Extraction of hierarchical causal loop diagrams from dynamic models

Gergely Honti
MTA-PE Complex Systems Monitoring Research Group,
Department of Process Engineering,
University of Pannonia, POB. 158, Veszprém, Hungary

Gyula Dörgő
MTA-PE Complex Systems Monitoring Research Group
Department of Process Engineering,
University of Pannonia, POB. 158, Veszprém, Hungary

János Abonyi (janos@abonyilab.com)
MTA-PE Complex Systems Monitoring Research Group
Department of Process Engineering
University of Pannonia, POB. 158, Veszprém, Hungary

Abstract

Dynamic models (DMs) are widely used for decision support. As the complexity of these models increases, it is getting more and more difficult to understand the interactions among the variables, so the identification and the validation of DMs require tools that can extract information from these models. We present an algorithm that transforms DMs into easily understandable causal loop diagrams (CLDs) and performs modularity and centrality analysis of the extracted networks to reveal the structure of CLDs and identify the roles of the critical variables. We apply the proposed methodology for the analysis of classical models of supply chain management.

Keywords: Decision support, Network analysis, Dynamic models

Introduction

Decision support systems require models for representing problem-specific knowledge. Among these models, causal loop diagrams (CLDs) are widely used to visualize how the variables of a complex system are interrelated (Spector, 2001). CLDs have a network like representation, where the edges show the relationship types between variables represented as nodes. Building CLDs require in-depth expert knowledge, so there is a need for tools that can support the identification and the validation of these models (Richardson, 1986; Richardson, 1997; Saysel, 2006).

Dynamic models (DMs) are used when the decision requires the simulation of the model to predict the states of the system. DMs are used in wide variety of fields, like physics (Melby, 2005), economics (Gandalfo, 1971), chemistry, and medicine (Jackson, 2015). DMs are represented by Stock and Flow (SF) diagrams containing variables which describe parameters, stocks (state variables changing in time), and flows, representing the change rates. The more complex a DM is, the more precise prediction it can provide (Bar-Yam, 1997). However, as the models evolve in complexity, it is getting more and more difficult to understand the mechanisms and the interactions among the variables.

Our fundamental idea is that we transform DMs into CLDs. Since these models are too complex, we follow the rule of simplification (Doyle, 1999), we develop an algorithm that structures complex CLDs, and defines modules and automatically determines the main subjects of the system.

Based on this concept we created a novel software tool to discover the hidden structure of dynamic models by converting it into a CLD. The proposed method helps in decision making, model creation, and validation by revealing the structure of complex systems, identifying key variables.

Our methodology follows four steps: (1) transforming DMs to CLDs, (2) modularity and centrality analysis of the CLD network, (3) identification of the roles of the modules, and (4) drawing the hierarchical CLD by connecting the extracted modules.

The following sections will present the methodology; we will demonstrate the usability of our tool on two didactic and easy-to-understand supply chain based examples. Finally, we wrap it up in the conclusions section.

Methodology

(1) Transforming DMs to CLDs: The methodology starts with the transformation of system dynamic models into networks, which is not a trivial task. Figure 1 represents the stock and flow (SF) diagram of the following equation:

Equation (1):

$$x_i(t) = \int_0^t k_1 x_i(t) - k_2\big(x_i(t) + x_j(t)\big)dt$$

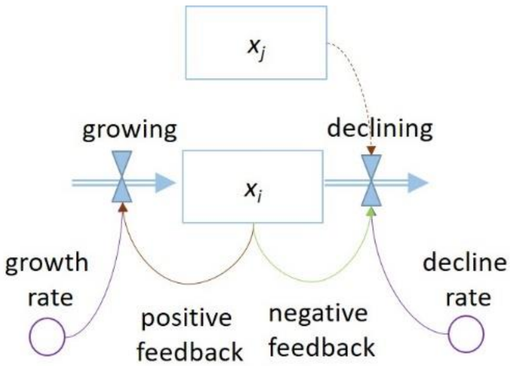

Figure 1 – Didactic illustration of the structure of a stock and flow diagram representing a differential equation

Although the SF diagram enables a direct network interpretation, it has an internal logic of dynamical systems which should also be taken into account. A significant contribution of our work is that we call the attention on the influence of the outlet flow to the stock itself, and extend the network with this additional edge highlighted with red color in Figure 2. The colors indicate the type of the node: purple represents the parameters, orange stands for the flows, while blue illustrates the stocks of the process. After this conversation step, we extract the network of state variables (stocks) by reachability analysis.

