Proceedings of the Sixth International Conference on
Railway Technology: Research, Development and Maintenance
Edited by: J. Pombo
Civil-Comp Conferences, Volume 7, Paper 14.4
Civil-Comp Press, Edinburgh, United Kingdom, 2024
ISSN: 2753-3239, doi: 10.4203/ccc.7.14.4

Network Theory Approach to Analysing Knock-On Effects in Rail Vehicle Design

S. K. Abburu, C. Casanueva and C. J. O'Reilly

**The Centre for ECO2 Vehicle Design,
KTH Royal Institute of Technology,
Stockholm, Sweden**

Abstract

Rail vehicle models have become increasingly complex, posing challenges in extracting insights using traditional model representations as they require numerous iterations to achieve a satisfactory solution. This complexity leads to high computational and time costs and possibly resulting in inefficient vehicle design. To alleviate these limitations, network models are proposed as an alternative representation in this paper. These models enable the analysis of structure, behaviour, and patterns of interactions, facilitating an understanding of knock-on effects across disciplines and subsystems. The terminology, benefits, and capabilities of network theory in early-stage vehicle design are presented in this paper, along with the aspects to consider and methods for developing network models. The applicability of network theory metrics and algorithms is demonstrated using a railway traction system example. Results indicate that the proposed representations can capture complex system knock-on effects across disciplines and subsystems.

Keywords: knock-on effects, early-stage design, rail vehicle design, network theory, subsystem interactions, traction system.

1 Introduction

The scope of sustainable vehicle design and development has evolved over the years. From being associated mainly to tail-pipe emissions, it has evolved to include economic, social, and environmental aspects. This evolution has gained significant traction in various industrial sectors, especially in the transport sector since the establishment of the Sustainable Development Goals (SDGs) by the United Nations General Assembly in 2015, and the Paris Agreement in 2016. The transport sector is responsible for 2% of the total transport energy consumption [1], contributes for approximately 25% of the total greenhouse gas (GHG) emissions [2, 3] and constitutes nearly 5 to 7% of the total GDP [4]. Moreover, the European Parliament, the Council and the Commission jointly proclaimed transport sector as one of the essential services that everyone has the right to access in the European Pillar of Social Rights [5]. Consequently, the transport sector plays an pivotal role in achieving the Sustainable Development Goals (SDGs). Thus, to achieve the SDGs, the sustainability aspect should be integrated into all stages of vehicle development, spanning from concept development and vehicle design to use-phase and end-of-life management.

Since railways are already among the most energy-efficient transportation modes during operation, resources should be devoted towards improving other aspects of rail vehicles development, including early-stage design, development, and modelling. With conventional vehicle design approach, it is challenging to achieve a satisfactory solution due to the lack of knowledge about the consequences of the design choices on the overall vehicle attributes. This is because the conventional approach typically does not consider interactions among the vehicle subsystems. This lack of consideration further leads to increased number of iterations. To overcome these limitations, it is vital to develop multi-disciplinary and holistic models that can accurately represent the complex interactions within the vehicle system.

Such complex models are being utilised to optimise the design of different train components [6–9] and to improve the vehicle dynamics using multi-body simulations [10, 11]. However, while these models address the limitation of overlooking interactions, they do not alleviate the issue of high computational and time costs. This is because, with such complex models, there are a large number of parameters and variables involved. Consequently, it becomes inherently difficult for the analysts to capture the various indirect interaction effects (or knock-on effects) which result from changes caused by other changes. Moreover, without gaining a fundamental understanding of the structure of the model and the manner in which the various factors influence each other, performing any analysis might result in mere satisfactory solutions despite consuming significant resources.

Therefore, to mitigate the computational and time costs while retaining the ability to analyse the complex indirect interaction effects, an alternate form of complex model representation is necessary. A common approach to represent such complex systems is through network theory. Representing such complex systems as network graphs allows us to gain deeper insights about the structure, behaviour, and pattern of interactions within the model, facilitating an effective utilisation of the models and eventually leading to an efficient vehicle design.

This paper presents a proposal for an alternate form of model representation and an analysis of how this representation can be used to gain insights on the knock-on/indirect interaction effects that can prove useful for a designer or an analyst during the early design stages.

