Identification of Systems With Similar Chains of Components for Simulation Reuse

Henri Sohier*, Louis Petitdemange* and Pascal Lamothe**
*IRT SystemX, Palaiseau, France - {firstname.lastname}@irt-systemx.fr
**Stellantis, Velizy-Villacoublay, France - pascal.lamothe@stellantis.com

***Abstract*—Simulation is an essential tool to evaluate a complex system's behavior. Simulation reuse can potentially improve simulation quality, cost, and delivery. However, identifying reusable simulations is a difficult task, often manual and based on limited information. This paper presents a method to facilitate the reuse of specific parts of past simulations. The system to simulate is compared to systems which have already been simulated. The comparison, which permits to identify similar chains of components in the systems' block diagrams, is formalized as inexact graph matching. The comparison takes into account standardized tags as well as block properties defined by a name, a value, and a unit. Similar systems are identified with a limited computational cost. When two systems are similar, a mapping between their components and interactions can be obtained at a higher computational cost. A software prototype is implemented to perform the necessary computations, visualize the results, and accordingly select the simulation parts to reuse. The software prototype is tested with the block diagram of an autonomous electric car. Similar chains of components are successfully identified in the powertrain of a non-autonomous electric car represented in a past simulation. The corresponding part of the past simulation can then be selected for reuse.**

***Index Terms*—diagram, graph, inexact matching, MBSE, reuse, simulation**

I. Introduction

A complex system relies on multiple interacting components. Its development is facilitated by simulation, in particular during the design phase [1]. Simulation reuse can potentially improve simulation quality, cost, and delivery. However, simulation reuse often relies on the manual review of limited meta data regarding past simulations. With this tedious process, past simulations may not be well understood nor trusted [2] and their reuse may not be fit for purpose [3]. Repositories sometimes include elementary simulation blocks representing, for example, a battery or a motor (e.g. [4]). These simulation blocks can be easier to retrieve, but their re-integration then requires more work. Simulation reuse can be improved by automatically reviewing more data on past simulations [5].

The design of a complex system is specified with requirements of different nature. Some requirements specify the system as a black box, and others give details of the components and their functions. In Model-Based Systems Engineering (MBSE), the whole system specification is formalized with diagrams and centralized [6]. Although MBSE can still be considered as difficult to adopt because of methodology and tooling shortcomings [7], representing a part of the system specification in diagrams is a wide-spread practice. For example, different levels of components can be represented with their interactions in a block diagram [8]–[10].

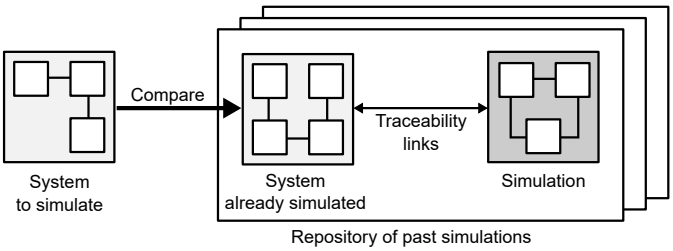


Fig. 1: Comparison of systems for simulation reuse

In this study, it is considered that a simulation starts with a clear specification of the system to simulate. It is also considered that, during the development of the simulation, traceability links are defined between the system to simulate and the simulation. Starting with a clear specification and defining traceability links are good practices not only beneficial to simulation reuse, but also simulation validation. Indeed, they contribute to answer the question "Did I build the right thing?" during conceptual validation [11], [12]. The traceability links can for example associate a simulation block to the system components it represents.

As shown in Fig. 1, systems which have already been simulated and their simulations form a repository. With such repository, a new system to simulate (STS) can be automatically compared to the systems already simulated (SAS) in order to find simulations or parts of simulation to reuse. If the system specification takes the form of a block diagram with block properties, both the block topology and the block properties can be compared. In this study, block topology is compared with graph theory.

Two graphs can be compared by graph matching [13]. Exact graph matching aims at finding an isomorphism between two identical graphs or subgraphs. It can typically focus on the maximum common induced subgraph or the maximum common edge subgraph [14]. Inexact graph matching has the more general aim of finding similarities between graphs which may not be identical. In this study, a new inexact graph matching is defined to reveal any similar subgraph in an STS and SAS. This matching tolerates differences with a limited impact in terms of system functions and simulation models. For example, two subgraphs are considered as similar if it is possible to transform one into the other by splitting or merging

components of the same type. A battery may be split into two batteries, or two controllers may be merged into one controller.

A software is prototyped to run the systems comparison and visualize the results in a new interface. This paper is part of a larger work on the development of complex simulations in the AMC project at IRT SystemX [15].

The remainder of this paper is structured as follow. Section II formalizes the representation of interacting components in a block diagram. Section III presents in detail a new way to compare the topology of two block diagrams, allowing to compare the block properties in Section IV. Section V then presents the software prototype. Section VI finally concludes this paper with a discussion on its results.

