Article Identifier
DOI: 10.22032/dbt.58905
URN: urn:nbn:de:gbv:ilm1-2023isc-007:1

60th ILMENAU SCIENTIFIC COLLOQUIUM
Technische Universität Ilmenau, 4 – 8 September 2023
URN: urn:nbn:de:gbv:ilm1-2023isc:1

INTEGRATING LIFE CYCLE ASSESSMENT IN MODEL-BASED SYSTEMS ENGINEERING

Lipšinić, Zvonimir; Pavković, Neven
University of Zagreb, Faculty of Mechanical Engineering and Naval Architecture, Chair of design and product development

Abstract:

The emergence of smart products has led to the development of an increasing number of multidisciplinary systems. For the successful development of such systems, a holistic approach is necessary, such as model-based systems engineering (MBSE). It is argued that certain product development activities could be integrated and improved with MBSE, one such activity being the assessment of environmental impacts. This article presents a case study on the usage of Life Cycle Assessment (LCA) on a MBSE system model. In the study a technical system is modelled with views according to the MagicGRID approach. The scope and goal of the LCA are defined by using SysML diagrams and elements. Additionally, different system variants are modelled to explore the capability of comparing LCA studies. At the end of the case study, the benefits, limitations, and shortcomings of the integration are discussed.

Key words: MBSE, LCA, SysML, model-based systems engineering, Life Cycle Assessment, Case study

1. INTRODUCTION

The methodology of Model-Based Systems Engineering (MBSE) has been developed in the aeronautic, space and defense industry to help systems engineers handle the complexity of the developing systems [1]. Since the emergence of smart products and IoT, the number, variety and complexity of multidisciplinary products has been observed to have a steady increase in all fields, from automotive [2], to medical [3] and agricultural. MBSE is becoming an essential approach for developing such complex products and significant research efforts are aimed at improving the rate of implementation [4] for which one of the main obstacles is the imperfection of software tools and platforms [5]. One of the benefits discussed in the literature is that MBSE can reduce the time and effort of system/product development processes and activities that were previously done manually in document-based approaches. By ensuring consistency and traceability between system elements and providing a clear definition of the system and its environment in the early stages of development, MBSE can also prevent errors in the system/product development process [6].

Along with the growing complexity of new products and systems, there is also an increasing emphasis on sustainability and the impact on the environment. This added aspect of eco design further increases the complexity of the designed product and therefore encourages the use of MBSE in the development process. In the literature [7] it is proposed that some product development activities and processes could be integrated and automated within MBSE, including the analysis of environmental impacts [8]. Current research in this area of MBSE is not sufficiently elaborated. Considering the methods used for the analysis of environmental impacts in document-based systems, Life Cycle Assessment (LCA), as a standardized method, appears to be commonly used in practice. Research has identified some key issues of the method that could be addressed with the application of MBSE. The mentioned issues include the handling of uncertainties within the LCA study [9] and the comparison of different LCA studies [10]. It is recognized that a valid comparison could only be made on products within the same product family. This could be further improved with MBSE since it supports the modelling of variants, both structural and behavioral [11] within the same system model. The end goal of the LCA and MBSE integration is to partially automate the process of conducting the LCA, thus helping the system architect or product designer to predict the environmental impact of the system and improve the decision-making process in the development phase.

1.1. Related work

Similar research, focused on the analysis of environmental impacts of complex systems has been conducted on the example of offroad construction equipment [12]. This research includes the discussion of potential opportunities for improvement and reduction of environmental impacts with a sensitivity analysis in a document-based approach. MBSE, additionally offers model execution and simulation which can enable early validation and verification (V&V) of the system model including the calculation of environmental impacts. This is why considerable research efforts are aimed at the improvement of specialized and domain specific simulation capabilities [13]. Another important aspect that is mentioned throughout the literature is the reusability of model elements [14] in MBSE. In this paper we will explore the potential of reusability with the goal of improving the partial automation of the LCA process.

Regarding the analysis of environmental impacts in the field of MBSE, Bougain et al [8] propose a method for modeling the environmental impact of a system in certain lifecycle phases, using the SysML programming language. The proposed modeling approach is supported by a model example of a 3D printer. Building on the modelling concept, Bougain et al [15] also propose a case-based reasoning (CBR) approach for the integration of eco design within MBSE supporting toolchains. In this case study we aim to include the lifecycle phases which the authors in [8] did not cover (e.g., transport). Further research in the field focuses on defining an approach for sustainable product development based on methods of MBSE using the term System Lifecycle Management [16]. This approach defines, similar to the first LCA phase, the goal of the analysis, after which the system elements with the largest environmental impact are identified and their behavior is analyzed. Means of validation and verification are also proposed, as well as the interpretation of results [17]. Frameworks for analyzing the environmental impacts of multiscale complex systems and system of systems (SoS) have also been proposed [18]. The authors in [18] argue that SysML could be used as an interface between modelling and LCA tools and that SysML has the capability to mitigate LCA boundary selection problems. Besides the boundary selection, other LCA uncertainties are addressed in research. Inkermann [19] identifies the causes and potential remedies for the epistemic uncertainties with the integration of MBSE. He also discusses the potential of a SysML profile extension which would include new classes, similar to the SafeML profile for safety analysis [20]. Regarding relevant research on the topic of sustainability in the manufacturing life cycle phase, Romaniw et al [21] propose an activity-based object-oriented approach for better sustainability assessment. One of the key aspects of their research is the refinement of manufacturing activities into operations which are defined as classes and their specializations. The system element undergoing manufacturing has an addition of relevant environmental values in the system element block which are calculated for each operation.

