System interaction, System of Systems, and environmental impact of products

Abdoli, Shiva; Kara, Sami; Hauschild, Michael Zwicky

Published in:
CIRP annals

Link to article, DOI:
[10.1016/j.cirp.2019.04.051](https://doi.org/10.1016/j.cirp.2019.04.051)

Publication date:
2019

Document Version
Peer reviewed version

[Link back to DTU Orbit](#)

Citation (APA):
Abdoli, S., Kara, S., & Hauschild, M. Z. (2019). System interaction, System of Systems, and environmental impact of products. *CIRP annals*, 68(1), 17-20. <https://doi.org/10.1016/j.cirp.2019.04.051>

General rights

Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

- Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
- You may not further distribute the material or use it for any profit-making activity or commercial gain
- You may freely distribute the URL identifying the publication in the public portal

If you believe that this document breaches copyright please contact us providing details, and we will remove access to the work immediately and investigate your claim.

Please cite this article as: Shiva Abdoli, Sami Kara and Michael Hauschild (2019). System interaction, System of Systems, and environmental impact of products. *CIRP Annals - Manufacturing Technology* 68, 17–20.

System interaction, System of Systems, and environmental impact of products

Shiva Abdoli ^{a,b}, Sami Kara (1)^{a,b,*}, Michael Hauschild (2)^{a,b}

^a Sustainability in Manufacturing & Life Cycle Engineering Research Group, School of Mechanical & Manufacturing Engineering, The University of New South Wales, Sydney, NSW 2052 Australia

^b Division of Quantitative Sustainability Assessment, Department of Manufacturing Engineering and Management, Technical University of Denmark, Produktionstorvet Building 424, Kongens Lyngby, DK-2800, Denmark 1.

* Corresponding author at: Sustainability in Manufacturing & Life Cycle Engineering Research Group, School of Mechanical & Manufacturing Engineering, The University of New South Wales, Sydney, NSW 2052 Australia. E-mail address: S.Kara@unsw.edu.au (S. Kara).

Abstract

Product-systems interact with their surrounding systems and together form a System-of-Systems (SoS). The interactions lead to emerging behaviour from the SoS, including rebound effects that influence the actual environmental impact of changes in the product-systems. Accordingly, the SoS impact can be higher than the sum of impacts from its interacting product-systems. This paper proposes a modelling approach for life cycle assessment of a product-system considering emerging behaviours in its SoS context beyond what is normally considered in a consequential LCA. The applicability of the proposed approach is demonstrated on a transportation system case study.

Keywords: Environment Life cycle System of Systems

1. Introduction

Generally, environmental impact of a product-system (PS) is considered as a function of the PS, which is considered as a foreground system drawing on a background system. However, a PS may interact with its surroundings (besides using background systems) to realize its functionality. A PS, together with other involved PSs, form a network. Gaia Theory [1] explains that all living organisms interact with their surroundings, forming of complex systems. In the engineering context, such networks or complex systems are called System of Systems (SoS). The interactions in an SoS may affect individual PSs, leading to unexpected behaviour in SoS, which is called emerging behaviour. As a result, the environmental impact of an SoS can be different from the sum of the environmental impacts of the individual PSs when the effect of interactions is not taken into account. Hence, it may be necessary to study a PS within its SoS to properly understand its environmental impact by means of Life Cycle Assessment (LCA).

According to the law of unintended consequences [2], a change in a PS may lead to unforeseen consequences in its SoS, which is called ‘rebound effect’. For instance, in the energy efficiency context, rebound effect is defined as the difference between expected energy savings and actual savings. Such unintended consequences may offset the actual end outcome of a change in a PS from the expected outcome in terms of reducing the environmental impact. Although, consequential LCA (cLCA) [3] addresses the consequences that decisions may have on other systems, the modelling of interactions between PSs offers a more comprehensive understanding of potential environmental impact at the SoS level [4]. This paper introduces a modelling framework to study the environmental impact of a PS within its SoS from an LCA perspective. The emphasis is put on the scope definition phase, which determines how a PS may be studied as part of an SoS. A car PS within its transportation-related SoS is used as a case to demonstrate the use of the framework.