II. System Modeling

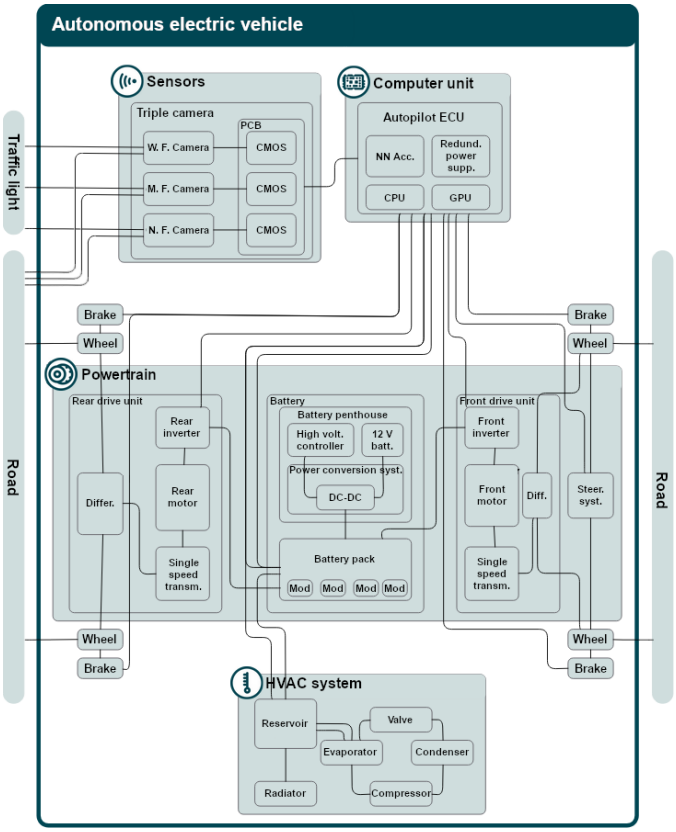


Fig. 2: Block diagram of an autonomous electric car

System modeling is reduced to a simplified block diagram. The system itself, the system's components, and the environment actors are represented by blocks. Functional interactions are represented by undirected lines between these blocks. Fig. 2 shows the block diagram of an autonomous electric car passing traffic lights.

A block is first characterized by a set of properties. Each property is defined by a name (e.g. "Range"), a value (e.g. "350"), and a unit (e.g. "mi"). A block is also characterized by a set of tags from a common database. As shown in Fig. 3, generic tags (e.g."Battery") are progressively refined

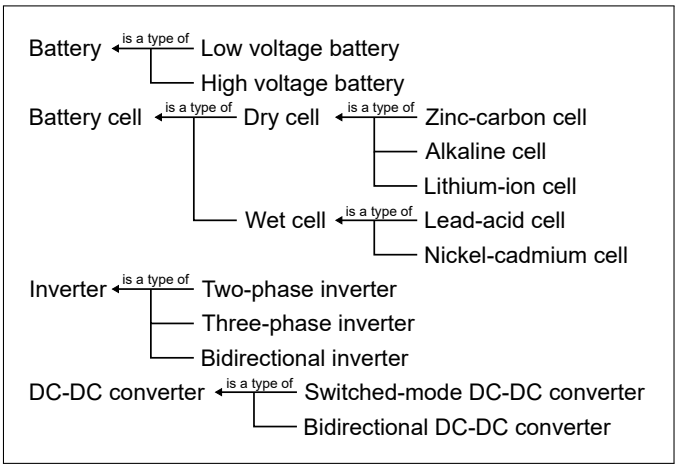


Fig. 3: Example of tag hierarchy for block characterization

with more specific tags (e.g. "Low voltage battery"). Thus, the tags form an hierarchy where a "parent" is refined by "children". Children of a given parent can be of different nature and are not mutually exclusive. For example, an inverter can be both a "Three-phase inverter" and a "Bidirectional inverter". If a tag characterizes a component, the tag's parent also does. For example, a component characterized as a "Low voltage battery" is also a "Battery". From a graph perspective, the tag hierarchy is a directed forest.

The nature of an interaction between two blocks can be characterized with at most one tag. Tags for interactions are different from tags for blocks, but they are also defined in a database used across the different systems. This study uses the tags "mechanical energy", "electrical energy", "electrical digital signal", and "fluid".

III. Topologies Comparison

This section presents a method to compare the geometry formed by the blocks and their interactions in the STS and SAS. All tags are taken into account, but not the block properties. A score is defined to measure the STS and SAS similarity with a low computational cost. If an SAS has a large score, similar STS and SAS parts are identified in detail in order to facilitate simulation reuse.

A. Representation of a diagram as a graph

The block diagram defined in the previous section can be mathematically formalized as a graph. Let G be the graph representing the STS:

$$G = (V, E, \phi) \tag{1}$$

The graph is defined by a set V of vertices v representing the blocks, a set E of edges e representing the interactions, and an incidence function ϕ. The incidence function maps an edge to its endpoints and a label. The endpoints represent the interacting blocks, and the label is a tag identifier. The total number of tags for interactions (e.g. "mechanical energy") is denoted n_{L_E}. Each tag is identified by an integer ranging from

1 to n_{L_E}, and 0 is additionally used to identify the absence of tag. Let $L_E = [\![0, n_{L_E}]\!]$ be the resulting set of integers. The incidence function ϕ is defined as:

$$\phi : E \to \{(\{v_1, v_2\}, l_E) \mid v_1, v_2 \in V, l_E \in L_E\} \tag{2}$$

Given this definition, the graph G is called a multigraph as there can be multiple edges between two vertices. Indeed, there can be multiple interactions between two blocks. The graph G is also called undirected as the edges e are mapped to unordered pairs of vertices $\{v_1, v_2\}$. Indeed, interactions are represented by simple lines, not arrows. Finally, Eq. 2 allows edges to form loops. A loop is formed when the two endpoints of an edge are the same vertex.

The total number of tags for blocks (e.g. "Battery") is denoted n_{L_V}. Each tag is identified by an integer l_V in $L_V = [\![1, n_{L_V}]\!]$. There exists a binary relation $\mathcal{R}_{VL_V}$ such that the statement $\mathcal{R}_{VL_V}(v, l_v)$ is true if and only if the block represented by the vertex $v \in V$ is characterized by the tag identifier $l_v \in L_V$. The departure set of $\mathcal{R}_{VL_V}$ is V, and its destination set is L_V.

As for the STS, let $G' = (V', E', \phi')$ be the graph representing the SAS. There exists a binary relation $\mathcal{R}_{V'L_V}$ such that the statement $\mathcal{R}_{V'L_V}(v', l_v)$ is true if and only if the block represented by the vertex $v' \in V'$ is characterized by the tag identifier $l_v \in L_V$.