1.2. Research questions and paper structure

With this case study we aim to answer the following research questions (RQ):

> **RQ1**: *What are the practical benefits, limitations, and shortcomings of using LCA on a MBSE system model?*
>
> **RQ2**: *How can the modelling approach be upgraded to better facilitate the analysis of environmental impacts in MBSE system models?*

The next section discusses the methods and approaches that are used for this case study. The first part of section 3 (case study) is dedicated to the system structure and the description of the analyzed system, after which the subsections are structured corresponding to each LCA phase according to ISO 14040 (Figure 1). The research questions will be addressed in the discussion (section 4), after which the conclusion and outlook are written.

2. METHOD

The methods used for this case study are described in the following three paragraphs. First, the basis for creating the system model is defined with the modelling language and approach, after which the LCA approach is established. The last segment describes the chosen variability modelling method.

2.1. Modelling language and approach

The literature review indicates that besides the Object-Process Methodology (OPM) [22] and the Modelica programming language, the most popular modelling language for the application of MBSE is the standardized System Modelling Language (SysML). In this case study the system model will be created using SysML [23] as the modelling language (notation). In the field of MBSE, it is recognized that a one-size-fits-all modeling approach may not adequately capture the different aspects of various systems, for this reason several modelling approaches have been developed for specific methods, purposes, tools, and programming languages. For this study we chose a more generalized approach designed for SysML: MagicGrid (formerly known as MBSE Grid [24]). MagicGrid has well defined abstraction layers and system aspects that coincide with the four pillars of SysML, making it a relatively easy-to-understand approach. It has also been recently updated and expanded with feedback from the industry [25]. The modelled areas of the MagicGrid, which can be mapped to systems engineering processes from the ISO 15288 standard, (Table 1) will be described along with the system model in the system structure section.

2.2. Life Cycle Assessment

Since the LCA affects all domain levels from the MagicGrid methodology, the relevant LCA diagrams will be constructed in a <<package>> (see Figure 2) that is separated from the general system model. This package is then further divided into two parts:

- the LCA sub-package dedicated for conducting the assessment of the particular system. This sub-package is sorted according to the corresponding LCA phases defined in ISO 14040 (Figure 1).

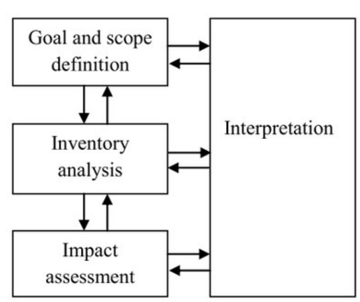

Figure 1 LCA phases

- the LCA elements sub-package which acts as a general object library valid for all systems. This library stores the relevant data types, LCA requirements and environmental constraints which can then be reused in the (active) system LCA package. For this case study, only the objects needed for this specific LCA are created, instead of the whole object library.

The package structure is shown in Figure 2.

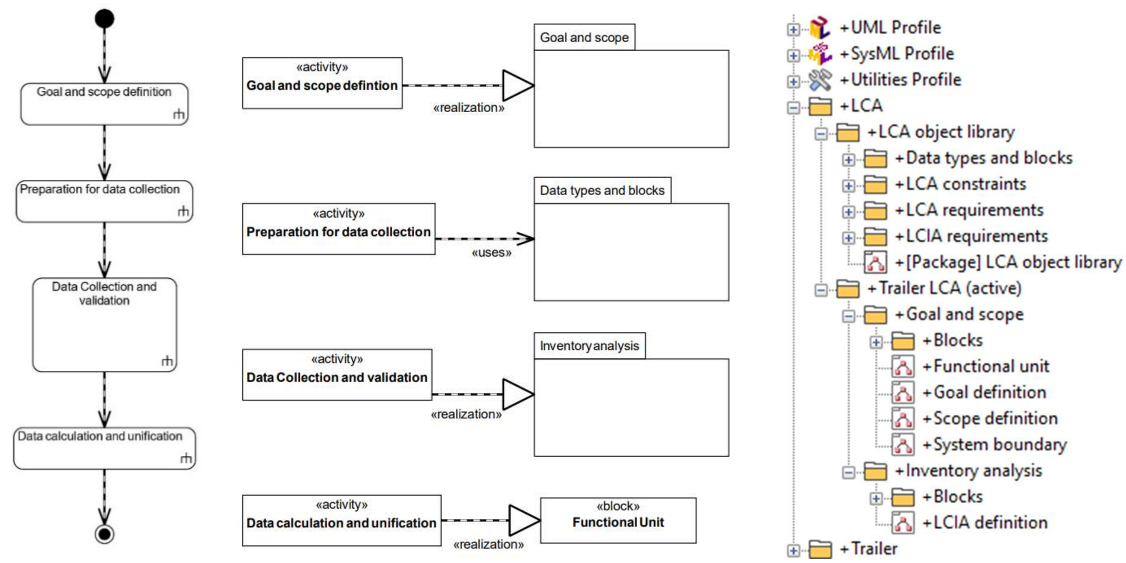

Figure 2 LCA activities with links to the package structure

In the first phase, the goal and scope of the LCA must be defined. This includes a variety of different elements, for example the LCA purpose, data quality, allocation method, interpretation, limitations, and boundary. These LCA elements are modelled as requirements in the general object library package. This method enables the integration of LCA elements with other system elements, with the added capability of verifying the requirement fulfillment and traceability. To further assist in the assessment process, an activity diagram inspired by the procedure diagram in ISO 14044 is created and linked to the package structure. This diagram expands on the LCA phases and details the assessment procedure. Figure 2. shows the top-level activities and links them to the corresponding SysML elements.

