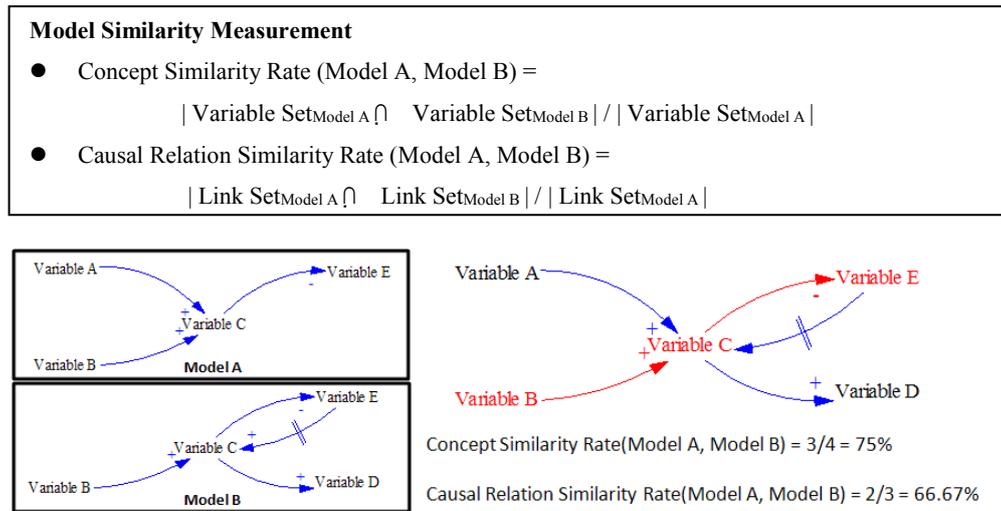
3. COLLABORATIVE MODELLING SUPPORT

Based on the model constructed above, we introduce the basic collaborative modelling support: namely Model Comparison, Model Merging & Completion and Model Decomposition.

3.1 Model Comparison

Model Comparison is used to compare the similarity between two models. To achieve this goal, we define two kinds of similarity measurements for conceptual models: Concept Similarity Rate (CSR) is used to explain the Variable similarity of the two conceptual

models; while the Causal Relation Similarity Rate (CRSR) is used to explain the Links' similarity of the models.

Model Similarity Measurement

- Concept Similarity Rate (Model A, Model B) =

$$| \text{Variable Set}_{\text{Model A}} \cap \text{Variable Set}_{\text{Model B}} | \, / \, | \text{Variable Set}_{\text{Model A}} |$$

- Causal Relation Similarity Rate (Model A, Model B) =

$$| \text{Link Set}_{\text{Model A}} \cap \text{Link Set}_{\text{Model B}} | \, / \, | \text{Link Set}_{\text{Model A}} |$$

Concept Similarity Rate(Model A, Model B) = 3/4 = 75%

Causal Relation Similarity Rate(Model A, Model B) = 2/3 = 66.67%

Figure 3. Comparison of Causal Models

Figure 3 shows an example of the model comparison. If we compare two causal models on the same subject, e.g. Model B in the lower left corner to the Model A in the upper left corner, we could identify that there are three Variables and two Links in common. Since there are three out of the four Model A Variables and two out of the three Model A Links overlapped, the CSR is 75% and CRSR is 66.7%.

3.2 Model Merging and Completion
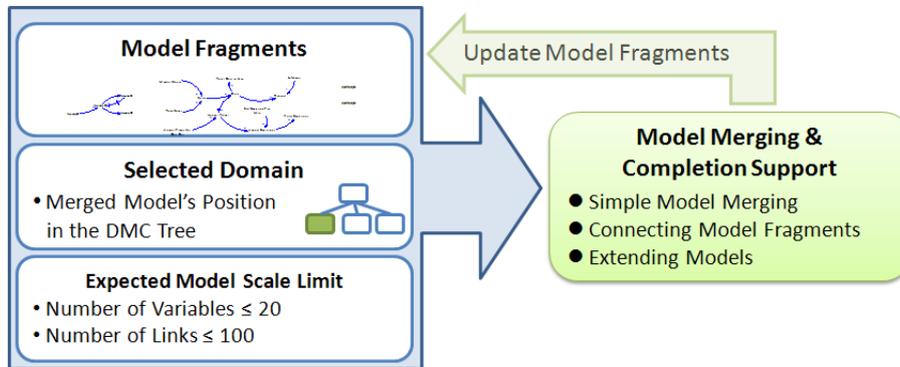
Figure 4. Cumulative Model Merging and Completion Process

As Figure 4 shows, based on the constructed model fragments, selected domain tree node of the model and the expected model scale, the Model Merging and Completion support provides three kinds of modelling facilitations for the user to merge, connect and extend the model.