B. Mapping for graph comparison

1) Mapping between blocks: The mapping of two blocks is represented by a pair (v, v') of vertices where $v \in V$ and $v' \in V'$. Let S_O be the set of all possible pairs of vertices:

$$S_0 = V \times V' \tag{3}$$

The following rule is followed: a) a block of the STS can only be mapped to the blocks of the SAS with which it shares the largest number of tags; b) a block of the SAS can only be mapped to the blocks of the STS with which it shares the largest number of tags. As a result, a battery can only be mapped to another battery, not to an inverter. This rule ensures a mapping between blocks with similar tags and it simplifies the rest of the graph comparison, but it can be relatively restrictive depending on how precise the tags are. Let S_1 be the set of pairs of vertices representing pairs of mappable blocks:

$$S_1 \subseteq S_0 \tag{4}$$

The formal definition of S_1 relies on the number of elements of L_V shared by a pair of vertices (v, v'). Let $f_{L_V}(v, v')$ be this number:

$$f_{L_V}(v, v') = \big|\{l_V \in L_V \mid \mathcal{R}_{VL_V}(v, l_v) \text{ and } \mathcal{R}_{V'L_V}(v', l_v)\}\big| \tag{5}$$

The vertical bars $|\ .\ |$ represent a cardinality. A pair of vertices (v_1, v_1') belongs to S_1 if and only if, $\forall (v_0, v_0') \in S_0$:

$$\begin{aligned} f_{L_V}(v_1, v_1') &\geq f_{L_V}(v_0, v_1') \\ \text{and } f_{L_V}(v_1, v_1') &\geq f_{L_V}(v_1, v_0') \end{aligned} \tag{6}$$

The powerset $\mathcal{P}(S_1)$ is the set of all subsets of S_1. Let $\mathcal{R}_{VV'}$ be a binary relation over the sets V and V' representing a given mapping. Each possible binary relation $\mathcal{R}_{VV'}$ is defined by a set $H_{VV'}$ of pairs of vertices such that:

$$H_{VV'} \in \mathcal{P}(S_1) \tag{7}$$

The statement $\mathcal{R}_{VV'}(v, v')$ is true if and only if the vertex $v \in V$ is mapped to the vertex $v' \in V'$. The definition of S_1 ensures that every binary relation $\mathcal{R}_{VV'}$ represents a mapping between blocks with similar tags. However, only some of the binary relations $\mathcal{R}_{VV'}$ represent a mapping between blocks with similar interactions.

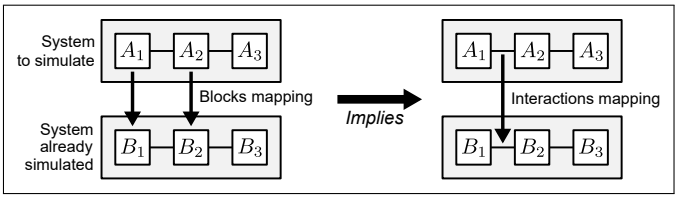


Fig. 4: Deduction of the mapping of interactions

2) Mapping between interactions: A mapping between interactions can be deduced from a mapping between blocks. For example, in Fig. 4, the mapping of the blocks A_1 and A_2 (to B_1 and B_2, respectively) allows to deduce the mapping of the interaction A_1-A_2 (to B_1-B_2).

Let $\mathcal{R}_{EE'}$ be a binary relation representing the mapping of interactions deduced from a given $\mathcal{R}_{VV'}$. The departure set is E, and the destination set is E'. The statement $\mathcal{R}_{EE'}(e, e')$ is true when the edges $e \in E$ and $e' \in E'$ verify two conditions. First, the vertices of e and e' must be $R_{VV'}$-related. If $\phi(e) = (\{v_1, v_2\}, l_E)$ and $\phi'(e') = (\{v_1', v_2'\}, l_{E'})$:

$$\begin{aligned} &\big(\mathcal{R}_{VV'}(v_1, v_1') \text{ and } \mathcal{R}_{VV'}(v_2, v_2')\big) \\ \text{or } &\big(\mathcal{R}_{VV'}(v_1, v_2') \text{ and } \mathcal{R}_{VV'}(v_2, v_1')\big) \end{aligned} \tag{8}$$

Second, the tag identifiers l_E and $l_{E'}$ must be compatible. Two tag identifiers are compatible if they represent the same tag or if one of them represents the absence of tag, that is if:

$$l_E = l_{E'} \text{ or } l_E = 0 \text{ or } l_{E'} = 0 \tag{9}$$

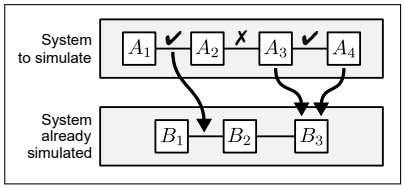


Fig. 5: Interactions considered to measure topology similarity

3) Similarity score: The identification of the best mappings requires a measure of interactions similarity. Interactions similarity can be measured by the number of STS interactions that are mapped to SAS interactions. For example, in Fig. 5, A_1-A_2 is mapped to B_1-B_2. It is also possible to include the number of interactions between STS blocks mapped to the same SAS block (e.g. between two controllers mapped to the same controller). For example, in Fig. 5, A_3 and A_4 are both mapped to B_3. In such case, an SAS with a lower level of detail is optimistically assumed to work similarly to the STS. Let E_0 be the subset of edges that represent these types of interaction:

$$E_0 \subseteq E \tag{10}$$

Formally, an edge e such that $\phi(e) = (\{v_1, v_2\}, l_E)$ belongs to E_0 if and only if:

$$\begin{aligned} &\exists e' \in E', \mathcal{R}_{EE'}(e, e') \\ \text{or } &\exists v' \in V', \mathcal{R}_{VV'}(v_1, v') \text{ and } \mathcal{R}_{VV'}(v_2, v') \end{aligned} \tag{11}$$