In the next phase, the inventory analysis is carried out. Structural and behavioral system elements are organized in a modular form so that each use case defined in the system boundary can be assessed separately and later summed up in the third phase. To calculate the environmental impact, we utilize the parametric diagrams of SysML. The calculated values of each system element or process are then shown as SysML block value properties. Examples of the created parametric diagrams are presented in section 3.3.

2.3. Variability modelling

Recent research [26] is working towards upgrading existing variability approaches that were originally aimed at software product lines to cover other disciplines, in order to better facilitate cyber-physical product line engineering (PLE). This integration of PLE [27] with MBSE is referred to as Model-Based Product Line Engineering (MBPLE). To demonstrate the comparison of the environmental impacts of different system variants for this case study, a variability modelling method is chosen and incorporated in the MagicGrid modelling approach. In the literature review we explored various existing variability modelling methods that have been developed over the years. One well established method in the software industry is the Feature-Oriented Domain Analysis (FODA) method, which focuses on identifying and analyzing the commonalities and variabilities within a product line. Another notable approach is the Common Variability Language (CVL), initially intended to become an Object Management Group (OMG) standard but was later abandoned. It is also worth mentioning that a new variability modelling approach is expected to be proposed in the upcoming SysML 2.0 specification, which holds the potential to provide enhanced support for variability management. For this paper, we selected the Orthogonal Variability Model (OVM), developed by PALUNO Institute [28], as the variability modelling approach due to its standardized nature (ISO 26550), implementation in software tools, and its joint utilization with the SysML notation which enables the modeling of structural and behavioral element variants [29].

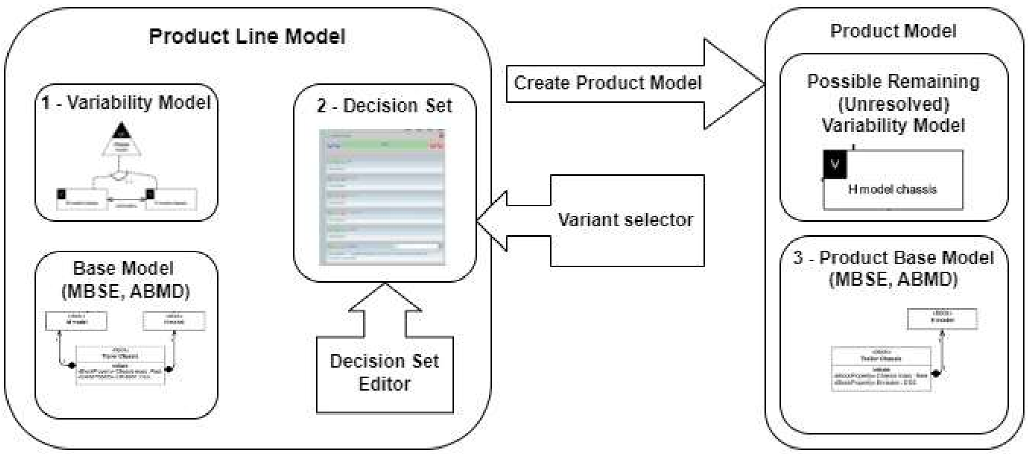

Figure 3 OVM approach in Windchill Modeler 9.5

The system model in the case study is modelled as the product line model, also referred to as a 150% model [29] which includes the LCA package. The tool then generates complete (100%) product models based on the decisions in the variant selector (Figure 3). The model structure and its variations are detailed in the System structure section.

3. CASE STUDY

The system of interest for this case study is a forestry trailer with 20 t loading capacity and active suspension. The trailer system consists of 3 interconnected subsystems (pneumatical, electrical, mechanical). The chosen system does not, due to its relative low complexity, necessarily warrant a MBSE approach, but it could still benefit from the simulation capabilities of MBSE. In the next section, the system model structure is defined, after which the LCA process is described in accordance with the ISO 14040 LCA phases shown in Figure 1.

3.1. System structure

Following the MagicGrid approach, the model structure is defined according to the domains and pillars shown in table 1. Considering the environmental impact of each lifecycle phase for this system, we can confidently say that the production phase has the highest impact, for this reason, not all view specifications of the MagicGrid have been modelled, for example, the system Measure of Effectiveness (MoE) and safety views. Starting from the requirements and behavior pillar, only the top-level system requirements and functions are modelled to demonstrate the traceability to system structure which is later used to define some of the LCA elements (e.g., functional unit). The next step is to model the system within its environment (context) as a black box, after which the subsystems and their interaction are defined in the white box model.

Figure 4 Forestry trailer with active suspension

Table 1 MagicGrid methodology for building the system model

Domains		Pillars				
		Requirements	**Behavior**	**Structure**	**Parameters**	**Safety**
Problem	**Black box**	Customer needs, Legal requirements	Use Cases	System Context	*System MoE (Not modelled)*	*Preliminary risk analysis (Not modelled)*
	White box		System functions	System structure	*Subsystem MoE (Not modelled)*	
Solution		System requirements	*System behavior (Not modelled)*	Refined System structure	Trailer LCA parameters	*Functional FMEA/FTA (Not modelled)*
		Subsystem requirements (Not modelled)	*Subsystem behavior (Not modelled)*	Subsystem structure	*Subsystem parameters (Not modelled)*	
		Component requirements (Not modelled)	*Component behavior (Not modelled)*	Component structure	*Component parameters (Not modelled)*	
Implementation		*Physical requirements (Not modelled)*	Production functions and processes	Production structure	Production parameters	*Design FMEA/FTA (Not modelled)*