3.2.1 Simple Model Merging

The idea of the simple model merging is rather straight forward, it integrates the given model fragments by merging the Variable Set and the Link Set of the fragments. Using this simple model merging operation, we have also defined Mental Model for modeller which

can be defined as the overall integrated model of all the models that the modeller has built, and the Domain model for DMT tree nodes representing the integrated model of all the models in that tree node and its descendent tree nodes. The implementation of such support can be done using a simple enumerative algorithm.

Simple Model Merging for a selected Model Set:

  Merge(*Merging Model Set*) :

    Variable Set$_{Merged}$ = { $\cup$ Variable Set$_{Modeli}$ |Model*i* $\in$ *Merging Model Set*}

    Link Set$_{Merged}$ = { $\cup$ Link Set$_{Modeli}$ |Model*i* $\in$ *Merging Model Set*}
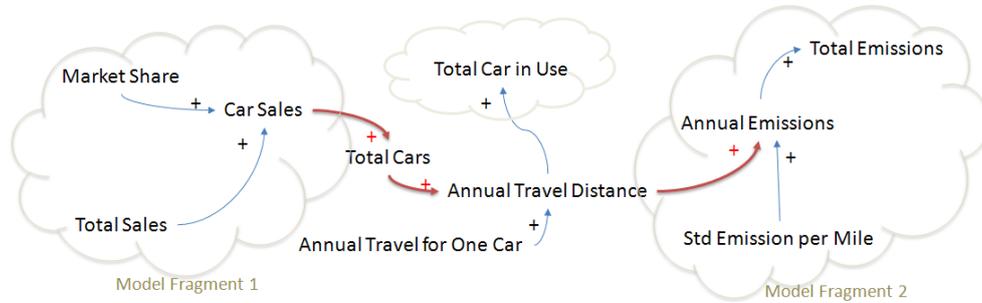
Mental Model for Modeller:

  MentalModel(*Modeller A*) = Merge({Model *i* | Modeller$_{Modeli}$ = *Modeller A*})

Domain Model for DMT Tree Node:

  DomainModel(*Node A*) =

    Merge({Model *i* | Model *i* $\in$ Model Set$_{Node\,j}$ $\wedge$ Node j $\in$ *Node A and its Descendent* })

3.2.2 Connecting Model Fragments

Figure 5 Connection Path Between Two Model Fragments

Another facility for model merging and completion is to connect the disconnected model fragments. As Figure 5 shows, a connection path between Car Sales and Annual Emission makes the two unconnected model fragments connected. Because the maximum expected size of the integrated model is defined, connecting model fragments means to find the connection paths linking the different fragments within the length limit.

Path<Link List>: Path is a list of end-to-end Links forming a path from one Variable to another.

  Length(Path) reprensents the number of links in the Path's Link List.

Connection Path Length Upper Limit =

  Min(Expected Number of Links - |Link Set$_{Merge(Model\,Fragments)}$| ,

    Expected Number of Variables - |Variable Set$_{Merge(Model\,Fragments)}$| + 1)

Finding the Connection Path i between Model Fragments:

  1. $\forall$Link j $\in$ Link List$_{Path\,i}$, Link j $\in$ Merge(Model Fragements) $\cup$ Domain Model(Node A)}

  2. Length(Path i) $\leq$ Connection Path Length Upper Limit

---

The software implementations for finding connection paths can be done with two steps:
1. Use Flood Fill Algorithm to identify the separate fragments in the Model Fragments.
2. For each Variable in the fragments, use a Depth First traverse to search for at most Connection Path Length Upper Limit steps. If the starting and ending Variables of the search don't belong to the same model fragment, then a new connection path is found.