2 Network Theory: Premise and terminology

Network theory, a subset of graph theory, is a branch of mathematics and computer science that studies complex systems by representing them as graphs. Mathematically, a graph is defined as a pair $G = (V, E)$ of sets wherein $E \subseteq [V]^2$ [12]. In other words, a graph G is defined as a finite set of vertices V (or nodes or points) and edges E (or links) where each edge is connected to two vertices. The vertices represent the discrete elements (i.e. variables, parameters, components, subsystems) and the edges represent the relationship between these elements. Generally, the networks are represented as matrices using either adjacency matrices or incidence matrices.

Depending on the relationships captured in the matrices, the models can be represented as different types of graphs as depicted in Figure 1. They can be represented either as simple graphs or multi-graphs depending on the number of edges between two nodes and the existence of self-loops. They can be directed or undirected graphs, depending on whether the edges have a specified direction. They can be weighted or unweighted graphs, depending on whether there are weights assigned to the edges. Additionally, they can be bipartite graphs in which the vertices can be divided into two disjoint and independent sets U and V. In such graphs, every edge $e \in E$ connects a vertex in U to a vertex in V and thus, the graph is defined G as $G = (U, V, E)$.

Representing complex system models as graphs provides various benefits. Foremost, it provides an intuitive means of visualising the connections among different parameters, variables, components, and subsystems. Furthermore, as graphs are generally expressed as matrices, it provides access to a wide array of matrix operations and tools that can be employed to manipulate graphs. Moreover, the graphical representation provides access to various metrics within graph theory such as centrality, density, eccentricity, etc., which can be utilised to analyse the structure, pattern, and behaviour of interactions.

These capabilities of network theory facilitated its applications in various fields, from neuroscience and psychology to transport planning, including engineering design [13]. In engineering design, however, network theory has predominantly been used to improve product and process architecture [14, 15] and develop modular products [16]. Moreover, there are only limited studies which integrate network theory as a tool in the early stages of vehicle design [17]. where it has a potential to address some of the limitations of the conventional vehicle design approach mentioned in Section 1. Particularly, its oversight of interactions among components and the challenges of the

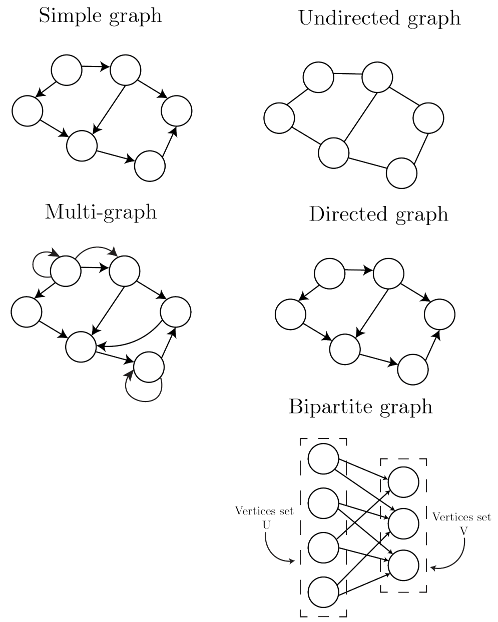


Figure 1: Types of graphs in network theory

traditional representation of complex multi-disciplinary and holistic models, which often require high computational and temporal resources. Moreover, it can assist designers in understanding the consequences of their design choices, their influence on the overall vehicle performance, and in deriving design spaces for different subsystems.

This article demonstrates the approach to developing network models of complex systems and the manner in which these models can be utilised to understand the indirect interaction effects. Specifically, a network model is developed for a rail vehicle traction system, including a 3-phase squirrel cage induction motor, an inverter model, and a drive cycle coupled together. Since the traction motor is a vital part of the vehicle and incorporates various significant disciplines, it serves as a good example of a complex multi-disciplinary model. Furthermore, since the traction motor and the inverter are coupled with a model of the operational drive cycle, it provides a holistic perspective, illustrating the influence of transport and system level requirements over subsystem and component level design.

The manner in which the motor, inverter, and the drive cycle are coupled is depicted in Figure 2. Initially, the necessary input parameters from the motor model are derived from the drive cycle. These parameters are then utilised to design the traction motor. Subsequently, the information from the motor such as its current, voltage, torque, and speed are utilised to calculate the power losses in the inverter. A more detailed account of the dependencies between inverter and the traction motor, and the different steps involved in the design of the traction motor are provided in [18, 19].