For the solution domain (Table 1), a refined system structure was modelled with some of-the-shelf components (e.g. axle, wheel…). In this view the different variants of the model are specified using the OVM method. This includes the decision points and the structural links shown in Figure 5. The decision points are later used to define the model variant (configuration). By selecting the desired variant, for example the chassis variant, from the Figure 5. decision point, the product model will include or exclude the model objects based on the structural links made in the diagram, in this case the M or H variant chassis block. After the structural components of the model are created, the behavior of the model needs to be defined. This includes the use cases that define the system functions in the problem domain and the production system use cases and processes from the implementation domain. Unlike the processes modelled with LCA software tools which have the form of energy and material flows, the processes used for this example had to be modelled with use-case, activity, and block definition diagrams. The use cases hence define the boundary of the analyzed process by including the main process <<activity>> with its allocated <<block>>. The main activity is further decomposed into sub-activities with their own allocated blocks. These atomic process blocks then reference blocks which are used as input materials for the atomic processes. The flow of these activities is shown in an activity diagram, while the <<activity>> and <<block>> allocation relation is defined in the block definition diagram for the high-level process as shown in Figure 7.

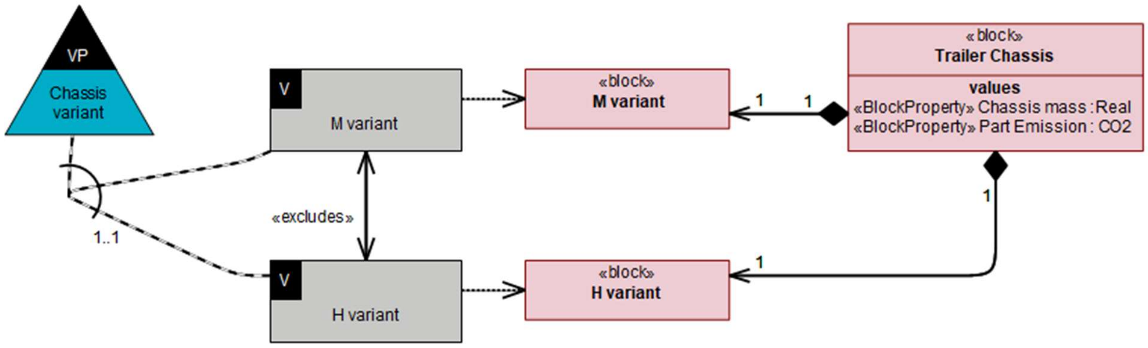

Figure 5 Model variability

3.2. Definition of LCA goal and scope

To effectively define the goal and scope, the imported LCA requirements from the general object library must be satisfied. These requirements are satisfied by the basic structural element used in SysML, a <<block>>. A block, which is a stereotyped extension of the UML class, contains the relevant LCA information and it is linked to other system elements as shown in Figure 6. One such example of how these blocks are utilized is the definition of the functional unit (Figure 6). By establishing the connection between the definition block and the system functions and top-level requirements we aim to improve the consistency of the functional unit and its alignment with the functions of the system of interest. Another example of this connection is the definition of the LCA scope. Since the environmental impacts of the selected product are concentrated in the manufacturing phase, the scope will be formed in a "Gate-to-Gate" manner which will include the manufacturing lifecycle phase of the physical subsystem (trailer chassis) and the assembly of the system. The system boundary will include the use cases from the implementation domain and the system elements from the solution domain. These system elements, which are a part of the manufacturing processes, are then linked to the definition blocks as shown in Figure 6. To further incorporate variability modelling into the LCA, we decided that the general purpose of this LCA example is to inform the customer stakeholders on the environmental impact of their chosen product configuration.