After all the candidate connection paths being found by the system, the modellers can make their choices to accept a set of paths that makes the fragments connected, and at the same time, making the model more expressive. If the modellers would like to find the path set with the minimum total length that makes the model fragments connected, the system could use the Minimum Spanning Tree Algorithm to find the path set for them.

3.2.3 Extending Models

Figure 7. Finding Extension Causal Relationships and Feedback Loop Completion

The last kind of facilitation for Model Merging and Completion is model extension. It helps the modellers to extend the selected model fragments. As Figure 7 shows, there are two kinds of typical model extension: identifying the additional causal relationships for a single Variable, and completing the feedback loops in the model. To searching for the candidate extension feedback loops, we use a path in the model as the search criteria. In this case, finding the feedback loop means finding another path that could link end to end with the searching path and form a loop.


Finding the Candidate Extension Link Set for Variable A:

Candidate Extension Link Set =

$$\{\text{Link } i| \text{ Link } i \in \text{Link Set}_{\text{Domain Model(Node A)}} \bigwedge \text{To}_{\text{Link } i} = \text{Variable A}\}$$

Finding the Candidate Extension Feedback Loop for Path from Variable A to Variable B

Finding Path i that:

1. Path i's first Link in the Link List: Link 1, $\text{From}_{\text{Link 1}}$ = Variable B
2. Path i's last Link in the Link List : Link n, $\text{To}_{\text{Link n}}$ = Variable A
3. ∀Link j ∈ Link List$_{\text{Path i}}$, Link j ∈ Domain Model(Node A)

Finding extension link set can be implemented by using an enumerative algorithm to check the **To** attribute of all the Links inside Domain Model(Node A)'s Link Set. While to find the extension feedback loop for a search path could be done by using a Deep First Search to search Domain Model(Node A)'s Link Set for Paths from Variable B to Variable A.

3.3 Model Decomposition
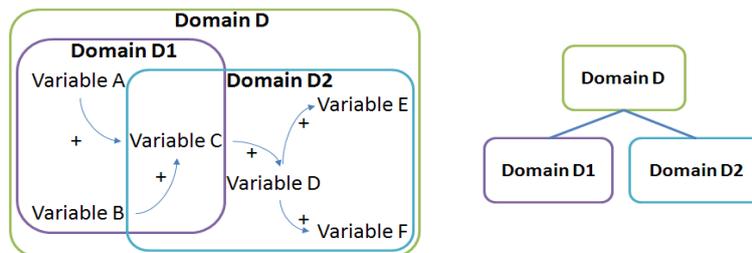
Figure 8. Model Decomposition Example

Cross Domain Link Set for relevant model Variables:

Link Set$_{Cross(Model\ m,\ Node\ A)}$ = {Link i| Link i $\in$ Link Set$_{Model\ m}$ $\wedge$ Link i $\notin$ Link Set$_{Domain\ Model(Node\ A)}$

$\quad\quad\wedge$ (From$_{Link\ i}$ $\in$ Variable Set$_{Model\ m}$ $\vee$ To$_{Link\ i}$ $\in$ Variable Set$_{Model\ m}$)}

Relevant Model Part:

Link Set$_{Relevant}$ = (Link Set$_{Model\ m}$ $\cap$ Link Set$_{Domain\ Model(Node\ A)}$) $\cup$ Link Set$_{Cross(Model\ m,\ Node\ A)}$

Variable Set$_{Relevant}$ = (Variable Set$_{Model\ m}$ $\cap$ Variable Set$_{Domain\ Model(Node\ A)}$) $\cup$

$\quad\quad$ {From$_{Link\ i}$| Link i $\in$ Link Set$_{Cross(Model\ m,\ Node\ A)}$} $\cup$ {To$_{Link\ i}$| Link i $\in$ Link Set$_{Cross(Model\ m,\ Node\ A)}$}

As Figure 8 shows, Model decomposition supports the modellers to break a large model into smaller pieces according to the Customized DMT structure. Given Model m and DMT Node A in the Domain-Model Tree, the system could identify the variables in Model m that belong to the Variable Set of Node A. By combining these variables and the links linking to these variables in Model m, a sub-model could be elicited from Model m for the domain represented by Node A. With further modification and documentation, the elicited sub-model could become a standalone model to be added to the DMT. The implementation of this could be done by simply enumerate the variables in Node A and the links in Model m.