(2) Community detection and centrality analysis: Network community detection algorithms find groups of nodes that have more inner connections than with the nodes with the rest of the graph (Barabasi, 2016). We detect communities in the network with the Girvan-Newman algorithm (Girvan, 2002) and assume that the extracted communities are corresponding to the topics of the (network) model.

The centrality of the nodes expresses the structural importance of the elements of the model. Beside the analysis of the degrees of the nodes (number of connections) we also calculated the PageRank centrality that was originally developed to evaluate the importance of web pages (Page, 1999).

(3) Identification of the roles of the modules: The previously detected communities simplifies the model into interacting modules. We identified the roles of the modules based on their top-ranked nodes.

(4) Drawing the hierarchical causal loop diagram (CLD) by connecting the extracted modules: The interacting groups can be represented as a hierarchical CLD, which visualisation can provide an easily interpretable overview of the system.

In our work, we transformed dynamic models represented by Insight Maker XML files. To demonstrate the applicability of the methodology in the following we present two case studies related to supply chain modelling.

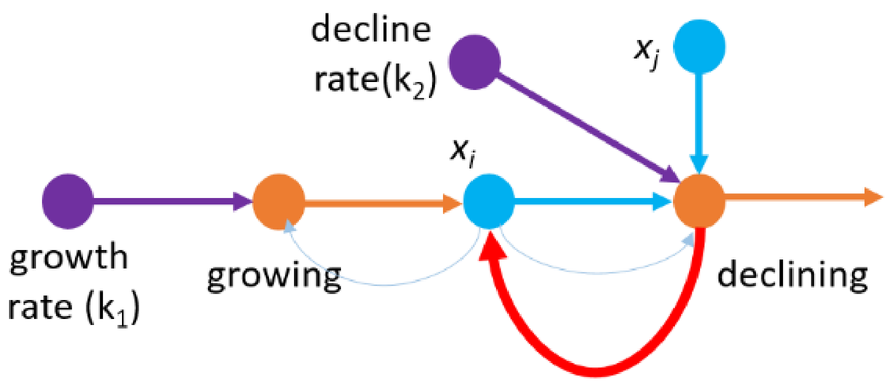

Figure 2 – CLD representation of the converted SF diagram

Analysis of a classical supply chain model

The first example studies a product flow including a factory, a warehouse, a depot, a retailer and a consumer and models transition, holding, ordering and purchasing costs (see Figure 3.). The model predicts total costs in time in respect of the consumer needs.

