INTERNATIONAL COUNCIL
ON COMBUSTION ENGINES

05 | 2023

# CIMAC Guideline

## Virtual System Integration & Simulation

A Performance-oriented Approach for Guiding System Simulation in the Field of Hybrid Marine Applications

From CIMAC WG20 System Integration

The first edition of this CIMAC Guideline was approved by the members of the CIMAC WG 20 (System Integration) at its meeting on March 15, 2023.

# Content

# 1 Introduction

Since virtual methods and model-based approaches are gaining more and more importance, the subgroup "Tools" was founded within WG20 (System Integration). The subgroup initially addressed issues related to the use of simulation tools and later with further aspects of virtual system integration. In the corresponding working meetings, a wide range of topics related to the simulation of marine applications with hybrid propulsion systems was discussed. The idea arose to summarize the contributions and discussed topics in a document that can be helpful and of interest to others who are dedicated to system simulation.

The need to mitigate climate change poses major challenges for the shipping industry. The IMO GHG strategy, local emission legislations and other future-proofing initiatives are motivating the development of innovative propulsion systems for maritime applications. In addition, the digital transformation and increasing system complexity are leading to a change in the shipbuilding process and in system integration. Virtual methods should provide the best possible support for system integration and help to improve the development and operation of environmentally compatible applications.

This guideline is intended to take advantage of virtual system integration and system simulation and avoid mistakes. It informs about technical aspects of virtual system integration and should enable the reader to better understand and / or evaluate the topic from a technical point of view. The focus is on a consideration of terminology, modeling fidelity, interfaces, use cases, and anticipated challenges. This guideline does not claim to be complete. Model-based methods are subject to rapid further development and the content reflects the level of experience and knowledge of the subgroup members in 2022. Notes, additions and comments are welcome and will be taken into account in a revision of the guideline if necessary.

# 2 Motivation & Aim

Hybrid systems are one of the most complex energy systems to use chemical, electrical and other energy sources in parallel by different energy conversion machinery producing mechanical and electrical output. Initially, hybrid systems were installed to reduce fuel consumption and $CO_2$ emissions. In recent years, additional requirements were added to further reduce the environmental impact and to achieve further goals. As a result, system complexity continues to increase and the effort and requirements for system integration become greater.

This is where virtual system integration (abbreviated to VSI in the following) comes into play: It is a suitable method for virtually investigating and optimizing engineering complex systems involving multiple independent elements with interactions that result in emerging properties that may not be fully understood or predicted from the full knowledge of its elements alone. Furthermore, co-simulation and the consolidation of the tool landscape are additional drivers for VSI. Therefore, the use of dedicated simulation as a first step before realization is essential to ensure optimal system design and operation as well as safety and regulatory compliance.

The CIMAC WG20 aims to define the core tasks of suitable system simulation tools, which serve as basis for standardization requirements regarding usability and effectiveness including interfaces according to common standards.

The initial working tasks of subgroup “Tools” are:

- Agreement on hybrid system structure and components (in cooperation with all WG20 members)
- Exemplary use cases for virtual system integration
- Overview of available simulation tools and its specifics
- Specification of tool requirements under aspects of hybrid system integration
- Definition of development needs
- Suggestion of minimum communication standards for input/output interfaces of systems, subsystems and components

It should be mentioned that virtual system integration does not only consist of pure system simulation. All topics of model-based development (e.g., model-based control system development, model-based calibration), the coupling of virtual with real components, virtual testing and the creation and use of digital twins in development and operation are also associated with virtual system integration.

This guideline paper will deliver a neutral perspective on all these tasks and will not recommend the application of specific products.