Nomenclature

ac–dc	Acceleration–declaration
CS	Constituent system
DP	Dynamic parameter
EI	Environmental impact
ES	Enabling system
FS	Forming system
FT	Transition function
IP	Independent parameter
LCA	Life cycle assessment
NDP	No. of DPs
NIP	No. of IPs
OO	Object oriented
PS	Product system
SE	Systems engineering
SoS	System of Systems

2. Methodology

The framework consists of two-step methodology described in the ensuing sections: first, determining the SoS boundary, second, modelling the SoS emerging behaviour. The first step includes three iterative steps: classification of participating systems based on their characteristics, identification of systems that interact with PS, inclusion of those systems that form an SoS with PS.

2.1. Determining the SoS boundary: modelling the SoS architecture

SoS is defined as an integration of Constituent Systems (CSs) that are independent, operable, and have their own stakeholders. CSs are networked together to achieve a higher capability that may not be achieved with individual CSs [5]. They can have different independency levels and are heterogenic.

Analysing a PS requires a model that embodies the characteristics of its SoS. Multi-disciplinary and complexity are the most agreed challenges in modelling the SoSs. Hence, many studies focused on modelling the structural aspects of an SoS. However modelling the SoSs in a holistic manner, capturing both structural aspects and interactions, allows having a better understanding regarding the SoS emerging behaviour from an LCA perspective [6]. We use a Systems Engineering (SE) approach to provide a foundation for modelling of SoSs because SE is a multi-disciplinary approach and SoSs are heterogenic. In modelling context, system architecture embodies a set of stable concepts and semantic relations to realize the achievement of system function in terms of structure and behaviour. In a SE approach, the system architecture is developed by decomposing a system to logical subsystems and developing their interaction interface. To model the SoS architecture, we use an Object Oriented (OO) approach to manage the complexity of SoSs by using two of its main principles; abstraction and encapsulation. Abstraction seeks to decompose a system to its essential elements without addressing their details such that the elements clarify the system boundary. An SoS can be defined by abstracting the SoS to its participating CSs, which are its essential elements for application of LCA on a PS. The CSs of an SoS are classified to three classes; Product System(PS), Enabling-System(ES), and Forming-System(FS), (Fig.1). ESs and FSs are different from background-foreground systems as they would be defined in an LCA of a PS without having the SoS perspective. Foregrounds include those processes that are specific to the PS, and it is possible to change them. Background processes are not specific to the PS, but a practitioner can include the average industry data representing the local background processes [7]. In fact, foreground-background systems define the PS boundary as an individual CS whereas; ES, FS, PS together define the SoS boundary that the PS functions in. PS, ESs, and FS (all as CSs) can have different levels of operational independency in the SoS. The first class addresses the PS subjected for the LCA (e.g. a car). ES class (an abstract class) abstracts those CSs that the PS needs to interact with to fulfil its function. For instance, a car needs roads as infrastructure. FS class (an abstract class) abstracts those CSs that the PS does not necessarily need to interact with, but that exists in the surroundings of the PS and may affect it, such as regulations. A PS may interact with several ESs and FSs that differ from several aspects. For instance, roads and fuel stations are two ESs that a car interacts with, but they function differently. From OO perspective, classes can be derived from abstract classes of ES and FS and each class can have its own properties. Therefore, one PS and several ESs and FSs are defined, having their own properties.

Fig. 1. Environmental impact of systems at system and SoS levels.

The explained approach allows embodying different types of ESs and FSs in the SoS architecture and accordingly capturing a heterogenic SoS holistically. Due to the exemplification on a car, this paper emphasizes the use stage, since the PS (car) delivers its function during the use stage. However in the production stage, the production facilities that produce the PS form another SoS. Likewise in the end of life stage, the recycling facilities and other required activities (e.g., logistics) form another SoS. Hence, the interactions are not limited to the use stage of an active PS. Instead, a PS moves through multiple SoSs in its life cycle and a proper application of LCA on a PS needs to be carried out in a multi-SoS perspective. All SoSs in different life cycle stages of a PS use resources and have environmental impact (EI) that should be considered in the LCA. The interactions may lead to improving the environmental performance of one CS but worsening in another CS, hence it is critical to take into account the interactions in the study (Fig. 1).