4. COLLABORATIVE MODELLING CASE STUDY

Most of the existing collaborative modelling solutions in system dynamics focus on the sociological aspect rather than the computational aspect. In order to facilitate group modelling for non-expert modellers, and to index and reuse existing conceptual models as effective as possible, we propose a six-step-process for the collaborative conceptual modelling based on the scripts for System Dynamics Group Model Building (Luna-Reyes et.al. 2006, Richardson et.al. 1994). Throughout this six-step-process, the collaborative modelling support introduced in Section 3 can be applied to assist the modelling team to complete the modelling task. A collaborative modelling example is given in this section to illustrate this process.

4.1 Set Modelling Task

Model Task:

- **Input:** EV Market Share **Output:** CO_2 Emission to Atomsphere
- **Important Causal Relations:** EV Market Share $\rightarrow$ EV Sales (+);

  Total Vehical Sales $\rightarrow$ EV Sales (+);

  EV in Use $\rightarrow$ Electricity Consumption (+);

  Electricity Consumption $\rightarrow$ Coal Consumption (+);

  Electricity Consumption $\rightarrow$ Oil Consumption (+);

  EV Market Share $\rightarrow$ Carbon fuel Veihchal(CV) Market MarketShare (-).
- **Important Variables:** EV Market Share, EV Sales, Total EV Cars, EV in Use, Electricity Consumption, CO2 Emission to Atomsphere, CV Market Share.

In this case study, the modelling task is to build a conceptual model to analyze the potential impact of Electric Vehicle's (EV) market share on the regional CO_2 emissions to the

atmosphere in China. After a round of brain storming and knowledge elicitation with the stakeholders the initial model task can be described in the following format.

To support the group modelling, we also need a Domain-Model Tree. The Sample DMC Tree used in this case study (shown in Figure 2) covers important variables and model fragments adapted from the Climate-Economy model FREE (Fiddaman 1997), Economy and Business models from the Business Dynamics (Sterman 2000) and Environment and Technology Models from Modelling the Environment (Ford 1999).
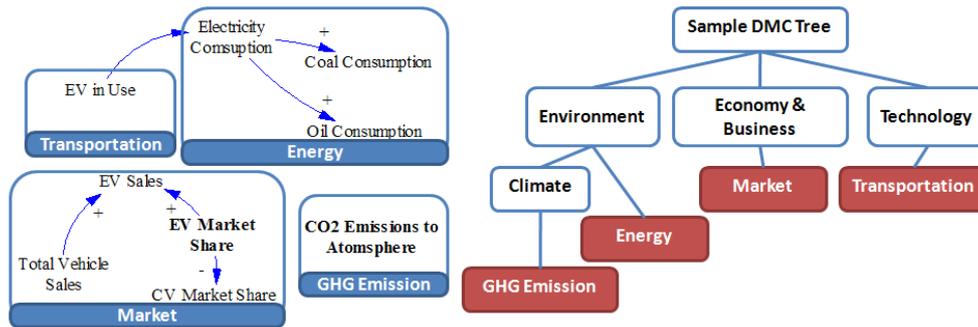
4.2 Determine System Boundary