The ratio $|E_0|/|E|$ represents the proportion of edges in E_0. This ratio only depends on the set $H_{VV'}$. Indeed, $H_{VV'}$ fully defines $\mathcal{R}_{VV'}$, $\mathcal{R}_{EE'}$, and E_0. Let g be the function that outputs this ratio :

$$g(H_{VV'}) = \frac{|E_0|}{|E|} \tag{12}$$

The mappings which pair blocks with similar interactions are the mappings which maximize g. The maximum value of g measures the similarity of the STS and SAS topologies. This value is in particular obtained for $H_{VV'} = S_1$, when all the mappable blocks are mapped. Thus:

$$\max(g) = g(S_1) \tag{13}$$

As the evaluation of $g(S_1)$ has a limited computational cost, the value of $\max(g)$ can be obtained for every SAS of the simulation repository. Simulation reuse can then be focused on any SAS with a relatively large $\max(g)$.

4) Similar chains of components: Once an SAS is selected, identifying the simulation parts to reuse requires a mapping which maximizes g without pairing blocks with dissimilar interactions. The set T_1 represents all the mappings maximizing g:

$$T_1 = \underset{H_{VV'} \in \mathcal{P}(S_1)}{\operatorname{argmax}} \; g(H_{VV'}) \tag{14}$$

The identification of T_1 requires a combinatorial optimization and has a very high computational cost. In T_1, only some mappings do not pair blocks with dissimilar interactions. When two blocks have dissimilar interactions, the presence of the corresponding pair of vertices (v, v') in $H_{VV'}$ does not affect the value of g. Thus, avoiding the pairing of blocks with dissimilar interactions is equivalent to minimizing the cardinality of $H_{VV'}$. The set T_2 represents the mappings that minimize this cardinality:

$$T_2 = \underset{H_{VV'} \in T_1}{\operatorname{argmin}} \; |H_{VV'}| \tag{15}$$

Given Eq. 11, these mappings can pair multiple STS blocks to the same SAS blocks. This type of pairing, which can reduce the number of mapped interactions in the SAS, should only be used when necessary. The number of mapped interactions in the SAS is maximized:

$$T_3 = \underset{H_{VV'} \in T_2}{\operatorname{argmax}} \; |\{e' \in E' \mid \exists e \in E, \mathcal{R}_{EE'}(e, e')\}| \tag{16}$$

The binary relations $\mathcal{R}_{VV'}$ defined by a set $H_{VV'} \in T_3$ represent mappings that permit to visualize and understand the similarity between the STS and SAS.

C. Toy examples

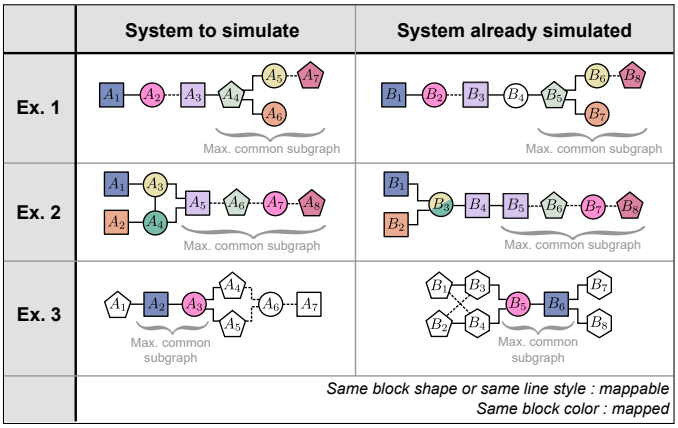


Fig. 6: Example of mappings obtained between blocks

Fig. 6 presents three examples of systems to compare. In each example, the diagram on the left represents the STS and the diagram on the right represents the SAS. Blocks with the same shape are mappable. Similarly, interactions with the same style are mappable. This mappability results from the block and interaction tags.

The colors illustrate the mapping of blocks defined by $\mathcal{R}_{VV'}$ in Section III-B. Thus, in the first example, the block A_1 is mapped to the block B_1 as they are both in dark blue. The maximum common subgraph, representing both the maximum common induced subgraph and the maximum common edge subgraph, is indicated for reference.

In the first example, each STS block is mapped to a different SAS block. Out of 6 STS interactions, 5 are mapped to an SAS interaction. The interaction A_3-A_4 is not mapped as A_3 and A_4 are mapped to B_3 and B_5 which do not directly interact. Thus, given the definition of E_0 in Eq. 11, $g(H_{VV'}) = 5/6$. It reveals a high similarity between the chains of components of the STS and SAT. The simulation of the SAS could be reused by excluding the modeling of B_4.

In the second example, A_3 and A_4 are both mapped to B_3. The block A_3 and A_4 may for example represent interacting

controllers that have the same function as B_3. Conversely, the block A_5 is mapped to both B_4 and B_5. The blocks B_4 and B_5 may for example represent two batteries in series that have the same function as the battery A_5. In both cases, the STS and SAS potentially have similar functions and simulation models. Out of 8 STS interactions, 7 are mapped to an SAS interaction. Although the interaction A_3-A_4 is not mapped, the corresponding edge also belongs to E_0 as A_3 and A_4 are both mapped to B_3. Thus, $g(H_{VV'}) = 8/8$. This high similarity reveals that the STS and SAS chain the same types of components, in spite of local variations in the number of components of a given type.

In the third example, the only STS blocks to be mapped are A_2 and A_3. The remainder of the STS is different from the SAS. In this case, the mapping is focused on the maximum common graph.