In the OO context, encapsulation seeks to hide internal details of the system elements (subsystems in the SE context) to manage the complexity when the system scale increases. In return, although the subsystems interact, they do not need to know how others perform their internal procedures. Two types of mutually exclusive parameters are defined for CSs, independent (IPs) and dynamic parameters (DPs). The first type of parameters defines the structure of a CS_{cs} (an arbitrary CS) and are the results of its development plans; $IP_{nip;cs}$; $nip \in NIP$. The set of IPs do not change in a certain time period. A set of $DP_{ndp;cs}$; $ndp \in NDP$ addresses the parameters that capture the dynamic behaviour of a CS_{cs} . NIP and NDP are the total number of defined IPs and DPs, respectively. IPs and DPs are modelled as attributes and functions of the CS_{cs} classes. $DP_{ndp;cs}$ of a CS_{cs} can take into account IPs and DPs from the CS_{cs} class itself or from other CSs. This forms a network of functions, embodying the interactions between CSs. Encapsulation allows hiding the formulation of DPs of CSs and managing the SoS complexity. $DP_{ndp;cs}$ and $IP_{nip;cs}$ of a CS_{cs} should address all the parameters that influence its LCA. For instance, some IPs are defined related to fuel consumption rate, while some DPs are defined related to acceleration–deceleration (ac–dc) frequency, because they impact fuel consumption. From Gaia theory perspective, everything is connected, so a PS can be connected to many other systems. Hence, a challenging issue is to define a cut off for the inclusion of FSs and ESs in an SoS such that its boundary is manageable for analysis. We suggest analysing the DPs and IPs of the PS to identify and include those CSs that directly impact DPs and IPS of the PS, see Fig. 2. Therefore, although the FS_2 impacts the PS, since it is an indirect impact or second tier connection, FS_2 is not included in the SoS boundary. This is an iterative approach that helps to identify/classify the potential CSs to be included in an SoS boundary and then cut them down to a manageable set.

Fig. 2. Cut off approach for defining SoS boundary.

2.2. Rebound effect and emerging behaviour

Direct rebound effect is defined as consumption (demand) increase of a product due to improvement in its efficiency. For instance, a more fuel-efficient car reduces the ownership cost, may be leading to more travel. Accordingly, if a 10% improvement in fuel consumption efficiency only leads to a 6% consumption reduction in the car's lifetime, the rebound effect is 40%. An indirect rebound effect is caused by demand increase in other goods due to the improved energy efficiency of a product. Rebound effect is mostly studied with demand elasticity, which describes how the demand of a PS responds to its price change [8]. Rebound effect can exceed 100%, meaning that the improvement in a PS leads to worsening the situation. Rebound effect is a subset of emerging behaviour, where the interactions between CSs hinder the achievement of improving environmental impact at the SoS level. The introduced perspective allows studying the rebound effect more broadly in a structured manner. First, environmental impacts are not limited to consumptions (e.g. energy) and may have other forms, such as emissions of hazardous substances. Second, rebound effect is mostly studied as how a user behavior leads to more consumption. Yet, the introduced approach defines FSs and ESs beside the user that also may react to the demand increase in different ways. This perspective allows studying the rebound effect by analysing questions such as; how other CSs react to demand elasticity? How such reactions may affect other CSs? How the environmental footprint of an SoS is altered due to such reactions? For instance, how road as a system may react to the increase in car travels and what is the impact of the road system reaction on other CSs and environment? This is also a point that the introduced approach expands cLCA, where a change in one CS may lead to a chain of interconnected-unforeseen consequences that are needed to be fully traced.

2.3. Modelling the emerging behaviour: state base modelling of SoS

It may take time for CSs to interact with their surroundings, hence the chain of reactions leads to emergent behaviour. To model the chain of CSs' reactions on each other, the study time period (T) is split into a number of time intervals (n). Each CS_{cs} class has a finite number of states (K) that can transit between in a time interval (time_i; $i \in n$) as shown in Fig. 3. If LCA is applied without an SoS perspective, it is assumed that the PS and its surrounding (CSs) stay in the initial state for the entire T. However, a CS_{cs} can transit to other states (evolved state_{cs,k}) as a result of reacting to its surrounding. If in the beginning of time_i, a CS_{cs} transits from its previous state in time_{i-1} (previous state_{cs,i-1,k}) to evolved state_{cs,k},

then this becomes the current state $_{cs,k,i}$ in time $_i$. For instance, increase in cars' travel may lead to higher congestion in roads in time $_{i-1}$, so an evolved state for the road in time $_i$ can be 'road-expansion'.