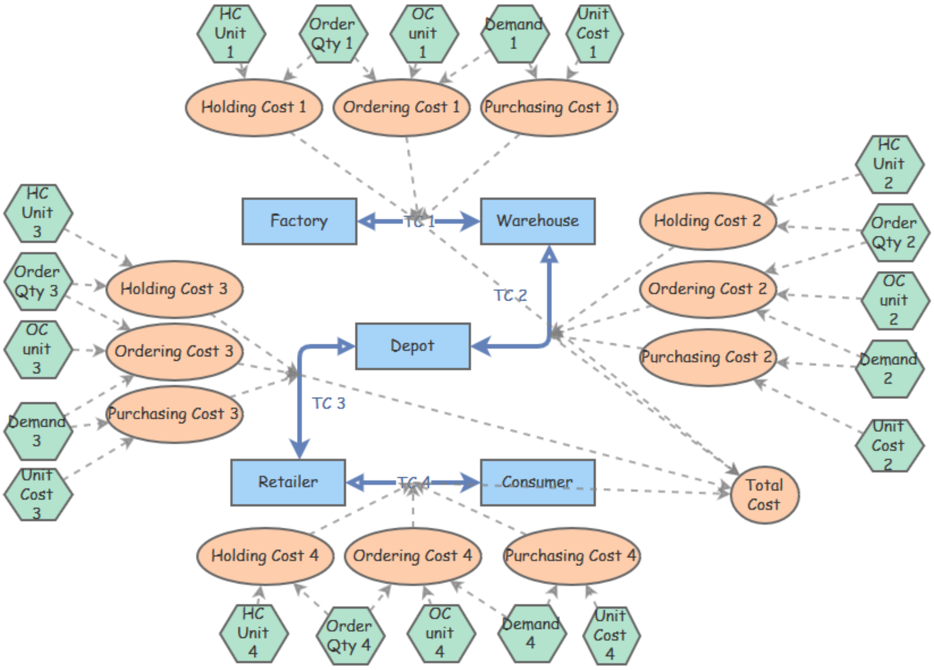

Figure 3 – Illustrative supply chain model in Insight Maker: https://insightmaker.com/insight/106743/Clone-of-Supply-Chain-Model

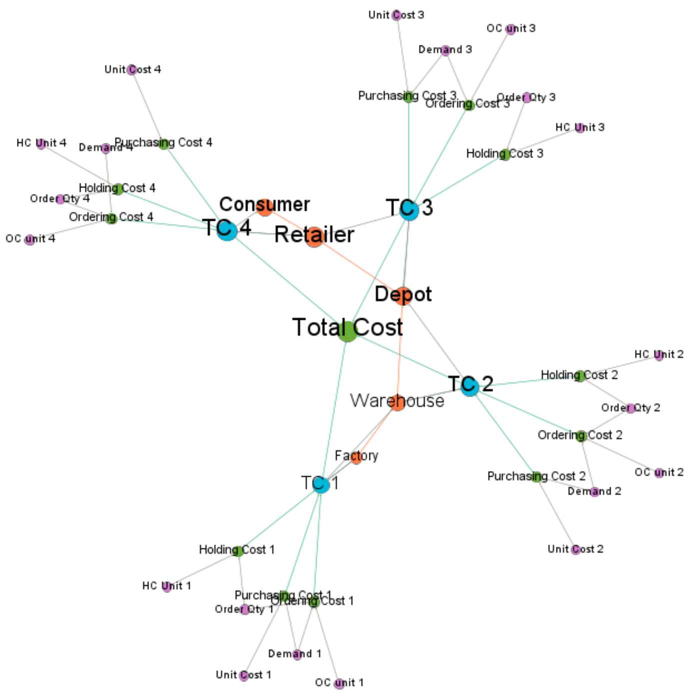

Figure 4 – Network representation of the supply chain model shown in Figure 3.

Figure 4 shows the network that we extracted from the dynamic model. In this figure the size of the nodes are scaled in respect of their PageRank centrality measure. Table 1 shows the five most significant elements of the model. This ranking provides the immediate insight of the essential elements in the model.

Table 1 – Most significant elements of the didactic supply chain model

Model element	Type of element	PageRank
Total Cost	Variable	0.08857
TC4 (Retailer<-> Consumer)	Flow	0.082987
Retailer	Stock	0.082565
TC3 (Depot<->Retailer)	Flow	0.078135
Depot	Stock	0.0711

Figure 5. highlights that modules identify groupings of the key elements in the model. In the next more complex example we will illustrate how a cognitive map of the models can be extracted from such groupings.