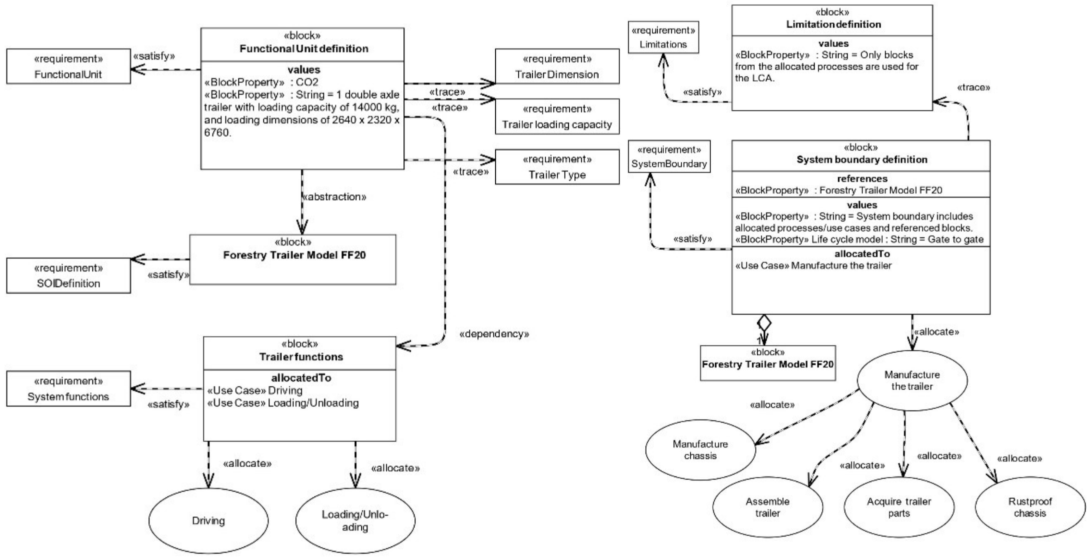

Figure 6 LCA requirements satisfaction

3.3. LCA Inventory analysis

The starting points for the inventory analysis are the production processes defined in the LCA boundary. These processes have been modeled as activity diagrams with activities allocated to blocks (Figure 7) that hold the relevant process information and parameters. Each block that represents a process must hold the corresponding LCA information for that particular process. This can be achieved by manually setting the value for the process based on available information or it can be calculated via parametric diagrams. The parametric diagram must include conversion values from an LCA database which are used for the calculation. For example, the CO_2 emissions of the laser cutting process (for the specific laser machine used) are parameterized and dependent on the input material (thickness and material) and the length of the cut which is defined in the CAD documentation.

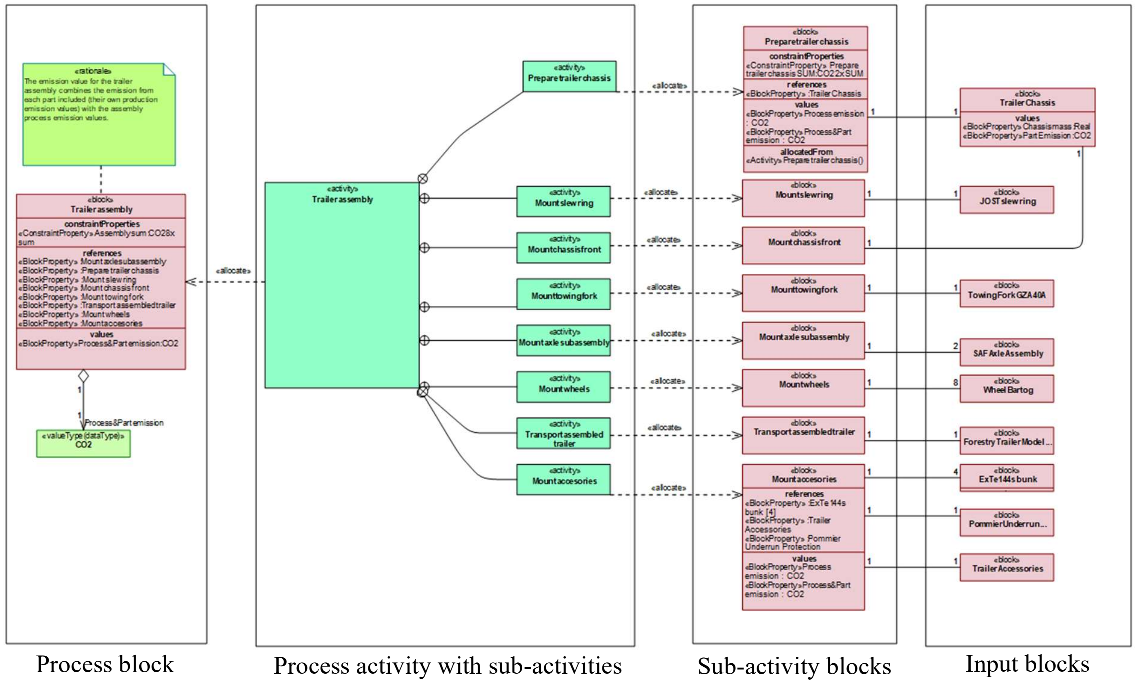

Figure 7 Assembly activities and blocks

The main part of the inventory analysis phase consists of modelling the necessary parametric diagrams and setting the default input values. This individual setting of parameters is necessary due to the lack of tool integration with CAD software and a LCA database. In this case study, we defined a SysML data type as a block value property for each system element. The chosen data type coincides with the LCA purpose (CO2 emission) and is defined in the lifecycle inventory analysis (LCIA) block definition diagram. It is also important to mention that the parametric diagrams are modelled considering the functional unit, in this case, the parametric diagrams calculate the relevant values for the production of one trailer. If this were not the case, some constraints would need to be added to enable the later unification of the calculated data. The constraint blocks which hold the necessary constraints (equations), together with the input and output parameters are imported from the LCA general object library to be used in the parametric diagrams for the LCIA. This approach enables the reusability of model elements and reduces modelling time and effort. An example of the parametric diagram for the "H variant" chassis which calculates the waste material mass and the emission recovery due to waste recycling is shown in Figure 8.