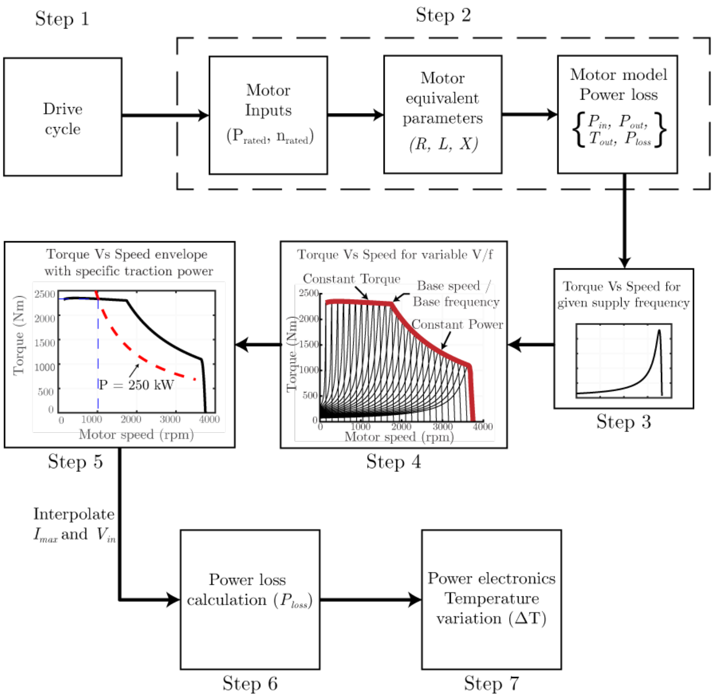


Figure 2: Overall procedure for traction system model

3 Building a network model

In this section, the procedure to develop the network model and the aspects to consider while developing it are described. In this paper, an adjacency matrix is used to represent the network model. Adjacency matrix $A_{n\times n}$ is a square matrix of size $n \times n$ where n is the number of vertices in the graph G. It captures the information on whether two nodes $(v_i$ and $v_j)$ are adjacent, or in other words if two nodes are connected by an edge. The diagonal elements of the adjacency matrix are usually zero unless the graph contains self-loops. A simple adjacency matrix $A_{n\times n}$ of the graph G can be defined using Equation 1.

$$a_{ij} = \begin{cases} 1 & \text{if } v_i v_j \in E, \\ 0 & \text{otherwise.} \end{cases} \tag{1}$$

Subsequently, all the nodes within the model are identified. These nodes become the rows and columns of the adjacency matrix. However, before constructing the adjacency matrix, it is vital to consider various aspects about the model, including:

- The complexity of the model, which indicates the number of nodes in the adjacency matrix.
- The nature of information flow, which indicates whether the network graph is directed on undirected based on whether the relationship is causal or bi-directional respectively.

- The existence of cycles and self-loops, which indicates whether the network graph is acyclic.
- The dynamics of the nodes, which indicates whether the network graph is static or dynamic based on whether the relationship among the nodes are static or temporal.

These aspects influence the type of network model that represents the identified traction system model. Furthermore, they facilitate understanding and verifying the structure of the adjacency matrix. The complexity of the model indicates the size of the adjacency matrix. The nature of information flow indicates whether the adjacency matrix is symmetric or asymmetric. The existence of cycles and self-loops denote determines whether the adjacency matrix is an upper triangular matrix. It is worth mentioning that there has been considerable research on temporal networks, especially in sociology [20] and transport systems [21]. However, there are limited studies which utilise static networks in early stage vehicle design to analyse the relationships among the factors, within and across subsystems and their influence on the overall vehicle attributes.

Therefore, in this paper, an analytical model of the traction system which includes motor, inverter, and drive-cycle is utilised to develop a network model. The different variables and parameters of the traction system model's equations are represented as nodes and the relationship between the variables and parameters are represented as edges. The developed analytical model of the traction system has no cycles or self-loops. This is because the traction motor design is developed based on the provided drive cycle and there is no feedback involved in the model. This indicates that the information flow is unidirectional and acyclic, representing the traction system as a Directed Acyclic Graph (DAG). Moreover, since this is an analytical model, the relationships between the different nodes (or factors) are predetermined. Therefore, a simple unweighted adjacency matrix can be constructed with the already available information on the relationship between the different nodes.