Figure 9. Determine System Boundary

After setting the modelling task, the modelling team could work together to determine the system boundary for the model under development. Using the formal description of modelling task and the DMT customization introduced in Section 2, the system can identify the relevant domains in the DMT. At first, Variable "EV in Use", "Total Vehicle Sales", "CO2 Emission to Atmosphere" are identified belonging to Transportation, Market and GHG Emission sub-domains respectively, and Variable "Electricity Consumption", "Coal Consumption" and "Oil Consumption" are Variables belonging to Energy sub-domain. As the team member decided to add "EV Sales", "EV Market Share" and "CV Market Share" to the Variable Set of the Market sub-domain, the sub-domains and model fragments belonging to each sub-domain can be identified, shown as Figure 9.
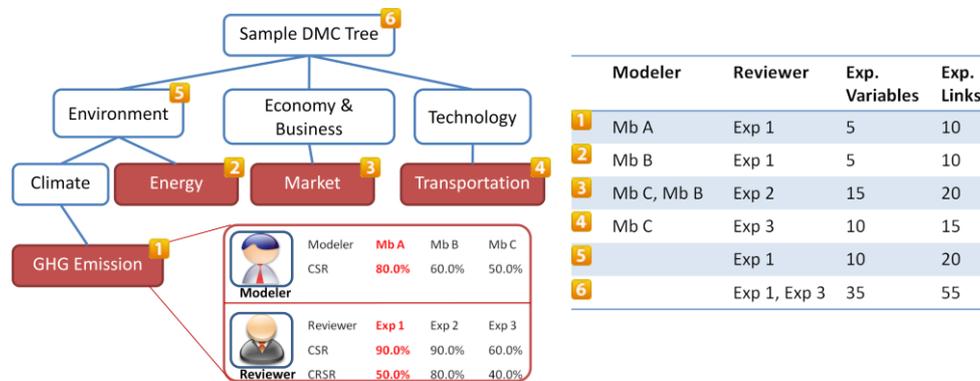
4.3 Team Work Assignment and Reviewer Recommendation

**Figure 10.** Team Work Assignment and Reviewer Recommendations

With the system boundary being identified, the system could then assist the modelling team in distributing the team work and recommending appropriate model reviewers to the team. For every leaf node and node that requires model merging in the Customized DMT, system would use the Concept Similarity Rate (CSR) and Causal Relationship Similarity Rate (CRSR) defined in Part 3.1 to make suggestions for appropriate modeler and reviewer.

The process has three steps shown as follows:
1. System compares the Team Members' Mental Model (defined in 3.2.1) with the tree node's Domain Model (defined in 3.2.1), the team members with top three CSRs will be suggested to take over the modelling work for that domain.
2. For the consulting team of experts, system compares the experts' Mental Model with the newly built model fragments, the experts with the top three CSR and lower CRSR will be suggested to be the reviewer of the model fragments of the node's domain.
3.   The modelling team could then assign the work and choose the reviewer according to the system suggestions. Detailed modelling goals for each domain and the Expected Model Scale (defined in 3.2.2) should also be made clear.

Figure 10 shows the team work assignment and reviewer recommendation results after user confirmation for the modelling case.

4.4   Building Conceptual Model in Single Domain
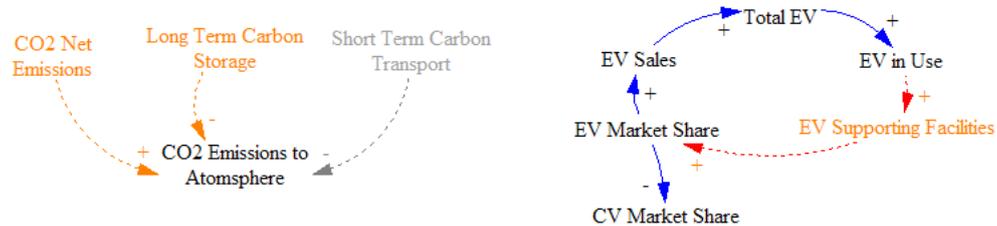
Figure 11. Model Extension Example

As the modelling work being assigned to the team members, the modellers build the conceptual model in their own domain. In this step, the Model Extension support introduced in Section 3 could be applied to help the modellers finish the modelling task in their assigned domain. As the left part of Figure 11 shows, when the modeller A deals with the GHG Emission domain, he chooses the Variable "CO2 Emissions to Atmosphere" to find the possible extended causal relations to that Variable. According to the models contained in the domain tree node, three candidate causal relations are recommended. The modeller then chooses two out of the three relations according to the context. Similarly, the right hand side of Figure 11 shows how the modeller deals with the Market domain, using the feedback loop completion support to find the candidate path to close the EV sales feedback. With this kind of support, the modeller could cumulatively build up the conceptual model for the problem in their assigned domain, and when necessary they could ask the model reviewer assigned to that domain to help review the model.