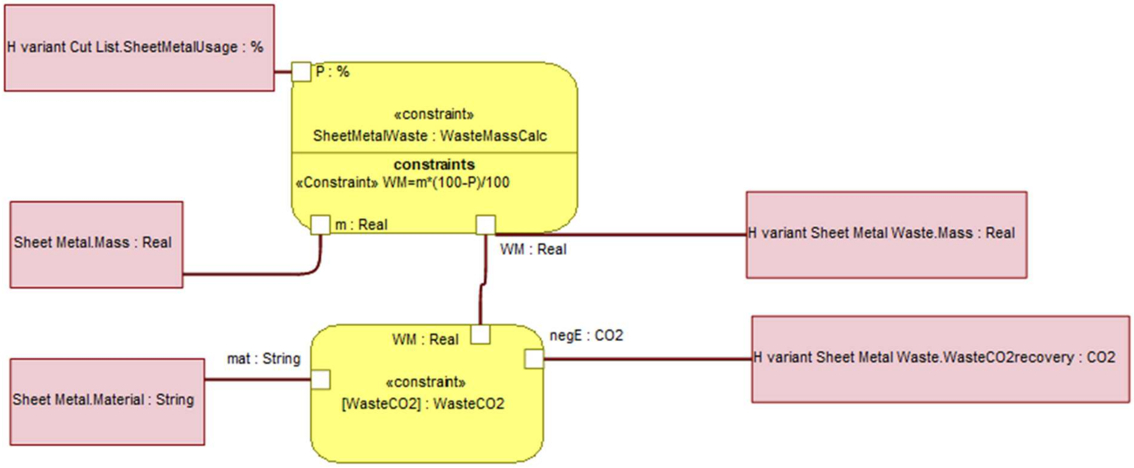

Figure 8 Parametric diagram for partial waste recycling

3.4. LCA Impact calculation and interpretation

The impact assessment for the chosen impact category is calculated for each process (module) with the corresponding parametric diagram created in the inventory analysis phase. The values are then summed up and displayed as a value property of the functional unit, which in this case corresponds to the top-level product model block "Forestry trailer".

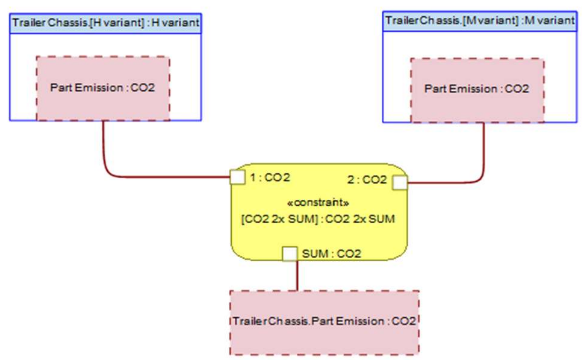

Figure 9 Parametric diagram with variable parts

The creation of product models from the product line model is enabled with the integration of OVM within the software tool. By choosing the desired variants in the variant selector (Figure 3) and executing the model variant generation option, the new product model is created. By generating this new product model, only the relevant blocks are included in the summarizing of the LCA values based on the structural links made in the block definition diagram (Figure 5). The example for a simple parametric diagram which calculates the emission of the chassis model based on the chosen variation point from Figure 5 is shown in Figure 9. After generating the product model, only the selected input from the chosen chassis variant remains. It is also important to mention that for product line model (Forestry trailer) the number of structural objects (blocks) is around cca. 200. It should be noticed that this is valid only for the partial model (see Table 1). For the complete model, the number of blocks will be significantly higher. Lastly, the method for the interpretation of the results is defined in the first phase of the LCA and will not be further detailed because it is out of scope of this paper.

4. DISCUSSION

Firstly, the answers to the research questions are provided and analyzed.

RQ1: *What are the benefits, limitations, and shortcomings of using LCA on a MBSE system model?*

Possible benefits:

- Reduced total time for LCA process modeling by using the already modelled manufacturing and use-phase processes, assuming that the system model has already been created for other purposes (besides LCA).
- Reduction of system boundary uncertainty by establishing connections between the use cases and LCA boundary definition block.
- Reduced modelling time due to reusability of objects from the LCA general object library.
- Early-stage insights into environmental considerations and a more representative LCA analysis for stakeholders which facilitates sustainability awareness.

Current limitations:

- Data uncertainty is not accounted for.
- LCA data availability.
- No LCA database integration.

Shortcomings:

- Extensive knowledge of the programming language (SysML or similar) and the system model structure are expected from the stakeholder to successfully conduct the LCA.
- The current tool capabilities require the modelling of production processes instead of importing partial or complete production process models from an integrated source.
- Overcrowding the system model elements with a large number of LCA values and parameters.
- The product line model complexity rises with the increase in number of variants, thus managing the whole model becomes very demanding.

RQ2: *How can the chosen modelling approach be upgraded to better facilitate the analysis of environmental impacts in MBSE system models?*

- One option is creating a package that expands the reference metamodel with new stereotypes. The proposed package (SysML profile extension) would include the LCA object library with a LCA database. However, the drawback of adding these new elements is that they may burden the already syntactically and semantically overloaded language which would require additional effort and knowledge from the system modeler.
- To minimize the cluttering of the model, the separation of LCA block property values from other system relevant values should be considered. This separation within the profile extension should then enable the stakeholder to sort, show or hide the desired value properties in the model.
- By expanding the profile to include the capability of producing a standardized LCA report form based on the LCA definition blocks (Figure 6).
- To help the LCA conducting stakeholders understand and navigate through the model, the structure of the LCA views (diagrams) should be defined with dedicated viewpoints.

The production processes considered in this case study were modelled based on existing production documentation which required additional effort and time from the system modeler. This could be enhanced with the combination of the proposed profile extension and the integration with different software tools. The activity-based object-oriented approach proposed in [30] demonstrates the addition of new classes for sustainability assessment which could, together with the integration of enterprise resource planning (ERP) (or a similar tool which manages the production process), greatly improve the feasibility of this approach. In the same way, a LCA database integration with an intuitive user interface is necessary for the successful implementation in industry.

While modelling the production process in the context of the LCA application, the structure of the processes needed to be taken into account. Unlike the modelling in LCA software tools (e.g. openLCA [31]) the modelled processes needed to be defined using use-case, activity and block definition diagrams as explained in section 3.1. This kind of process structure enables the analysis of each use case, like the approach proposed in System Lifecycle Management [16], for the purpose of identifying the process with the highest environmental impact.