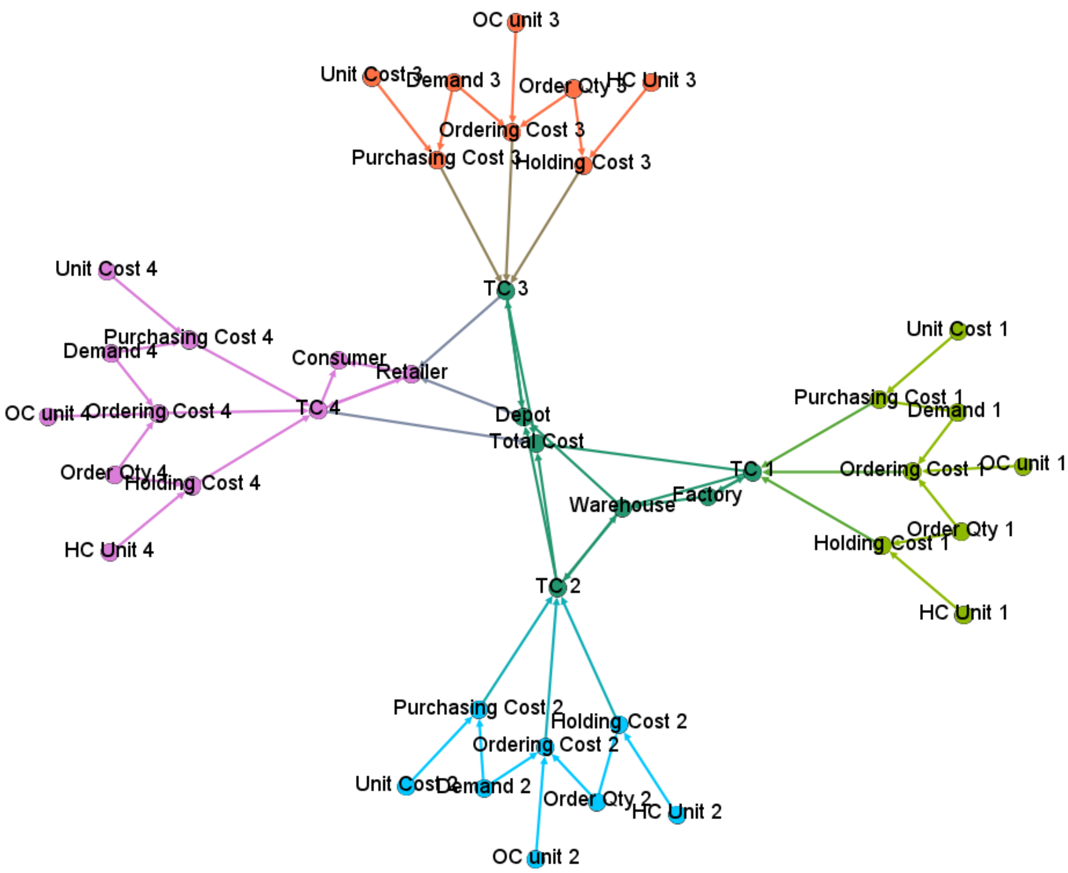

Figure 5 – Communities in the supply chain model represented by different colours

Analysis of the beer distribution game

As a complex illustrative example we study the dynamical model of the beer distribution game created by the Systems Dynamics Group at Massachusetts Institute of Technology in the 1960s. This example became essential as it effectively illustrates the bullwhip effect and the importance of sharing information in the supply chain (Lee, 1997; Nienhaus, 2006). As can be seen in Figure 6., this four-tier supply chain involves retailer, wholesaler, distributor and the factory that should rely on orders from the resellers to create product forecasts, capacity, inventory planning and production schedule (Lee, 1997).