4.5 Bottom-Up Model Merging
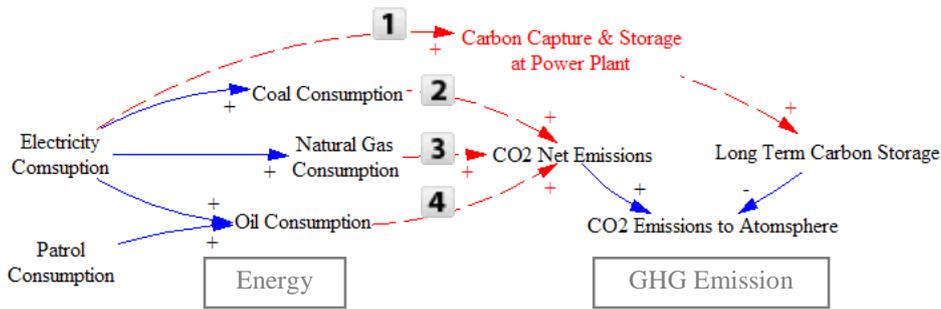
Figure 12. Bottom-Up Model Merging Example

After the modelling team finishes the models in each single domain, the team could then use the model merging support introduced in Section 3.2 to integrate the models. To merge the model more effectively, the team could use the bottom up method to merge the models, from the Customized DMT leaf nodes all the way up to the root. In this case study, the modellers should first merge the Energy and GHG Emission model fragments, and then integrated the merged model with the Market and Transportation model fragments. Figure 12 shows how the system could assist the modellers to merge the Energy and GHG Emission model fragments. The expected maximum scale for the merged model is set at 10 Variables and 20 Links. So as the modellers use the model connection support introduced in Section 3.2, the system automatically identifies four candidate paths linking the two fragments within the model scope. The modellers could choose the appropriate paths to accept and make future adjustment and extensions to the merged model.

4.6 Model Decomposition & Explanation
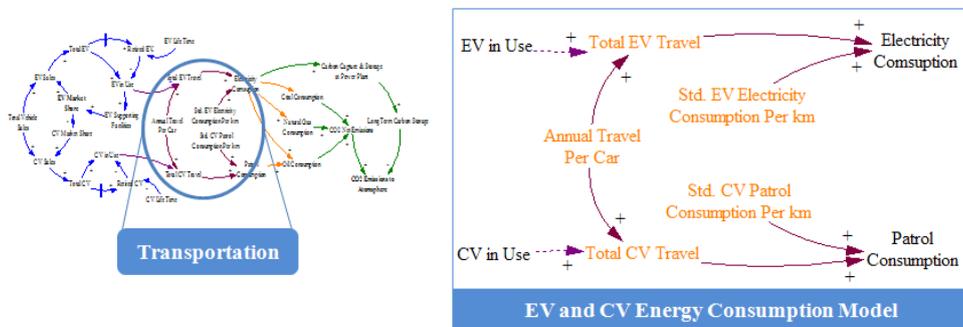
Figure 13. Model Decomposition Example

After the model is completed and evaluated, the modelling team could then use the Model Decomposition support introduced in Section 3.3 to decompose the model according to the Customized DMT structure. Take the tree node Transportation as an example, the modellers use the model decomposition support to the finished model, and the Variables in orange colour in Figure 13 are the ones belong to the tree node Transportation's domain. After identifying the domain variables, the system selects a sub set of Variables and Links from the original model according to the Model Decomposition support mentioned in Section 3.3. With some further modification and additional documentations made by the modellers, the selected part of the model can become a new standalone model to be added to the Model Set of DMT's Transportation tree node.

5.   CONCLUSION & DISCUSSION

In this paper, we propose a goal and causal relation integrated approach to model the cross-disciplinary target system's overall environmental impact. The approach extends and combines causal loop diagram in system dynamics modelling and goal-oriented modelling in system requirements engineering. This combine use is introduced to represent the correlations between various environmental and non-environmental entities and between different sub-systems. By defining and formalizing the three basic collaborative modelling supports, a six-step collaborative modelling process is proposed to facilitate cross-disciplinary modelling and model reuse. Finally, a collaborative conceptual model building example is introduced to illustrate the modelling process.