Another aspect that needs to be considered when modelling is the data unification for the functional unit. In this case study, the production process resulted in one produced system which meant that the functional unit in the LCA corresponds to the whole modelled system. To better understand the data unification process from the LCA activity diagram (Figure 2), a different system should be modeled which would require additional parametric diagrams for the data unification. The proposed system should also have a different LCA goal and purpose, while also including other lifecycle phases which would help better understand the handling of other LCA uncertainties [19].

5. CONCLUSION AND OUTLOOK

The limitation of the presented case study is that the considered product (Forestry trailer) has a small environmental impact in exploitation. A system that has a greater impact in that phase of the life cycle should also be analyzed to get a better understanding of the proposed approach in the context of the whole system lifecycle. Similar to other research, which is currently conducted in the field, further case studies on this topic should contribute to the development of a modelling approach by specifying the modelling guidelines. We recognize that for the creation of such modelling guidelines, a significant amount of time and effort is needed, both for the definition and the validation. Furthermore, it is important to consider the challenges associated with the integration of MBSE and LCA in practice. The successful MBSE-LCA integration is predominantly reliant on the development state of software tools. This, together with the lack of integration of MBSE and other tools is a major obstacle for the satisfying implementation of this approach. With the current state of research it is difficult to estimate in which cases the return of investment for the LCA-MBSE integration would be favorable because the implementation process of MBSE is a very resource intensive process. The benefits of this approach could be more attainable in organizations with a well-established MBSE implementation on a nearly complete system model.

The introduction of new Lifecycle classes for key concepts of the lifecycle phases (in the proposed SysML profile) could improve stakeholder understanding and therefore increase productivity. The proposed new classes should own partially complete parametric diagrams for the calculation of key lifecycle parameters, while a link to an LCA database for consistent data extraction could further enhance this concept. For future work, together with the proposed SysML profile extension, the quantitive LCA uncertainties should be addressed with the help of a probability distribution and a SysML capability for a Monte Carlo simulation should be explored in this context.

Acknowledgements

The authors would like to express their sincere gratitude to EAG Center Technologies, Zagreb, Croatia, for their valuable contribution in providing the necessary software licenses for the completion of this research.

CONTACTS

Zvonimir Lipšinić email: zvonimir.lipsinic@fsb.hr
ORCID: https://orcid.org/0000-0002-0920-8495

Prof. dr. sc. Neven Pavković email: neven.pavkovic@fsb.hr
ORCID: https://orcid.org/0000-0002-1291-0429

6. References

[1] Z. Li, J. Lu, G. Wang, L. Feng, D. G. Broo, and D. Kiritsis, “A Bibliometric Analysis on Model-based Systems Engineering” *ISSE 2021 - 7th IEEE International Symposium on Systems Engineering, Proceedings*, Sep. 2021, doi: 10.1109/ISSE51541.2021.9582526.

[2] J. D’ambrosio *et al.*, “An mbse approach for development of resilient automated automotive systems” *Systems*, vol. 7, no. 1, Mar. 2019, doi: 10.3390/SYSTEMS7010001.

[3] J. A. Martínez Rojas, J. L. Fernández, R. S. Montero, P. L. L. Espí, and E. Diez-Jimenez, “Model-based systems engineering applied to trade-off analysis of wireless power transfer technologies for implanted biomedical microdevices” *Sensors*, vol. 21, no. 9, May 2021, doi: 10.3390/S21093201.

[4] A. M. Madni and S. Purohit, “Economic analysis of model-based systems engineering” *Systems*, vol. 7, no. 1, Mar. 2019, doi: 10.3390/SYSTEMS7010012.

[5] M. L. Valdivia Dabringer, A. Dybov, C. Fresemann, and R. Stark, “Towards Integrated Safety Analysis as Part of Traceable Model-Based Systems Engineering” *Proceedings of the Design Society*, vol. 2, pp. 2005–2014, May 2022, doi: 10.1017/PDS.2022.203.

[6] E. R. Carroll and R. J. Malins, “Systematic Literature Review: How is Model-Based Systems Engineering Justified?” Mar. 2016, doi: 10.2172/1561164.

[7] S. Husung, C. Weber, A. Mahboob, and S. Kleiner, “Using model-based systems engineering for need-based and consistent support of the design process” *Proceedings of the Design Society*, vol. 1, pp. 3369–3378, 2021, doi: 10.1017/PDS.2021.598.

[8] S. Bougain and D. Gerhard, “Integrating Environmental Impacts with SysML in MBSE Methods” *Procedia CIRP*, vol. 61, pp. 715–720, Jan. 2017, doi: 10.1016/J.PROCIR.2016.11.196.

[9] E. Igos, E. Benetto, R. Meyer, P. Baustert, and B. Othoniel, “How to treat uncertainties in life cycle assessment studies?” *International Journal of Life Cycle Assessment*, vol. 24, no. 4, pp. 794–807, Apr. 2019, doi: 10.1007/S11367-018-1477-1/TABLES/3.

[10] D. Collado-Ruiz and H. Ostad-Ahmad-Ghorabi, “Comparing LCA results out of competing products: developing reference ranges from a product family approach,” *J Clean Prod*, vol. 18, no. 4, pp. 355–364, Mar. 2010, doi: 10.1016/J.JCLEPRO.2009.11.003.

[11] R. A. Colletti, A. Qamar, S. P. Nuesch, and C. J. J. Paredis, “Best Practice Patterns for Variant Modeling of Activities in Model-Based Systems Engineering,” *IEEE Syst J*, vol. 14, no. 3, pp. 4165–4175, Sep. 2020, doi: 10.1109/JSYST.2019.2939246.