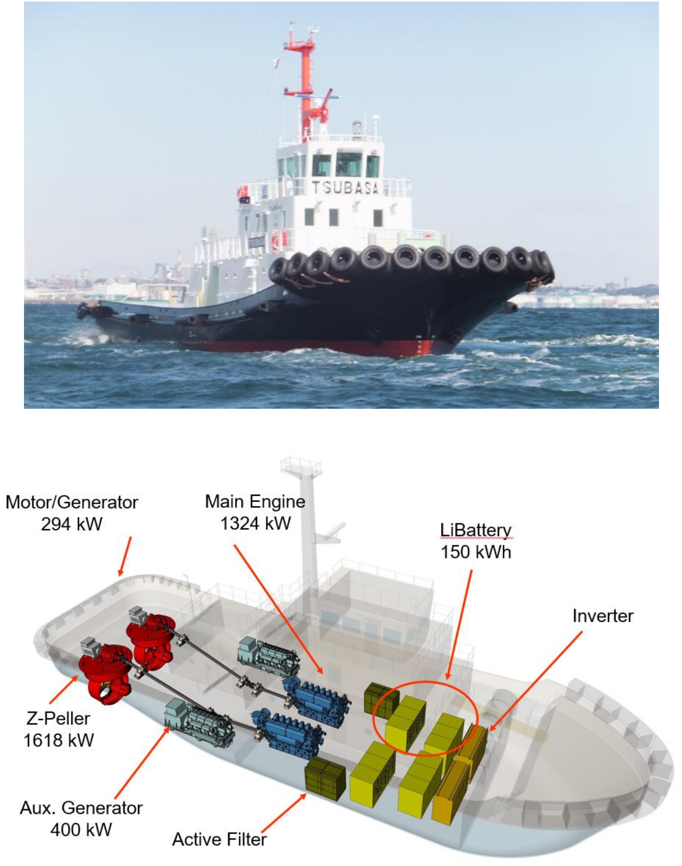


*Figure 1: Example of hybrid tug boat: “Tsubasa” developed by Niigata Power Systems*

## 2.1 Importance of Virtual System Integration

Ships have always been complex due to the need of a vast operational independence from the outside world. Since a long time, simulation is used in ship building to achieve an optimum layout and safety. In the past it was always strived for keeping the ship's machinery part as simple and effective as possible enabling a safe, reliable and efficient operation under nearly all circumstances. Today these systems are getting more and more complex due to the need to reduce the environmental footprints drastically. The use of hybrid machinery systems is one important solution for this. This adds another level of complexity to the already complex process of designing and building ships, especially due to the increased number of systems, subsystems and components, as well as a large number of parties/suppliers involved. The virtual system integration is the most important methodology to keep the layout, specification and optimization manageable. This helps ship owners and operators, shipyards, authorities and classification societies to efficiently build and operate new ships with significantly reduced environmental impacts.

The following are the key items that should be examined when considering a hybrid system:

1. Examination of CAPEX
2. Examination of OPEX
3. $CO_2$ reduction potential or – even better – the overall GHG reduction potential (including usage of alternative fuels)
4. Evaluation of operational benefits of hybrid systems, such as spinning reserve, peak shaving, electrical crank up, zero-emission operation in sensitive areas, etc.
5. Safety, reliability and compliance with regulations
6. Commissioning (alternative approach to physical testing only)

Of these, 1. and 2. are items which were examined already in the past, but today 3. and 4. are issues that need to be additionally examined in detail due to the recent requirements to reduce the environmental impact and take full advantage of hybrid systems. Requirements for the use of non-diesel fuels such as gas, hydrogen, methanol and ammonia are expected to increase rapidly for the engines that have been mainly used to date. It is likely that the use of alternative fuels will increase the complexity of the engine architecture and operation, so a hybrid system can help to ease these issues by enabling a simpler operating regime. In this context, preliminary investigation through virtual system integration and system simulation is becoming increasingly important, which also supports commissioning and ensures safety and regulatory compliance.

## 2.2 Use Cases & Fields of Application

### 2.2.1 Introduction

As outlined in the previous chapter, there are many reasons for virtual system integration (VSI). Hereinafter, some selected use cases for the application of VSI are presented. Aiming to cover the whole development cycle of a hybrid propulsion system (i.e., from layout/design to integration and even up to system operation) the following use cases will explicitly demonstrate examples regarding the initial concept design, energy management optimization, and system verification and validation.

Obviously, the full scope of deployment of VSI may include, among others, all kind of ship types (e.g., cruise ships, ferries, ice-class ships, merchant vessels, tugboats, yachts) and operation scenarios (like dynamic positioning, loading/unloading, maneuvering in ice, PTI/PTO operation, sea trials according to ISO 19019:2005(en), etc.). In addition to that, each application scenario may be addressed by different simulation platforms and tools (e.g., AVL CRUISE™ M, GT-SUITE, Open Simulation Platform, Siemens Simcenter), methods or modeling fidelities (see chapter 4).