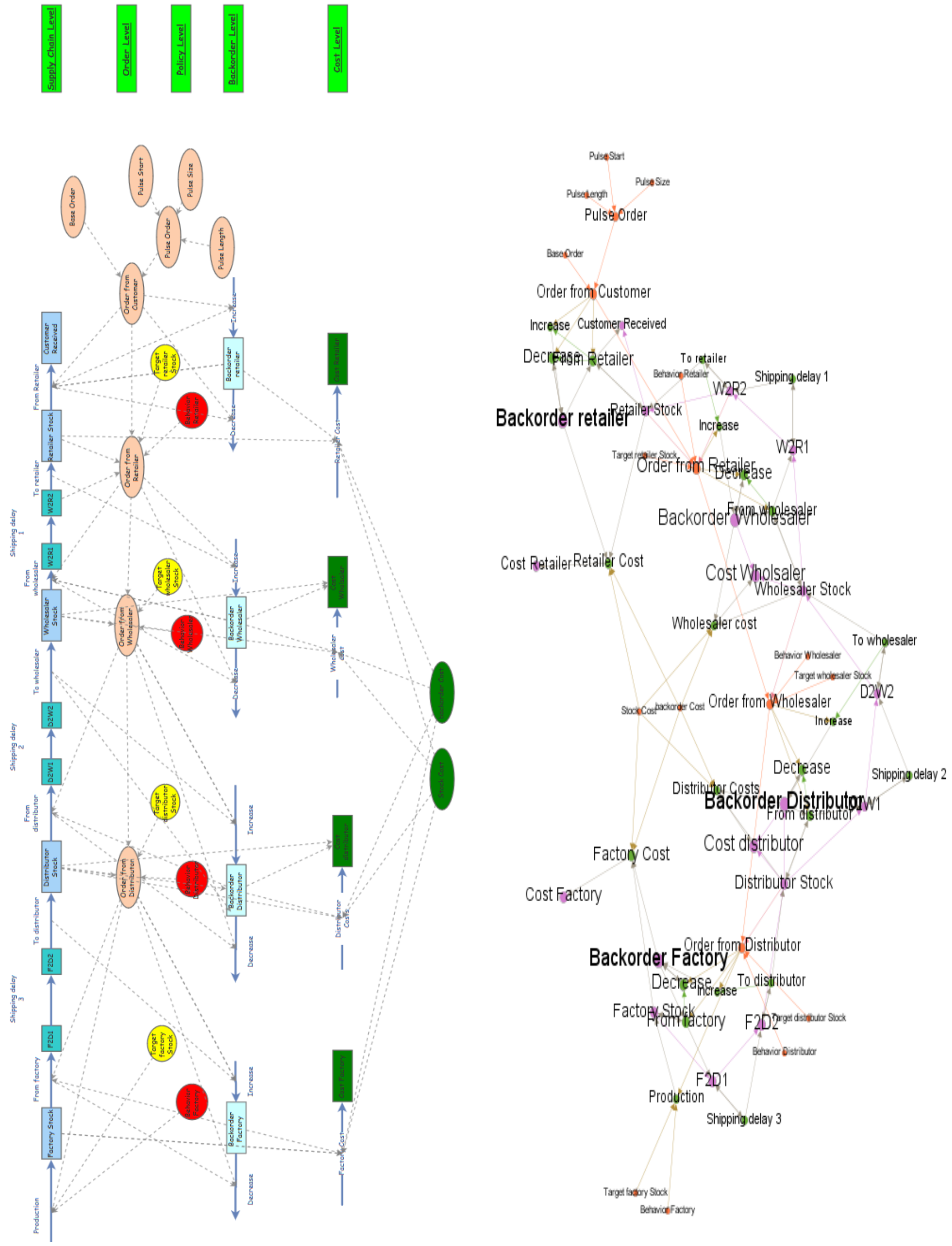

Figure 6 – The beer distribution game simulation model in Insight Maker and its extracted network representation

In the simulation, we see the phenomena in the orders, how a single consumer request stimulates the supply chain. Studying the bullwhip effect brought the following conclusions as to the primary causes of the phenomena (Lee, 1997): Demand forecast updating (backorders), Order batching (backorders), Price fluctuations, and Rationing and shortage gaming (stocks)

As we can see in Figure 7, the most significant elements of the model are the backorders, which is also confirmed by the centality analyis of the network that we extracted from the simulator. The results of this analysis is summarised in Table 2 that shows the PageRank of the first six elements in the model.

Table 2 – Most significant elements from the beer distribution game

Model element	Type of element	PageRank
Backorder Factory	Stock	0.0349
Backorder Distributor	Stock	0.03359
Backorder retailer	Stock	0.03337
Backorder Wholesaler	Stock	0.031918
Cost Distributor	Stock	0.028088
Cost Wholesaler	Stock	0.02757