However, in case of an empirical or numerical models, the data needs to be processed using certain algorithms to identify the dependencies between different factors of interest. One approach is to utilise Local Sensitivity Analysis (LSA) or Global Sensitivity Analysis (GSA) to analyse the sensitivity of the input factors over the different factors and construct the adjacency matrix based on the sensitivity indices. There are also dimensionality reduction techniques that allow to identify the most important factors among all the factors of interest and thus, reducing the number of nodes in the network model. For dynamic systems, there are algorithms such as Sparse identification of Nonlinear Dynamics (SiNDy) that allows to identify the underlying governing equations of the dynamic systems which can further be utilised to understand the dependencies between different factors of interest.

Once all the nodes and their relationships are identified, they can be used to construct the adjacency matrix where the factors (or parameters or variables) of the model are represented as the rows and columns of the matrix. As indicated in Equation 1, if node i influences node j, then the corresponding cell $a_{i,j}$ of the adjacency matrix

$A_{n\times n}$ has a value of 1, otherwise it has a value of 0. This procedure is followed for every node pair of the adjacency matrix. The constructed adjacency matrix then represents the relationships captured in the traction system model. Thus, the constructed adjacency matrix is then used to develop the network model of the traction system model using the *digraph* function in MATLAB. The resulting network graph is depicted in Figure 3. These nodes represent the geometrical, electrical, magnetic, electro-magnetic, and thermal parameters which determines the design of the traction motor.

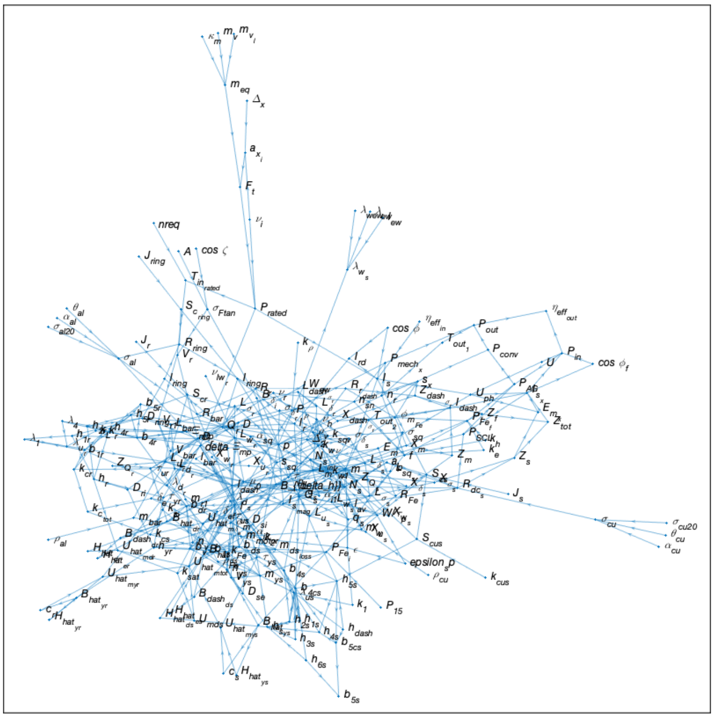


Figure 3: Network representation of the traction system model

However, it is evident from Figure 3 that the network representation of complex models can become quite cumbersome. Thus, it is necessary to utilise the metrics of network theory to gain meaningful insights about the model.

4 Analysis of traction system through network metrics and algorithms: A case study

In this section, the capabilities and applicability of various metrics and algorithms within network theory are discussed and exemplified with a railway traction system example. These metrics can be utilised to analyse the structure, behaviour, and pattern of interactions within model. Furthermore, they facilitate the understanding of indirect (or knock-on) interaction effects within the model.

At the node level, measures such as *degree centrality* provide insights on how well a node is connected to other nodes based on its direct connections. This information can be used to understand the most influential and most influenced nodes. The measure *closeness centrality* indicates the proximity of a node to all the other nodes in the network based on the average shortest distance between the nodes. This information can be utilised to identify nodes that can be modified to swiftly spread the change within the network. Thus, these metrics can provide information on the manner in which different nodes are connected, or in other words, the structure of the network.