In a CS_{cs} , the transition to a new evolved state $_{cs,k}$ happens when there is a need to change the $IP_{nip;cs}$. In fact each evolved state $_{cs,k}$ has its own specific values for $IP_{nip;cs}$. For instance, if the road transits to road-expansion state, the capacity of roads increases. Hence, as soon as a CS_{cs} transits to an evolved state $_{cs,k}$ in the beginning of time $_i$, then $IP_{ip,c,i}$ are updated accordingly.

The DPs are determined at the end of time $_i$ and can be a function of several $IP_{nip,cs,i}$, see Eq. (1). For each CS_{cs} , the existing empirical studies or models can be used to estimate its DPs according to the existing IPs. If the demand increases 20% and no new roads are added, the traffic increases in a range of (0–20%).

A transition function (FT_{cs}) decides about state transition in a CS_{cs} . The user can define the relation between CSs by addressing IPs and DPs in FTs. In time $_i$, the set of $\{IP_{nip,cs,i-1}, DP_{ndp,cs,i-1} \forall cs \in CS, \forall nip \in NIP, \forall ndp \in NDP\}$ have their values from time $_{i-1}$. This set is called priori set. The user defines what the first CS_{cs} is (e.g. CS_{cs1}) and its FT_{cs} takes the priori set to decide about CS_{cs} evolvement (determining $IP_{nip1,cs1,i}$). The determined $IP_{nip1;cs1,i}$ is allocated to a posterior set. The next CS_{cs} (e.g. CS_{cs2}) has access to priori and posterior set. Hence, FT_{cs2} decides about CS_{cs2} evolvement by using both sets, see Eq. (2). This continues until to the last CS_{cs} . The circles in Fig. 3 show the sequence of updating the $IP_{nip,cs,i}$ set. Each state may have a capacity to absorb the dynamics in its surroundings, so staying in previous state $_{cs,i-1}$ and not evolving to a new state in time $_i$. A CS_{cs} reaction to its surroundings depends on its stakeholders' behaviour in making decisions regarding development plans. The FT_{cs} is modelled as a decision tree, imitating the decision making in a CS_{cs} . For instance, if traffic was at least 0.0005% higher than last year and roads' age is more than 20 years, then the road evolves to road-expansion state. In this paper, the decision tree is used because it fits well to the high-level decision making when data are nonlinear and a mix of quantitative and qualitative.

Fig. 3. State-based modelling of SoS.

As shown in Fig. 3, in the proposed approach each CS has its own state-base sub-model that allows embodying different levels of modelling details depending on the data availability, while the sub-models still exchange IPs-DPs at time_i. Hence, this approach can handle issues such as different temporal scales between CSs or their complexity. The proposed approach also allows modelling users' behaviours with regards to cultural issues. In this case, the sub-model of the user can include states with values for demand increase to demonstrate the user behaviour within a society.

Such a state-based model of SoS shows how it may evolve by modelling the evolvement of its CSs during study time. An SoS may also evolve by introduction of a new CS; CS_{new}. In this case, a class is equivalent to CS_{new} with its IP_{new} and DP_{new}. The connections of CS_{new} with existing CSs are captured by defining relations between the parameters (IP_{new}, DP_{new}; IP_{cs}, DP_{cs}) in FTs.

$$DP_{ndp,cs,i} = f(\{IP_{nip,cs,i} \forall cs \in CS, \forall nip \in NIP, \}) \quad (1)$$

$$\text{Evolved state}_{cs,k} = FT_{cs}(\{\text{priori set}\}, \{\text{posterior set}\}) \quad (2)$$

2.4. Impact assessment

Environmental Impact (EI) of each CS_{cs} (PS, FS, ES) is calculated using standard LCA approaches based on the Life Cycle Inventory of time_i, LCI_{cs,i}, developed by taking into account the IP_{nip,cs,i} and DP_{ndp,cs,i}. EI of the SoS, Elsos, is calculated as given in Eq. (3).

$$\text{Elsos} = \sum_{i=1}^n \sum_{cs=1}^{CS} EI_{cs,i}(LCI_{cs,i}) \quad (3)$$

For instance, a total driving distance with a certain frequency of ac–dc (as a function of road congestion) is used to calculate impact in time_i. Generally, LCA results are subject to data uncertainty and lack of data, hence their impact in estimating DPs on EI_{cs,i} can be tested by performing Monte Carlo analysis. Likewise, the impact of DPs at the SoS level can be analyzed by variance-based sensitivity analysis (Sobol indices). Choice-related uncertainties concerning decision making in CSs (their evolvement) can be modelled by configuring some extreme behavioural scenarios for the SoS and observe their magnitude in changing the LCA results.