[12] M. Kwak and H. Kim, “Exploring Opportunities to Improve Life Cycle Environmental Performance of a Complex Product,” pp. 735–744, 2013, doi: 10.1007/978-3-642-30817-8_72.

[13] C. Laing, P. David, E. Blanco, and X. Dorel, “Questioning integration of verification in model-based systems engineering: an industrial perspective” *Comput Ind*, vol. 114, Jan. 2020, doi: 10.1016/j.compind.2019.103163.

[14] A. Mahboob and S. Husung, “A Modelling Method for Describing and Facilitating the Reuse of SysML Models during Design Process” *Proceedings of the Design Society*, vol. 2, pp. 1925–1934, May 2022, doi: 10.1017/PDS.2022.195.

[15] S. Bougain and D. Gerhard, “A CBR approach for supporting ecodesign with SysML” 2017.

[16] M. Eigner, T. Dickopf, and H. Apostolov, “System Lifecycle Management - An Approach for Developing Cybertronic Systems in Consideration of Sustainability Aspects” *Procedia CIRP*, vol. 61, pp. 128–133, 2017, doi: 10.1016/J.PROCIR.2016.11.210.

[17] M. Eigner and M. T. Zavareh, “Digitalization of the Engineering Supported by System Lifecycle Management (SysLM)” *IFIP Adv Inf Commun Technol*, vol. 640 IFIP, pp. 240–254, 2022, doi: 10.1007/978-3-030-94399-8_18.

[18] K. Azevedo, B. Bras, S. Doshi, and T. Guldberg, “Modeling Sustainability of Complex Systems: A Multi-Scale Framework Using SysML” *Proceedings of the ASME Design Engineering Technical Conference*, vol. 2, no. PARTS A AND B, pp. 1437–1448, Jul. 2010, doi: 10.1115/DETC2009-87496.

[19] D. Inkermann, “Potentials of integrating MBSE and LCA to handle uncertainties and variants in early design stages” in *DS 119: Proceedings of the 33rd Symposium Design for X (DFX2022)*, The Design Society, Oct. 2022, pp. 10–10. doi: 10.35199/dfx2022.19.

[20] G. Biggs, T. Sakamoto, and T. Kotoku, “A profile and tool for modelling safety information with design information in SysML” *Softw Syst Model*, vol. 15, no. 1, pp. 147–178, Feb. 2016, doi: 10.1007/S10270-014-0400-X/TABLES/7.

[21] Y. Romaniw, B. Bras, and T. Guldberg, “Sustainable manufacturing analysis using an activity based object oriented method” *SAE International Journal of Aerospace*, vol. 2, no. 1, pp. 214–224, 2010, doi: 10.4271/2009-01-3229.

[22] D. Dori, “Model-based systems engineering with OPM and SysML,” *Model-Based Systems Engineering with OPM and SysML*” pp. 1–411, Jun. 2016, doi: 10.1007/978-1-4939-3295-5/COVER.

[23] S. Friedenthal, A. Moore, and R. Steiner, “A Practical Guide to SysML: The Systems Modeling Language, Third Edition” *A Practical Guide to SysML: The Systems Modeling Language, Third Edition*, pp. 1–606, Jan. 2014, doi: 10.1016/C2013-0-14457-1.

[24] A. Morkevicius, A. Aleksandraviciene, D. Mazeika, L. Bisikirskiene, and Z. Strolia, “MBSE Grid: A Simplified SysML-Based Approach for Modeling Complex Systems” *INCOSE International Symposium*, vol. 27, no. 1, pp. 136–150, Jul. 2017, doi: 10.1002/J.2334-5837.2017.00350.X.

[25] A. Morkevicius, A. Aleksandraviciene, A. Armonas, and G. Fanmuy, “Towards a Common Systems Engineering Methodology to Cover a Complete System Development Process” *INCOSE International Symposium*, vol. 30, no. 1, pp. 138–152, Jul. 2020, doi: 10.1002/J.2334-5837.2020.00713.X.

[26] M. Li, B. Firat, G. Li, Grigg. Alan, M. Ingham, and P. Bull, “Model-Based Systems Engineering with Requirements Variability for Embedded Real-Time Systems,” *MoDRE*, 2015

[27] D. Bilic, D. Sundmark, W. Afzal, P. Wallin, A. Causevic, and C. Amlinger, “Model-based product line engineering in an industrial automotive context,” pp. 56–63, Sep. 2018, doi: 10.1145/3236405.3237200.

[28] K. Pohl, G. Böckle, and F. Van Der Linden, “Software product line engineering: Foundations, principles, and techniques” *Software Product Line Engineering: Foundations, Principles, and Techniques*, pp. 1–467, 2005, doi: 10.1007/3-540-28901-1/COVER.

[29] M. Hause and J. Hummell, "Model-based Product Line Engineering -- Enabling Product Families with Variants" *INCOSE International Symposium*, vol. 25, no. 1, pp. 1320–1332, Oct. 2015, doi: 10.1002/J.2334-5837.2015.00132.X.

[30] Y. Romaniw, B. Bras, and T. Guldberg, "An Activity Based Approach to Sustainability Assessments," *Proceedings of the ASME Design Engineering Technical Conference*, vol. 2, no. PARTS A AND B, pp. 1131–1139, Jul. 2010, doi: 10.1115/DETC2009-87396.

[31] Ciroth A, C. Di Noi, Lohse T, and Srocka M, *Comprehensive openLCA manual for version 1.10.,* 2020.