At network level, algorithms such as Breadth-First search (BFS), Depth-First search (DFS), and Dijkstra's algorithm can provide information about the manner in which the information flows in the network. BFS is used to identify the shortest path between the source node and all other nodes connected to the source node. Starting at the source node, BFS identifies all the nodes connected to the source node. Then, it systematically identifies all nodes connected to the nodes at the current depth level before moving to the next depth level. Thus, if the goal is to identify all the direct and indirect interaction effects caused by modifying a given input, BFS is utilised. However, if the goal is to identify all the direct and indirect interactions that are caused to influence a given output, then BFS needs to be executed in reverse. This algorithm is termed as Reverse Breadth-First Search (RBFS) in this paper. The algorithm of RBFS works in a similar fashion as BFS except now the target node becomes the source node and instead of identifying the successors of the given node, the algorithm identifies the predecessors of the node. In contrast to BFS and RBFS, DFS explores one branch completely before backtracking to the source node and then continues to explore the next branch. DFS is commonly used to identify cycles and self-loops in the network.

Dijkstra's algorithm [22] is commonly used to find the shortest path from a source node to all other nodes in a non-negative weighted directed graph. It builds on the principles of BFS, but it iteratively selects the node with the shortest distance from the source node. The algorithm ensures that all nodes that are reachable by the source node are visited. Thus, Dijkstra's algorithm identifies the shortest paths from the source nodes to all other nodes reachable by the source node. If the goal is to identify the most direct (or shortest) path to influence an output with a given input, Dijkstra's algorithm is utilised. These algorithms therefore, serve different purpose based on the intent of the analysts. Thus, before utilising these algorithms, the analyst must clarify the insights that the analyst wishes to obtain. To exemplify these algorithms, the traction system with the induction motor, inverter, and the drive cycle is considered.

To exemplify these algorithms, the traction system with the induction motor, inverter, and the drive cycle represented in Figure 3 is considered. In this case study, if the input factor of the motor model, required rated power P_{rated}, which is derived from the drive cycle, is modified, then, all the factors that are influenced by this change can be identified using BFS as depicted in Figure 4. To identify all the factors that are modified to influence the output, mass of the motor m_{motor}, RBFS is utilised and the result is depicted in Figure 5.