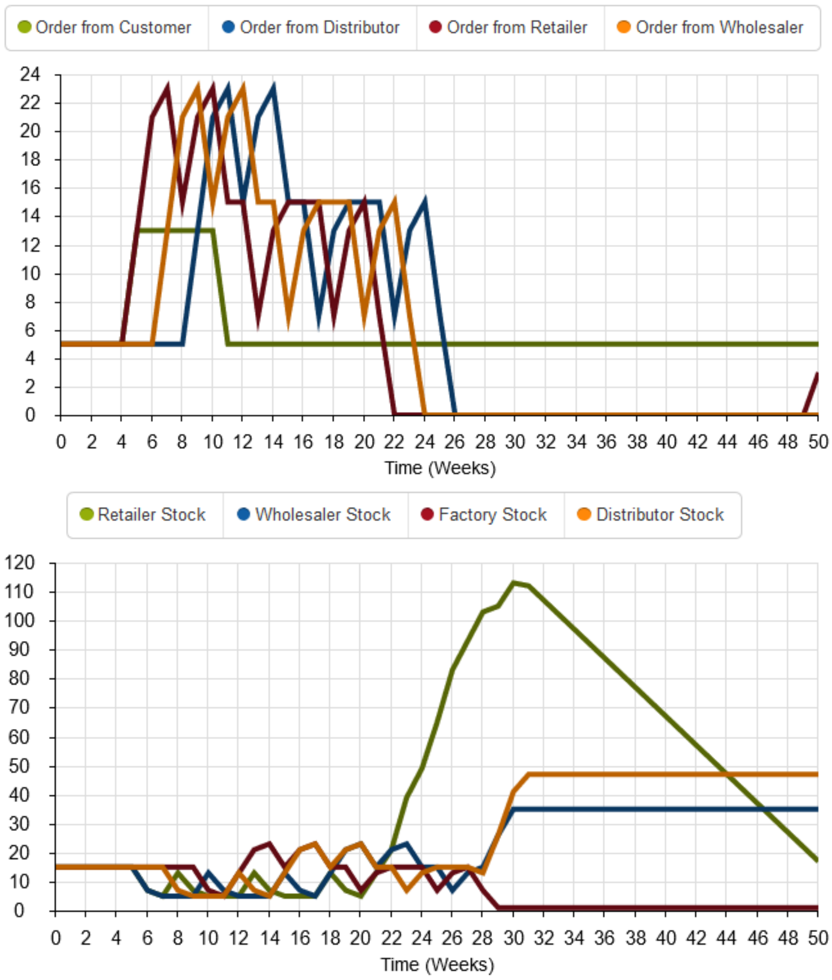

Figure 7 – The simulation of the beer distribution game shows the bullwhip effect. Top: Orders of each player. Bottom: The inventory levels illustrate how a customer order influences the players and how it affects the orders and inventories of the supply chain

Based on the modularity analysis of the extracted network we can identify a cognitive map of the model (see Figure 8.). The reachability analysis (RA) shows more detailed information by highlighting how the state variables influence each other. In Figure 9 the edge weights are inversely proportional to the transitive path lengths, so the thicker is the edge, the more direct is the influence between the state variables.
In this example, the RA explains that the bullwhip effect is formed due to there is no significant feedback to the factory.