The major focus of our research is to provide a collaborative modelling platform based on the reconciliation of existing conceptual models. This platform could help people to understand the target cross-disciplinary system by modelling rather than to estimate the exact impact of the target system. Comparing with the existing cross-disciplinary ontology-based knowledge modelling approaches such as the indirectly-driven knowledge modelling approach (Pennington, et al., 2008) and ontology-driven model linking approach (Rizzoli, et al. 2008), our approach pays more attention to the collaborative modelling based on the reconciliation of existing conceptual models, rather than knowledge elicitation for building new ontology or integration of existing quantitative models. On the other hand, in contrast to the existing system dynamics group model building approaches (Akkermans 1997, Andersena 1997, Richardson 1995), the paper extends the team collaborative working scripts and processes in system dynamics modelling and provides a possible collaborative model building and management solution on the tool implementation level.
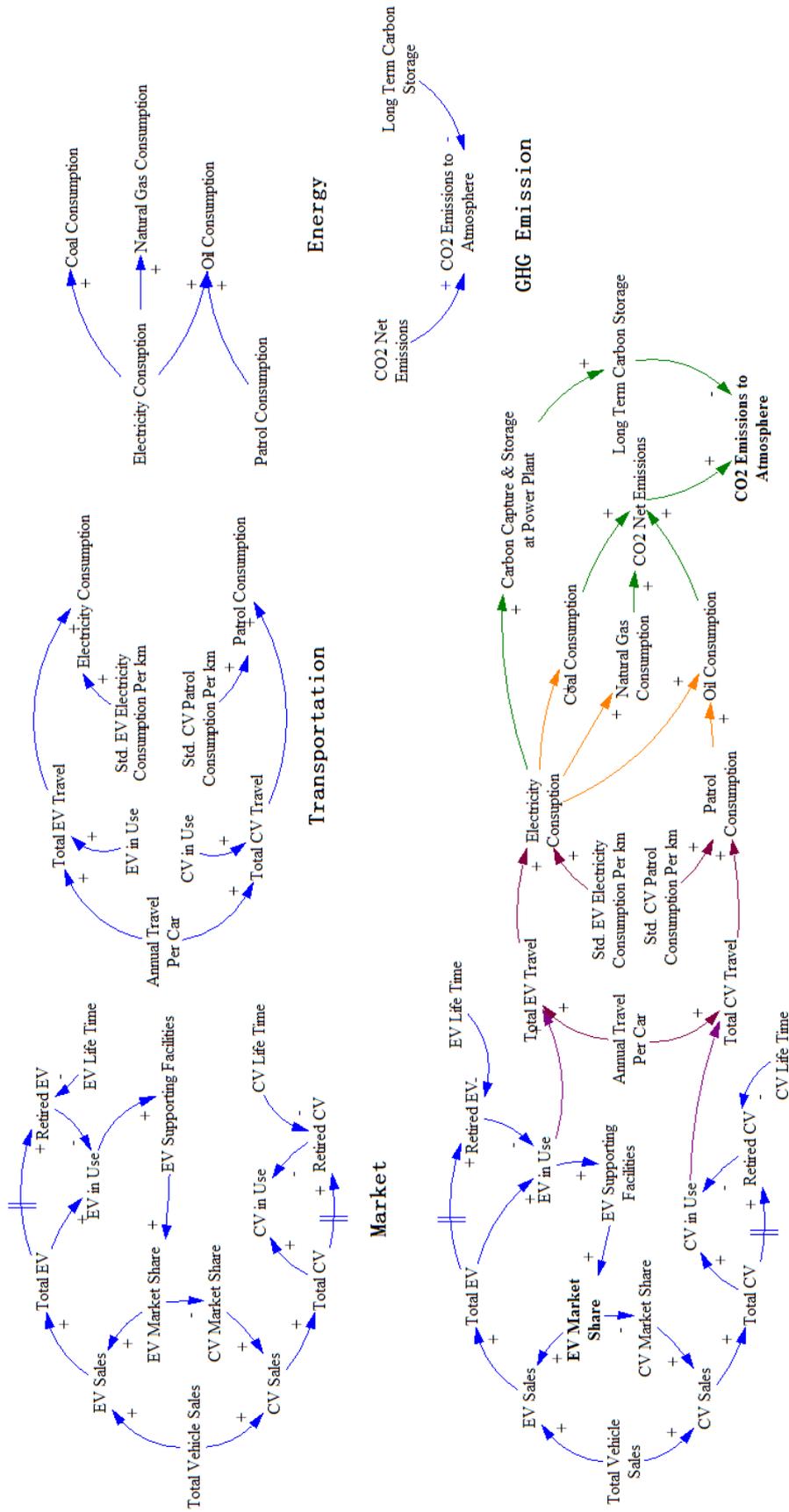
The long term goal of our research is to provide a wiki-style open modelling platform for people to share their modelling knowledge and to learn about the complicated environment related systems. However, in the short term, there are two specific goals to achieve to improve the proposed modelling approach: First is to put the semantic relations between variables (such as is-a and a-kind-of relations) into consideration. In this way, the meaning of the variables could become more explicit, and the model reuse would become more effective. The other short term goal is to extend the model description language to comply with the MIF standard in system dynamics (Myrtveit 1995), and to support stock and flow diagram in system dynamics modelling (Sterman 2000) for further quantitative modelling.