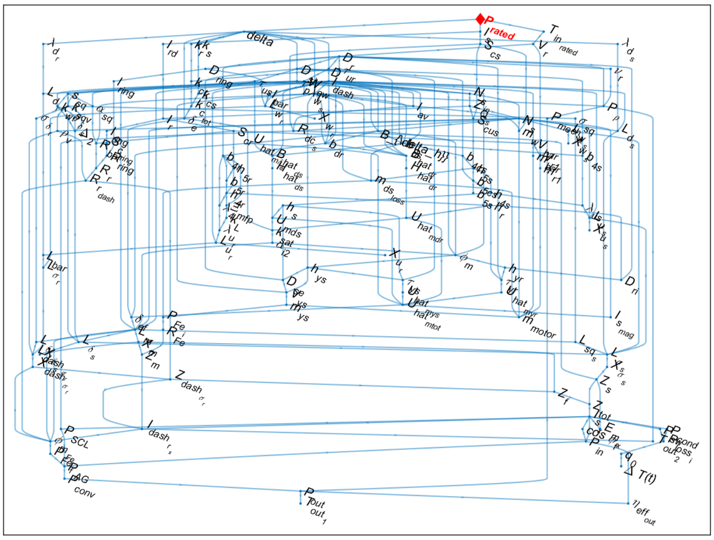


Figure 4: Breadth-First Search of motor input

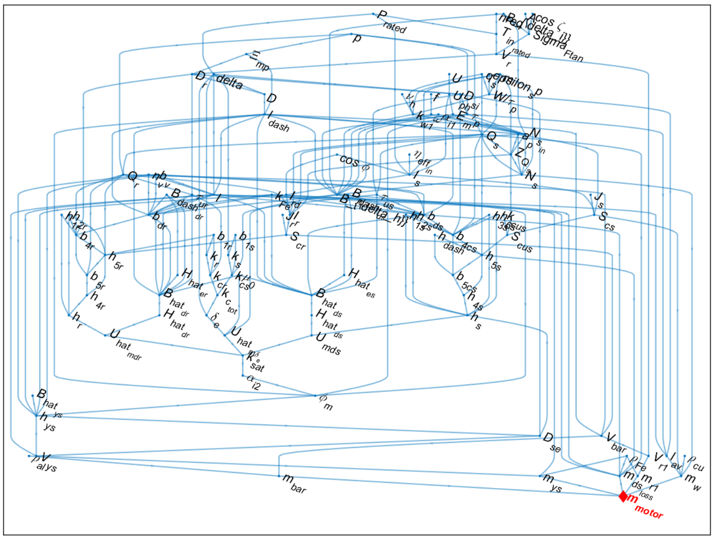


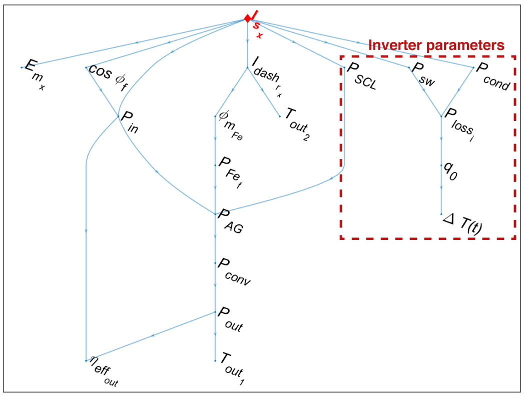
Figure 5: Reverse Breadth-First Search of motor output

These metrics and algorithms can prove instrumental from a design perspective, as they help the designer to focus on the influential parameters while pruning the non-influential parameter branches from the analysis. Thus, reducing the complexity of the analysis. Furthermore, they help in identifying all the parameters and variables that are modified by modifying the input factor. This helps the designer understand the consequences of modifying a factor and also keep track of the changes that occur in different subsystems. It can even help in deriving feasible design spaces of the input factors.

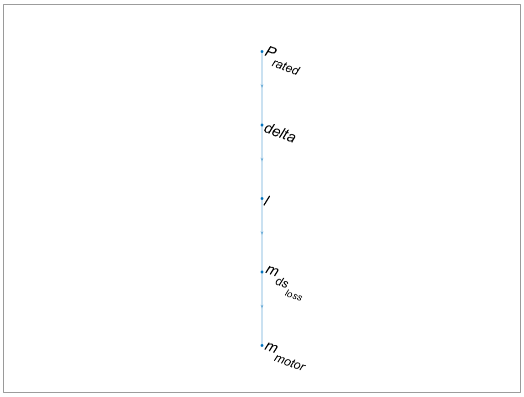
Moreover, these algorithms can be used to identify the path of influence between two subsystems. For example, it is assumed in this paper that the inverter design is based on the motor design, and the inputs to the inverter from the motor are known.

Using BFS, the path of influence from the the motor input to the parameters and variables in the inverter can be identified. This is achieved by assigning the stator current value I_{s_x} as the source node for the BFS and the result is depicted in Figure 6a. Thus, the designer can identify not only the factors influenced by the change but also the change propagation's path.

Additionally, Dijkstra's algorithm can be used to identify the most direct path of influence without considering the knock-on effects. For example, the direct path between the input P_{rated} and the output m_{motor} identified using Dijkstra's algorithm is depicted in Figure 6b. However, it must be noted that although Dijkstra's algorithm can be utilised for unweighted or equally-weighted graphs, it is more efficient to utilise BFS for such graphs due to simplicity. Moreover, the time complexity of BFS is $O(V + E)$, and for Dijkstra's algorithm, it is $O((V + E)logV)$. Therefore, in general, for equally weighted or unweighted graphs, it is efficient to use BFS.



(a) BFS of inverter and motor coupling factors



(b) Shortest path of motor input and output

Figure 6: BFS and Dijkstra's algorithm application to motor network model

5 Possible analysis expansions

It can be noted from Figure 4 and Figure 5 that while analysing the knock-on effects, there are a large number of factors that are involved. Consequently, there are a large number of paths that connect these input factors and output factors. However, all paths do not have similar influence on the outputs. There can be situations where only few paths have a major influence while rest of the paths have no influence at all. Identifying these influential paths can further reduce the complexity of the network models. Therefore, to identify these influential paths, a combination of Local Sensitivity Analysis (LSA) and Global Sensitivity Analysis (GSA) methods shall be performed in the future. These methods provide sensitivity indices which indicate the most influential input parameters for a given combination of inputs and outputs. This would allow the designer to focus only on the influential paths. Moreover, by assigning sensitivity indices between factors as edge weights, Dijkstra's algorithm can be utilised more efficiently.

Furthermore, the present network model of the traction system shall be expanded to include other subsystems such as pantograph, frictional and regenerative brakes, gearbox design and wheel geometries. This expansion is possible due to the presence of factors that couple two subsystems. In the traction system example, the factors motor current I_{s_x} and voltage E_m couple the motor and the inverter designs. Similarly, there are coupling factors between different subsystems as depicted in Figure 7. These coupling factors are then utilised to expand the network model. This expansion can be achieved by utilising multilayered networks where each layer is a subsystem and the coupling factors act as the edge between two layers. Integrating other subsystem in the developed network model will enable to gain holistic insights about the vehicle. This will further help in facilitating effective utilisation of resources and developing efficient vehicle designs.