3. Demonstration case

A city car is considered as the PS. New technology in the car reduces its fuel consumption rate. The functional unit is defined as ten years of car driving by a user. ESs classes are defined as; user, road, fuel station, and tire. The regulation subclass is defined as an FS. LCA results from Refs. [9–11] are used respectively for the tire, road, and fuel station. The study period is divided into three time steps. An initial state is defined for CSs according to the assumption in the car LCA report [12]. IPs and DPs are defined for CSs by studying their LCA reports. For instance, the road has IPs related to material use per km, while traffic congestion is defined as a DP, affecting its environmental impact due to the needed maintenance. It is assumed that the traffic can impact maintenance around 10%. However, the tire consumption is a direct function of the car travel distance [9].

The evolved states for the user are defined as changes in the travel demand with increments of 6%, 23%, 30%, and 37% with respect to the initial demand. Likewise, roads and fuel station expansions are

defined as evolved states for road and fuel station systems. These evolved states address building new roads and fuel stations to keep the traffic and average queue time in fuel station similar to the initial state according to the existing length of national roads and number of petrol stations [13,14]. It is assumed that the market share of the car is 0.35%; this number is kept low to assure that the results are not biased in supporting the paper objectives. According to the existing number of cars and sales volume in 2017 [14], the new car sale increases traffic by 0.0027%. However, road congestion or fuel station queue will not increase proportionally, since they are not on 100% utilization. It is assumed that they increase with a factor of 5% (e.g. the traffic increases 0.000135% due to this car's sale). Fig. 4 shows the case study results. 'Basic' line demonstrates LCA results without using the proposed SoS approach. The results include the LCA of the car plus its tires according to 15,000 km travel distance annum for ten years use stage, disregarding travel increase and its consequences. Change in travel demand due to having a more fuel-efficient car is estimated 6%–100% in different countries for short and long runs [8]. In fact, it is uncertain how the defined CSs may evolve and as a result how their interactions affect the total demand increase in the ten years study period. Hence, to address the uncertainties in CSs evolvment, three scenarios about emerging behaviour of the SoS are configured as follow; 'Absorb dynamic': all the CSs (except the user) stay in initial state for the entire 10 years and the demand increase only affects their DPs (not IPs). 'Postpone evolving': road and fuel station stay in the initial state until end of time₂ and they evolve to the expansion states in time₃. 'Evolve proactively': road and fuel station evolve to expansion states in time₂. This is the outcome of the decision tree models in the CSs.

Fig. 4. Comparison of different emerging behaviours of the car transportation SoS with the common LCA without SoS perspective.

Generally, demand increase happens gradually. Hence, in time₁ the user evolves to '6% demand increase' state. This increases LCA values related to use stage of road, fuel station, and tires in all three scenarios and shifts their LCA values higher than 'basic'. However, if SoS evolves proactively or with a postponement, then building new roads/fuel stations leads to an increase in LCA values in respectively time₂ and time₃. Under 'absorb dynamic' behaviour, the increased road congestion also affects the car average fuel consumption due to more needed ac-dc as a function of traffic. Although it is not addressed in this paper, more congestion can also shorten the tire lifetime. Under the 'evolve proactively' behaviour, although the road expansion in time₂ keeps the car average consumption similar to the initial state, the reduced congestion combined with a more fuel-efficient car can lead to more car driving, which is called induced traffic. Hence in 'evolve proactively' behaviour in time₃, the user evolves to 'demand increase of 37%', whereas it evolves to '30% demand increase' in 'postpone evolving' behaviour. In fact, as a result of interactions between the defined CSs, the total demand increase in the

ten years study period is 20% for the 'postpone evolving' and 'absorb dynamic' behaviours, whereas the total demand increase is 23% for the 'evolve proactively'. Although the roads' expansion happens in both 'Evolve proactively' and 'Postpone evolving' scenarios, the total impact of roads is higher in the 'evolve proactively' because of the extra needed maintenance due to induced traffic. This illustrates how the introduced approach captures the impact of CSs evolution on each other. Although the results are subject to uncertainty, all behaviour scenarios show higher environmental impact when the car PS is studied using the SoS approach compared to 'basic' LCA. Rebound effect can also have an impact on resource consumption due to producing more energy efficient products. This may lead to sales volume increase due to the availability of more fuel efficient cars, hence the automotive industry may produce more fuel-efficient cars. In this context, a manufacturer can be defined as a CS and the impacts of a more fuel-efficient car can be studied in an even broader scope.