D. Greedy heuristic

The maximizations of Eq. 14 is a combinatorial optimizations over $2^{|S_1|}$ possible sets $H_{VV'}$. The cardinality $|S_1|$ depends on the tags of the STS and SAS blocks, but it is at most equal to $|S_0| = |V| * |V'|$. Thus, the worst-case computational complexity is $O(2^{|V|*|V'|})$. For instance, $2^{50\times50} = 3.8 \times 10^{752}$. A greedy heuristic can be used to reduce the number of computations. In a greedy heuristic, a global optimal solution is approximated by local optimal solutions. Let P_1 be a partition of S_1. The partition P_1 is a collection of n mutually disjoint subsets $S_1^{(i)}$ of S_1 whose union equals S_1:

$$\bigcup_{i=1}^{n} S_1^{(i)} = S_1 \text{ and } S_1^{(i)} \cap S_1^{(j)} = \emptyset \; \forall i \neq j \tag{17}$$

It can be chosen to group pairs with a vertex in common. In this case, if S_1 includes three pairs $(v_1, v_1'), (v_1, v_2'), (v_2, v_1')$, the three pairs are grouped in the same subset $S_1^{(i)}$. Such definition of the subsets can then facilitate the analysis of the results. The greedy heuristic involves finding, for each subset $S_1^{(i)}$, the set $T_1^{(i)}$ of local optimal solutions $H_{VV'}^{(i)} \in \mathcal{P}(S_1^{(i)})$:

$$T_1^{(i)} = \underset{H_{VV'}^{(i)} \in \mathcal{P}(S_1^{(i)})}{\operatorname{argmax}} g\big(H_{VV'}^{(i)} \cup (S_1 \setminus S_1^{(i)})\big) \tag{18}$$

The absolute complement $(S_1 \setminus S_1^{(i)})$ is included so that the evaluation of g can take into account the interactions between the blocs related to $H_{VV'}^{(i)}$ and the other blocs of the diagram. An approximated global optimal solution T_1^* is obtained by focusing the evaluation of g on combinations of the local optimal solutions. Let T_0 be the set of all these combinations:

$$T_0 = \Big\{ \bigcup_{i=1}^{n} H_{VV'}^{(i)} \mid H_{VV'}^{(i)} \in T_1^{(i)} \Big\} \tag{19}$$

The approximated global optimal solution T_1^* is then:

$$T_1^* = \underset{H_{VV'} \in T_0}{\operatorname{argmax}} g(H_{VV'}) \tag{20}$$

While the set T_1 is the result of an optimization over a total of $2^{|S_1|}$ alternatives, the local optimal solutions $T_1^{(i)}$ and the approximated global solution T_1^* are the results of optimizations over a much lower number of alternatives. In total, this number is equal to:

$$\sum_{i=1}^{n} |\mathcal{P}(S_1^{(i)})| + |T_0| = \sum_{i=1}^{n} 2^{|S_1^{(i)}|} + \prod_{i=1}^{n} |T_1^{(i)}| \tag{21}$$

Finally, the approximated solutions T_2^* and T_3^* are derived from T_1^* :

$$T_2^* = \underset{H_{VV'} \in T_1^*}{\operatorname{argmin}} |H_{VV'}| \tag{22}$$

$$T_3^* = \underset{H_{VV'} \in T_2^*}{\operatorname{argmax}} |\{e' \in E' \mid \exists e \in E, \mathcal{R}_{EE'}(e, e')\}| \tag{23}$$

E. Matrix interpretation

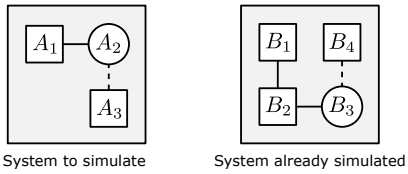


Fig. 7: Example of systems to compare

Most data can be visualized and processed as matrices. Rows represent elements from the STS and columns represent elements from the SAS. Fig. 7 is a simplified example of systems to compare. The diagram on the left represents the STS, and the diagram on the right represents the SAS. As in Fig. 6, blocks of the same shape are mappable. Similarly, interactions with the same style are mappable.

Table I represents the number of tags that two blocks have in common. The values in Table I are consistent with Fig. 7. The number of tags is obtained with the function f_{L_V} (Eq. 5). Two blocks are considered as mappable if the number of tags they have in common is maximum both row-wise and column-wise. This is a matrix interpretation of the definition of S_1 (Eq. 6).

TABLE I: Number of tags in common

	B_1	B_2	B_3	B_4
A_1	4	4	2	4
A_2	2	1	3	3
A_3	4	4	1	4

Table II represents the mapping of the blocks. The variables $\alpha_{i,j}$ are boolean which are equal to 1 if A_i is mapped to B_j, and 0 otherwise. In Table II, a value 0 is shown instead of a variable $\alpha_{i,j}$ when two blocks are not mappable. Let v_i and v_j' be the vertices representing A_i and B_j. From a graph perspective, $\alpha_{i,j} = \mathcal{R}_{VV'}(v_i, v_j')$.

Table III summarizes: a) whether an STS interaction is mapped to an SAS interaction; b) whether an STS interaction is between two blocs that are mapped to the same SAS block (column "Internal"). Table III is deduced from Table II.