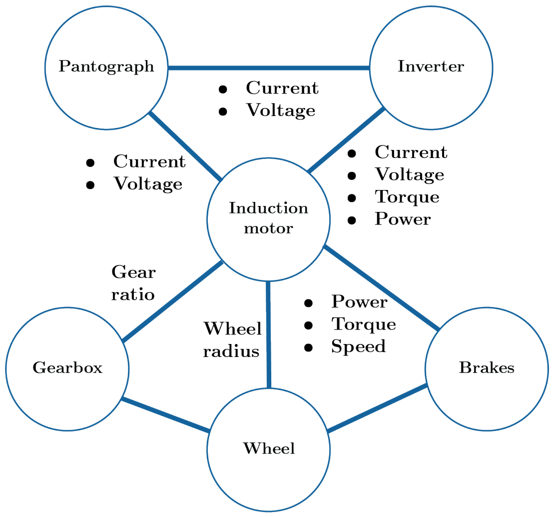


Figure 7: Future expansion of network model with different subsystems and their coupling factors

6 Summary

In this paper, an alternate form of representing complex models is presented and its applications were demonstrated. Initially, the terminology required to understand the concepts of network theory were provided. A brief account on the different forms of graphs and the manner in which they are represented was provided. The various benefits of representing complex systems via network theory, the capabilities of network theory, and its application in various fields were discussed. It was identified that although network theory has been predominantly utilised in vehicle development in areas such as product and process architecture. However, the utilisation of network theory within early stage vehicle design is limited. The benefits of utilising network theory at this phase were explained. A complex traction system model was introduced to exemplify the applications and benefits of utilising network theory in early stage vehicle design.

Further, an introduction to adjacency matrix is given, which is the matrix representation of network model. The different factors that need to be considered while developing the network model and their impact on the network model type were also discussed. The different approaches and algorithms that can be utilised to construct the adjacency matrix of analytical, empirical, and numerical models were discussed. A brief account on the procedure followed to develop the adjacency matrix, and sequentially, the network model of the traction system analytical model was provided.

The necessity and the benefits of utilising metrics and algorithms on the developed network model were discussed. The different metrics of network theory including, degree centrality and closeness centrality, and their significance in identifying the knock-on effects in traction system model were discussed. The established traversal algorithms such as Breadth-First Search, Depth-First Search, and Dijkstra's algorithm, their principle, capabilities, and significance in identifying the knock-on effects were discussed. The Reverse-Breadth-First Search (RBFS) algorithm developed in this paper to identify knock-on effects for a specific output was introduced. The capabilities of these algorithms were demonstrated using the traction system model.

It is observed that BFS is more useful to identify the knock-on effects of modifying an input, RBFS is useful for identifying all the factors that are modified to influence a given output, and Dijkstra's algorithm is useful for identifying the shortest path between two factors, and it is especially efficient when dealing with non-negative weighted directed graphs. Thus, the proposed network representation was able to identify knock-on effects in the traction system model.

However, while exemplifying using BFS and RBFS, it was observed that there are a large number of factors and paths involved that might not have a significant impact on the outputs. Therefore, it was mentioned that, in the future, a combination of Local and Global Sensitivity Analysis shall be performed to filter out these non-influential factors and paths. Furthermore, other subsystems and components (pantograph, brakes, gearbox, and wheels) shall be integrated into the proposed network model to expand the knowledge that can be gained about the vehicle model. This will further translate into effective utilisation of resources and efficient vehicle design.

Acknowledgements

The authors would like to thank the Centre for ECO^2 Vehicle Design, which is funded by the Swedish Innovation Agency Vinnova (Grant Number 2016-05195) and the strategic research area TRENoP for their financial contributions to this work.

References

[1] International Energy Agency, *The Future of Rail: Opportunities for energy and the environment*. OECD, Feb. 2019.

[2] United Nations Department of Economic and Social Affairs, *Sustainable Transport, Sustainable Development: Interagency Report | Second Global Sustainable Transport Conference*. United Nations, Oct. 2021.