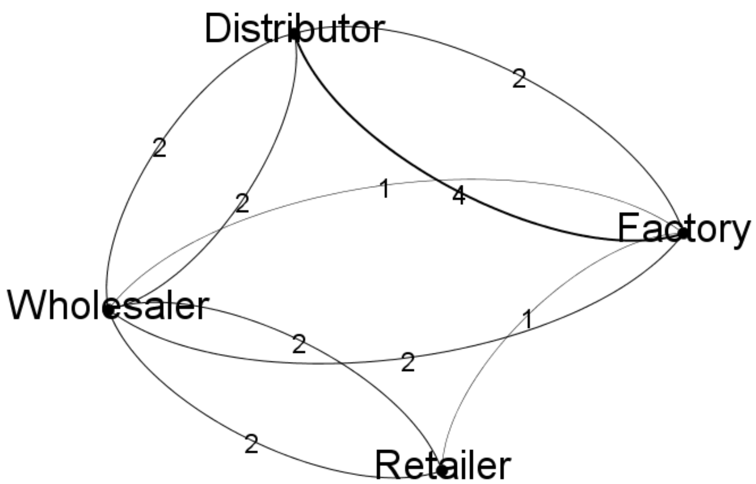

Figure 8 – Beer distribution game cognitive map

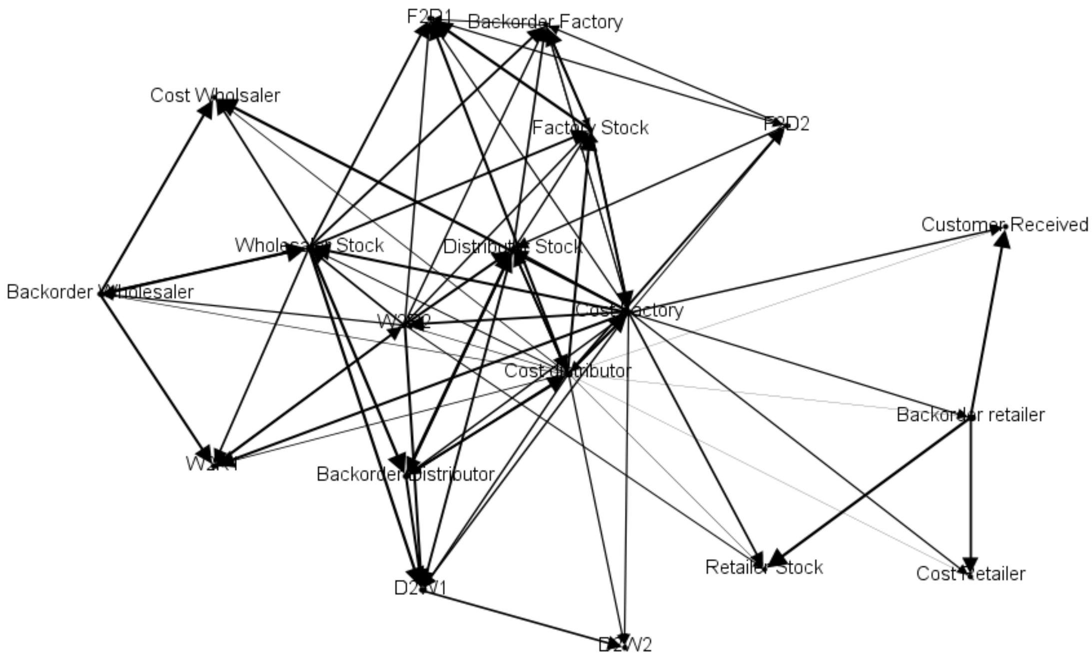

Figure 9 – Beer distribution game reachability analysis

Conclusion

We proposed a network-based methodology to structure and visualize knowledge extracted from system dynamic models. Based on the analysis of two classical models of supply chain management we demonstrated that the developed toolset can be used for the model validation and decision support by visualizing the importance and interconnectedness of the state variables on easily interpretable cause loop diagrams.

References

Bar-Yam, Y. (1997). Dynamics of complex systems (Vol. 213). Reading, MA: Addison-Wesley.

Barabási, A. L. (2016). Network Science. Network Science, by Albert-László Barabási, Cambridge, UK: Cambridge University Press, 2016.

Birkland, T. A. (2014). An introduction to the policy process: Theories, concepts and models of public policy making. Routledge.

Doyle, J. (1999). Rational decision making. MIT encyclopedia of the cognitive sciences, 701-703.

Fedorowicz, J., & Williams, G. B. (1986). Representing modelling knowledge in an intelligent decision support system. Decision Support Systems, 2(1), 3-14.

Girvan, M., & Newman, M. E. (2002). Community structure in social and biological networks. Proceedings of the national academy of sciences, 99(12), 7821-7826.

Jackson, T., & Radunskaya, A. (Eds.). (2015). Applications of Dynamical Systems in Biology and Medicine (Vol. 158). Springer.

Lee, H. L., Padmanabhan, V., & Whang, S. (1997). The bullwhip effect in supply chains. Sloan management review, 38(3), 93.

Melby, P., Weber, N., & Hübler, A. (2005). Dynamics of self-adjusting systems with noise. Chaos: An Interdisciplinary Journal of Nonlinear Science, 15(3), 033902.

Nienhaus, J., Ziegenbein, A., & Schönsleben, P. (2006). How human behaviour amplifies the bullwhip effect. A study based on the beer distribution game online. Production Planning & Control, 17(6), 547-557.

Gandolfo, G. (1971). Economic dynamics: methods and models (Vol. 16). Elsevier.

Page, L., Brin, S., Motwani, R., & Winograd, T. (1999). The PageRank citation ranking: Bringing order to the web. Stanford InfoLab.

Richardson, G. P. (1986). Problems with causal-loop diagrams. System dynamics review, 2(2), 158-170.

Richardson, G. P. (1997). Problems in causal loop diagrams revisited. System Dynamics Review, 13(3), 247-252.

Saysel, A. K., & Barlas, Y. (2006). Model simplification and validation with indirect structure validity tests. System Dynamics Review, 22(3), 241-262.

Spector, J. M., Christensen, D. L., Sioutine, A. V., & McCormack, D. (2001). Models and simulations for learning in complex domains: Using causal loop diagrams for assessment and evaluation. Computers in Human Behavior, 17(5), 517-545.