TABLE II: Mapping of the blocks

	B_1	B_2	B_3	B_4
A_1	$\alpha_{1,1}$	$\alpha_{1,2}$	0	$\alpha_{1,4}$
A_2	0	0	$\alpha_{2,3}$	0
A_3	$\alpha_{3,1}$	$\alpha_{3,2}$	0	$\alpha_{3,4}$

TABLE III: Mapping of the interactions

	B_1-B_2	B_2-B_3	B_3-B_4	Internal
A_1-A_2	$\beta_{1,1}$	$\beta_{1,2}$	0	$\beta_{1,4}$
A_2-A_3	0	0	$\beta_{2,3}$	$\beta_{2,4}$

For example, $\beta_{1,1} = (\alpha_{1,1} \wedge \alpha_{2,2}) \vee (\alpha_{1,2} \wedge \alpha_{2,1})$ and $\beta_{1,4} = (\alpha_{1,1} \wedge \alpha_{2,1}) \vee (\alpha_{1,2} \wedge \alpha_{2,2}) \vee (\alpha_{1,4} \wedge \alpha_{2,4})$. From a graph perspective, a row of Table III is not a zero vector when the corresponding edge e is in E_0.

The similarity of the STS and SAS topologies is measured by setting all the variables $\alpha_{i,j}$ to 1 and computing the ratio of non-zero rows to the total number of rows in Table III (Eq. 13). If this ratio is considered as large, the variables $\alpha_{i,j}$ are adjusted to find a mapping focused on the similar parts of the STS and SAS. Finding such mapping is equivalent to finding the smallest set of non-zero $\alpha_{i,j}$ (Eq. 15) resulting, in Table III, in: a) the maximum number of non-zero rows (Eq. 14); b) the maximum number of non-zero columns, ignoring the last column (Eq. 16).

In the greedy heuristic, the rows of Table II are grouped according to the position of the zeros showing non-mappable blocks. For example, the rows of A_1 and A_3 are in the same group. The values of $\alpha_{i,j}$ are then optimized in a given group, while being set to 1 in the other groups (Eq. 18). It provides optimal sets of rows that are then combined with each others to find an approximated global optimal solution (Eq. 20, 22 and 23).

IV. Properties comparison

The method presented in Section III permits to compare block topologies, but it does not take into account block properties. As the method presented in Section III results in pairs of STS and SAS blocks, block properties can be compared within each pair of blocks. Block properties are important to compare as they can have an important impact on simulation. For example, the simulation of a car largely depends on the car's shape, mass, battery capacity or motor power.

As an example, Table IV shows the properties of an high voltage battery that is part of an electric car to simulate. Table V shows the properties of a similar battery that is part of an electric car that has already been simulated. In case of mapping between these two batteries, their properties are compared.

This example of mapping illustrates that comparable properties can have different names (e.g. "Voltage" and "Total voltage"). It also illustrates that comparable properties can have different units (e.g. "lb" and "kg"). For each pair of STS and SAS blocks, a mapping between comparable properties is

TABLE IV: Example of properties from an STS block

Name	Value	Unit
Capacity	80	kWh
Voltage	350	V
Weight	1000	lb
Module number	4	
Cell number	4400	
0-100% 3kW charge time	20	h
0-80% 50kW charge time	1.25	h
0-80% 120kW charge time	20	min

TABLE V: Example of properties from an SAS block

Name	Value	Unit
Weight	325	kg
Capacity	50	kWh
Total voltage	400	V
3.7kW charging time (0-80%)	15	h
50kW charging time (0-80%)	1	h

first defined. A property is a triple (a_n, a_v, a_u) where a_n is a name, a_v a value, and a_u a unit. Let A be the set of properties of the STS block, and A' the set of properties of the SAS block. Let $A_d \subseteq A$ and $A'_d \subseteq A'$ be two subsets including all the properties of A and A' with a given dimension. For example, properties in kilograms and properties in pounds all have the dimension of a mass. Finally, let f_s be a function that converts two strings to lowercase and measures their similarity. The Sørensen–Dice coefficient was used in this study . This coefficient is always between 0 and 1, 0 representing highly dissimilar strings and 1 representing identical strings.

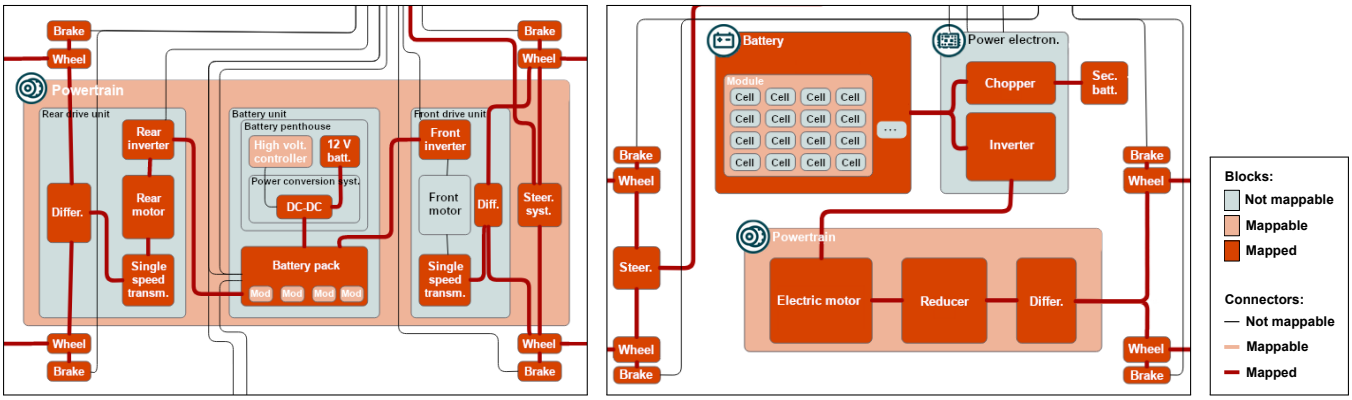
Two properties $(a_n, a_v, a_u) \in A_d$ and $(a'_n, a'_v, a'_u) \in A'_d$ are mapped if: a) the similarity of the names a_n and a'_n is larger than a threshold γ; b) a_n is the property name of A_d that is the most similar to a'_n; c) a'_n is the property name of A'_d that is the most similar to a_n. In other terms, the two properties are mapped if $w = f_s(a_n, a'_n)$ is such that $\forall x_n \in \{x_n \mid (x_n, x_v, x_u) \in A_d, x_n \neq a_n\}, \forall x'_n \in \{x'_n \mid (x'_n, x'_v, x'_u) \in A'_d, x'_n \neq a'_n\}$:

$$\begin{aligned} & w \geq \gamma \\ \text{and } & w > f_s(x_n, a'_n) \\ \text{and } & w > f_s(a_n, x'_n) \end{aligned} \tag{24}$$