[3] European Commission and Directorate-General for Mobility and Transport, *EU transport in figures : statistical pocketbook 2022*. Publications Office of the European Union, 2022.

[4] European Commission, Eurostat, N. Jere, L. Corselli-Nordblad, E. Ford-Alexandraki, and G. Xenellis, *Key figures on European transport : 2022 edition*. Publications Office of the European Union, 2023.

[5] European Commission. Secretariat General., *European pillar of social rights*. LU: Publications Office, 2017.

[6] A. Collina and S. Bruni, "Numerical Simulation of Pantograph-Overhead Equipment Interaction," *Vehicle System Dynamics*, vol. 38, pp. 261–291, Oct. 2002.

[7] Y. Yang, W. Zeng, W.-s. Qiu, and T. Wang, "Optimization of the suspension parameters of a rail vehicle based on a virtual prototype Kriging surrogate model," *Proceedings of the Institution of Mechanical Engineers, Part F: Journal of Rail and Rapid Transit*, vol. 230, pp. 1890–1898, Nov. 2016.

[8] J. Cao, H. Yan, W. Li, D. Li, and Y. Wang, "Optimization of stator ventilation structure of high-speed railway traction motor based on the genetic algorithm," *IET Electric Power Applications*, vol. 17, pp. 281–292, Mar. 2023.

[9] S. Ahn, C. Nam, S. Choi, D. An, I. Kim, K. Seo, and C. Sohn, "A study on squeal noise reduction considering the pad shape of the disc brake system for urban railway vehicles," *Journal of Mechanical Science and Technology*, vol. 35, pp. 1923–1933, May 2021.

[10] S. Bruni, J. P. Meijaard, G. Rill, and A. L. Schwab, "State-of-the-art and challenges of railway and road vehicle dynamics with multibody dynamics approaches," *Multibody System Dynamics*, vol. 49, pp. 1–32, May 2020.

[11] S. Bruni, J. Vinolas, M. Berg, O. Polach, and S. Stichel, "Modelling of suspension components in a rail vehicle dynamics context," *Vehicle System Dynamics*, vol. 49, pp. 1021–1072, July 2011.

[12] R. Diestel, *Graph Theory*, vol. 173 of *Graduate Texts in Mathematics*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2017.

[13] S. Shafiei-Monfared and K. Jenab, “A novel approach for complexity measure analysis in design projects,” *Journal of Engineering Design*, vol. 23, pp. 185–194, Mar. 2012.

[14] E. Devendorf, M. Devendorf, and K. Lewis, “Using Network Theory to Model Distributed Design Systems,” in *13th AIAA/ISSMO Multidisciplinary Analysis Optimization Conference*, (Fort Worth, Texas), American Institute of Aeronautics and Astronautics, Sept. 2010.

[15] D. F. Wyatt, D. C. Wynn, J. P. Jarrett, and P. J. Clarkson, “Supporting product architecture design using computational design synthesis with network structure constraints,” *Research in Engineering Design*, vol. 23, pp. 17–52, Jan. 2012.

[16] M. E. Sosa, S. D. Eppinger, and C. M. Rowles, “A Network Approach to Define Modularity of Components in Complex Products,” *Journal of Mechanical Design*, vol. 129, pp. 1118–1129, Nov. 2007.

[17] M. C. Parker, “A Contextual Multipartite Network Approach to Comprehending the Structure of Naval Design,”

[18] S. K. Abburu, C. Casanueva, and C. J. O’Reilly, “A Holistic Design Approach to the Mathematical Modelling of Induction Motors for Vehicle Design,” *Procedia CIRP*, vol. 119, pp. 1246–1251, 2023.

[19] S. K. Abburu, *Vehicle Conceptualisation, Compactness, and Subsystem Interaction*. PhD thesis, KTH Royal Institute of Technology, Oct. 2023.

[20] N. Santoro, W. Quattrociocchi, P. Flocchini, A. Casteigts, and F. Amblard, “Time-Varying Graphs and Social Network Analysis: Temporal Indicators and Metrics,” 2011. Publisher: arXiv Version Number: 1.

[21] D. B. Tomasiello, M. Giannotti, R. Arbex, and C. Davis, “Multi-temporal transport network models for accessibility studies,” *Transactions in GIS*, vol. 23, pp. 203–223, Apr. 2019.

[22] E. W. Dijkstra, “A note on two problems in connexion with graphs,” *Numerische Mathematik*, vol. 1, pp. 269–271, Dec. 1959.