REFERENCES

Akkermans, H. and J. Vennix (1997). "Clients' opinions on group model-building: an exploratory study." System Dynamics Review 13(1): 3-31.

Andersena, D. and G. Richardsona (1997). "Scripts for group model building." System Dynamics Review 13(2): 107-129.

Argent, R. (2004). "An overview of model integration for environmental applications: components, frameworks and semantics." Environmental Modelling and Software 19(3): 219-234.

Brandmeyer, J. and H. Karimi (2000). "Coupling methodologies for environmental models." Environmental Modelling and Software 15(5): 479-488.

Burchill, G. and C. Fine (1997). "Time versus market orientation in product concept development: empirically-based theory generation." Management Science 43(4): 465-478.

C-Roads (2009), The C-Roads Simulation, http://climateinteractive.org/simulations/C-ROADS

Fiddaman, T. (1997). Feedback complexity in integrated climate-economy models, Massachusetts Institute of Technology.

Ford, D. and J. Sterman (1998). "Expert knowledge elicitation to improve formal and mental models." System Dynamics Review 14(4): 309-340.

Ford, F. (1999). Modelling the environment: an introduction to system dynamics models of environmental systems, Island Pr.

Forrester, J. and J. Wright (1961). Industrial dynamics, MIT press Cambridge, MA.

Kepler (2004). The Kepler projet. http://www.kepler-project.org

Kim, H. (2009). "In search of a mental model-like concept for group-level modelling." System Dynamics Review 25(3).

Luna-Reyes, L., I. Martinez-Moyano, et al. (2006). "Anatomy of a group model-building intervention: building dynamic theory from case study research." System Dynamics Review 22(4): 291-320.

Myrtveit, M. and A. ModellData (1995). "Models crossing the boundaries of tools." System Dynamics 95(1): 170-179.

Payne, P., E. Mendon a, et al. (2007). "Conceptual knowledge acquisition in biomedicine: A methodological review." Journal of Biomedical informatics 40(5): 582-602.

Pennington, D., et al (2008). Indirectly driven knowledge modelling in ecology. International Journal of Metadata, Semantics and Ontologies. 3, 210-225

Richardson, G. and D. Andersen (1995). "Teamwork in group model building." System Dynamics Review 11(2): 113-137.

Rizzoli, A., et. al. (2008). "Semantic links in integrated modelling frameworks." Mathematics and Computers in Simulation 78 (2008): 412-423

Sterman, J. (2000). Business dynamics: Systems thinking and modelling for a complex world with CD-ROM, Irwin/McGraw-Hill.

Van Lamsweerde, A. (2001). Goal-oriented requirements engineering: A guided tour, Toronto, Canada.

Vennixa, J. (1999). "Group model-building: tackling messy problems." System Dynamics Review 15(4): 379-401.

Villa, F., I. Athanasiadis, et al. (2009). "Modelling with knowledge: A review of emerging semantic approaches to environmental modelling." Environmental Modelling and Software 24(5): 577-587.

Visser, M. (2007). "System dynamics and group facilitation: contributions from communication theory." System Dynamics Review 23(4): 453-464.

APPENDICES