In this study, $\gamma = 0.7$. When two properties (a_n, a_v, a_u) and (a'_n, a'_v, a'_u) are mapped, a coefficient c is defined to compare their values. If a_v and a'_v are numbers of the same sign, their ratio is calculated after converting them to SI units with the function g_{SI}:

$$c = \frac{|\min(\{g_{SI}(a_v, a_u), g_{SI}(a'_v, a'_u)\})|}{|\max(\{g_{SI}(a_v, a_u), g_{SI}(a'_v, a'_u)\})|} \tag{25}$$

If a_v and a'_v are numbers of different signs, $c = 0$. The same method is applied to each unit dimension so that the subsets A_d progessively cover the whole set A. Table VI represents the mapping of the properties of Table IV and Table V. Although Table IV includes three different charge times, only one is mapped. The others could not be mapped given Eq. 24.



(a) System to simulate

(b) System already simulated

Fig. 8: Mapping visualization in the software prototype

TABLE VI: Example of mapping between properties

a_n	a'_n	w	c
Weight	Weight	1.00	0.93
Capacity	Capacity	1.00	0.63
0-80% 50kW...	50kW charging...	0.75	0.80
Voltage	Total voltage	0.71	0.88

Once properties are mapped for each pair of blocks, their values can be compared over the whole STS and SAS. Let $H_{VV'}$ be a set of pairs of vertices representing a block mapping, as defined in Section III. Let N be the total number of properties in all the blocks of the STS. The values of the properties are compared with the following function h. The term $\sum wc$ is a weighted sum over every mapping of property in the whole STS and SAS:

$$h(H_{VV'}) = \frac{\sum wc}{N} \tag{26}$$

The larger h is, the more similar the properties are. The set T_3 defined in Eq. 16 is further refined by identifying the mappings which maximize h:

$$T_4 = \underset{H_{VV'} \in T_3}{\operatorname{argmax}}\, h(H_{VV'}) \tag{27}$$

An approximated solution T_4^* can similarly be based on the set T_3^* defined in Eq. 23.

V. Tool Prototyping

A tool was prototyped to support the reuse of simulations by comparing an STS and an SAS. The tool includes: a) a backend that computes a mapping between similar parts of the STS and SAS based on Sections III and IV; b) a frontend that permits to visualize this mapping and select the parts of the simulation of the SAS to reuse. The tool is web-based and was implemented in Node.js for the backend and Vue.js for the frontend. The display of the diagrams more specifically relied on JointJS. The data were saved in JSON. The tool was tested with an example of STS and SAS.

The STS is shown at the beginning of the paper in Fig. 2. It represents an autonomous electric car passing traffic lights. The SAS represents a non-autonomous electric car. The STS includes more sensors and controllers than the SAS, and it also has two electric motors while the SAS only has one. The STS includes 57 blocks and 54 interactions. The SAS includes 54 blocks and 37 interactions. Given the tags of the blocks, there is a total of 4.6×10^{18} possible sets $H_{VV'}$. The greedy heuristic defined in Section III-D was applied with 15 different sets $S_1^{(i)}$ (Eq. 17). A total of $131,642$ different local solutions was first evaluated (term $\sum_i |\mathcal{P}(S_1^{(i)})|$ in Eq. 21). The global optimal solution was then approximated by evaluating $65,536$ combinations of local optimal solutions (term $|T_0|$ in Eq. 21). While this still requires almost 5 minutes of computations with the first version of the software prototype and a 2.7 GHz Intel Core i5 processor, code optimization would realistically result in much better performances.

The tool's interface permits to visualize the STS and SAS diagrams next to each other. As the full diagrams can only be displayed on a large screen, Fig. 8 shows a focus on the powertrains of the cars. The colors represent a mapping. Blocks in red are mapped according to the binary relation $R_{VV'}$ defined by $H_{VV'} \in T_4$ (Eq.27). Interactions in red are mapped according to the corresponding binary relation $R_{EE'}$. Blocks in pink are not mapped but they are mappable according to the pairs of vertices in S_1 (Eq. 4). Interactions in pink are mappable in the sense that there exists a binary relation $R_{EE'}$ (Eq. 8 and 9) where they are mapped.

Thus, the color red represents high similarities between the STS and SAS. Red chains of blocks and interactions reveal similar functional chains with similar purposes. For example, both systems include a battery that provides direct current, an inverter that converts the direct current into alternating current, a motor that converts the alternating current into rotary motion, a transmission that converts the torque and speed, and a differential that equally divides the torque and transmits it to the wheels. Such a red chain is important for simulation reuse. Indeed, it reveals that there are simulation

models for these components and, most importantly, that these simulation models have already been integrated. The integration of consistent models is a difficult and expensive task in simulation.