4. Discussion and conclusion

This paper has introduced a framework for analysing the environmental impact of a product in a SoS perspective by integrating several methods for the purpose of mitigating the risk of shifting the environmental impact from a product itself to its surroundings or overlooking unintentional consequences that the changes in the product have on its surroundings as a function of the rebound effect. Quantification of the rebound effect requires considerable simplification; particularly in LCA context, the results are subject to uncertainty due to the existence of numerous influencing parameters and the lack of reliable data sources, where a broad study scope can further complicate the analysis. However, the approach given in the paper allows the user to estimate the possible magnitude of the environmental impact of a product in the SoS context. In the case study, the regulation as an FS, can thus also evolve by introducing new legislation to reduce the negative impacts of the increase in travel demand, such as introducing emission trading rules or promoting public transportation. The insights offered by this approach can help decision makers ensure that the improved environmental performance of a product will not be offset by rebound effects due to product interactions with its surroundings. In the process of selecting between two technologies, they can thus be compared in a broader and more realistic perspective; which product with its interacting infrastructure, policy, and equipment together can lead to better performance?

References

- [1] Volk T (2002) Toward a Future for Gaia Theory. *Climatic Change* 52(4):423–430.
- [2] Huesemann M, Huesemann J (2011) *Techno-Fix: Why Technology Won't Save Us or the Environment*, New Society Publishers.
- [3] Sandén BA, Karlström M (2007) Positive and Negative Feedback in Consequential Life-cycle Assessment. *Journal of Cleaner Production* 15(15):1469–1481.

- [4] Little JC, Hester ET, Elsayah S, Filz GM, Sandu A, Carey CC, Iwanaga T, Jakeman AJ (2019) A Tiered, System-of-Systems Modeling Framework for Resolving Complex Socio-environmental Policy Issues. *Environmental Modelling and Software* 112:82–94.
- [5] Jamshidi M (2008) System of Systems — Innovations for 21st Century. 2008 IEEE Region 10 and the Third international Conference on Industrial and Information Systems, 6–7.
- [6] Yang Y, Heijungs R (2019) Moving from Completing System Boundaries to More Realistic Modeling of the Economy in Life Cycle Assessment. *The International Journal of Life Cycle Assessment* 24(2):211–218.
- [7] Hauschild MZ, Rosenbaum RK, Olsen S (2018) *Life Cycle Assessment*, Springer.
- [8] Jägerbrand AK, Dickinson J, Mellin A, Viklund M, Dahlberg S (2014) Rebound Effects of Energy Efficiency Measures in the Transport Sector in Sweden, Statens väg-och transportforskningsinstitut.
- [9] Krömer S, Eckhard K, Reichenbach D, Rainer S (1999) Life Cycle Assessment of a Car Tire Continental, Hannover, Germany .
- [10] Strippel H (2001) Life Cycle Assessment of Road: a Pilot Study for Inventory Analysis. IVL RAPPORT (1210)96.
- [11] Junnila S, Horvath A, Guggemos AA (2006) Life-cycle Assessment of Office Buildings in Europe and the United States. *Journal of Infrastructure systems* 12 (1):10–17.
- [12] Samaras C, Meisterling K (2008) Life Cycle Assessment of Greenhouse Gas Emissions From Plug-in Hybrid Vehicles: Implications for Policy. *Environmental Science and Technology* 42:3170–3176.
- [13] Official Statistics of Sweden (2018) Vehicle Statistics 2017 . Accessed 20 December 2018<https://www.trafa.se/en/road-traffic/vehicle-statistics/>.
- [14] BIL Sweden© Statista (2018) Number of Passenger Cars Sold in Sweden from 2008 to 2017 . Accessed 30 January 2019<https://www.statista.com/statistics/424949/passenger-car-sales-in-sweden/>.