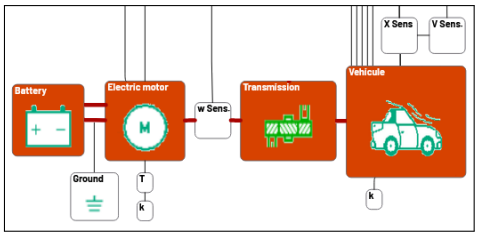


Fig. 9: Simulation of the SAS

In the tool's interface, hovering the mouse pointer over a block highlights both the block in question and the block it is mapped to. Similarly, hovering an interaction higlights both the interaction and the interaction it is mapped to. Clicking on a block shows its properties and their mapping. The tool's interface also includes the diagram of the simulation of the SAS. A part of this diagram is shown in Fig. 9. Thanks to the traceability links available in the simulation repository, a block or an interaction of the simulation diagram is colored the same way as the SAS block or interaction it represents. In the simulation diagram, the red block and their interactions represent good candidate for simulation reuse.

VI. Conclusion

In this study, a system to simulate (STS) is compared to systems already simulated (SAS) in order to find simulations or parts of simulations to reuse. The comparison is focused on block diagrams representing interacting components such as batteries or controllers. A new method is defined to compare both the topology and the properties of the components. A new software prototype implementing this method is presented. The software prototype includes an interface to visually compare the STS and SAS, and accordingly select the simulation parts to reuse.

When two sub-components interact, it can be inferred that the components they belong to also interact. This inference could be computed during a preprocessing step, before the comparison of the STS and SAS. The comparison could also take into account additional information from the STS and SAS specification, regarding for example their states or functions. The comparison could even take into account simulation objectives or simulation scenarios. The main constraint is the availability of formal data. This study focuses on the STS and SAS components as their formal representation in a block diagram is relatively easy to adopt by most organizations.

The different SAS of the simulation library can be screened at a low computational cost, but the identification of a mapping between similar subgraphs then has a high computational cost. A mapping between similar subgraphs is computationally more expensive to find than a bijective mapping between common subgraphs. Furthermore, interactions between components are defined as undirected. Such representation is less restrictive, but it also results in an additional computational cost. The identification of a mapping between similar subgraphs must be shortened by optimizing the code of the software prototype. The issue of the computational cost can also be tempered by running comparisons as a background task.

Beyond this study, the AMC project at IRT SystemX addresses the development of complex simulations. Other functionalities have been implemented to facilitate the traceability between systems and simulations, or to facilitate the specification of a new simulation model with no reuse. A preliminary work was presented in [16].

Acknowledgment

This work was supported by funds from the French Program "Investissements d'Avenir".

References

[1] D. D. Walden, G. J. Roedler, K. J. Forsberg, R. D. Hamelin, and T. M. Shortell, *INCOSE Systems Engineering Handbook*, 4th ed. Wiley-Blackwell, 2015.

[2] S. Robinson, R. E. Nance, R. J. Paul, M. Pidd, and S. J. E. Taylor, "Simulation model reuse: definitions, benefits and obstacles," *Simulation Modelling Practice and Theory*, vol. 12, no. 7, pp. 479–494, Nov. 2004.

[3] M. Pidd, "Simulation software and model reuse: a polemic," in *Proceedings of the Winter Simulation Conference*, vol. 1, Dec. 2002, pp. 772–775 vol.1.

[4] A. I. Bertelrud, O. Balci, C. M. Esterbrook, and R. E. Nance, "Developing a library of reusable model components by using the visual simulation environment," in *Proceedings of the 1997 Summer Computer Simulation Conference*, 1997, pp. 253–258.

[5] J. Aronson and P. Bose, "A model-based approach to simulation composition," in *Proceedings of the 1999 symposium on Software reusability*, ser. SSR '99. New York, NY, USA: Association for Computing Machinery, May 1999, pp. 73–82.

[6] A. Doufene and D. Krob, "Towards a comprehensive design approach for complex systems architecture and optimization," Jun. 2020.

[7] A. Vogelsang, T. Amorim, F. Pudlitz, P. Gersing, and J. Philipps, "Should I stay or should I go? On forces that drive and prevent MBSE adoption in the embedded systems industry," in *Product-Focused Software Process Improvement*, ser. Lecture Notes in Computer Science. Cham: Springer International Publishing, 2017, pp. 182–198.

[8] "SysML FAQ: What is an Internal Block Diagram (IBD)?" [Online]. Available: https://sysml.org/sysml-faq//sysml-faq//sysml-faq/what-is-internal-block-diagram.html

[9] N. M. Karayanakis, *Advanced system modelling and simulation with block diagram languages*. CRC Press, Jun. 1995.

[10] M.-P. Algebra, A. Imaev, and R. P. Judd, "Block diagram-based modeling of manufacturing systems using max-plus algebra," in *2009 American Control Conference*, Jun. 2009, pp. 4711–4716.

[11] D. K. Pace, "Modeling and simulation verification and validation challenges," *Johns Hopkins APL Technical Digest*, vol. 25, no. 2, pp. 163–172, 2004.

[12] R. G. Sargent, "Verification and validation of simulation models," *Journal of Simulation*, vol. 7, no. 1, pp. 12–24, Feb. 2013.

[13] K. Riesen, X. Jiang, and H. Bunke, "Exact and inexact graph matching: Methodology and applications," in *Managing and mining graph data*, ser. Advances in Database Systems. Boston, MA: Springer US, 2010, pp. 217–247.

[14] J. W. Raymond, E. J. Gardiner, and P. Willett, "RASCAL: Calculation of graph similarity using maximum common edge subgraphs," *The Computer Journal*, vol. 45, no. 6, pp. 631–644, Jan. 2002.

[15] "IRT SystemX - Agility and Design Margins." [Online]. Available: https://www.irt-systemx.fr/en/projets/amc/

[16] J.-P. Brunet, H. Sohier, M. Yagoubi, M. Bisquay, P. Lamothe, and P. Menegazzi, "Simulation architecture definition for complex systems design: a tooled methodology," in *Complex Systems Design & Management*. Cham: Springer International Publishing, 2020, pp. 153–163.