For these reasons, the selected use cases are intended to provide a tool-neutral overview of a possible use of VSI, aimed primarily at better illustration and understanding of the subject.

### 2.2.2 Example A: Optimization of Topology

This example describes a comparison of ROPAX vessel propulsion systems. The main dimensions of the ship are given in Table 1: Vessel main characteristics.

| | |
|---|---|
| Length between perpendiculars | 75 m |
| Ship DWT | 720 metric tons |
| Breadth | 12 m |
| Draught | 3.2 m |
| Capacity | 400 persons |
| Number of propellers | 2 |
| Hotel load’s average power | 500 kW |

*Table 1: Vessel main characteristics*

The vessel is aimed to make two trips per day, both at a cruising speed of 14 knots. The daily mission profile is shown in Figure 2.

.

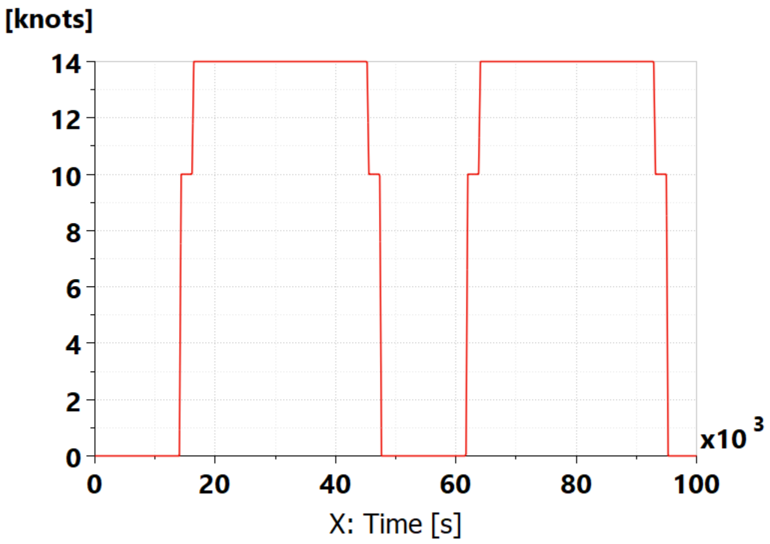
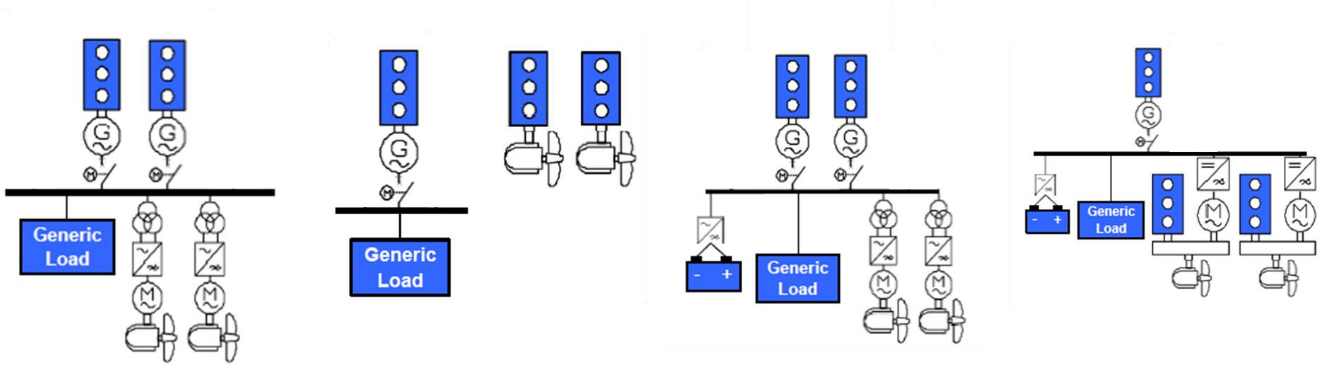


*Figure 2: Vessel targeted speed as a function of time*

To power this ROPAX vessel in such conditions, the following propulsion architectures are compared (Figure 3):



Diesel on demand Diesel mechanical Hybrid Mechanical assist

*Figure 3: Propulsion architectures*

The engines are modeled using tabulated values for brake mean effective pressure (BMEP) / torque and fuel flow / brake specific fuel consumption (BSFC).

In this work, these maps are generated for three 4-stroke diesel MTU engines: 8-cyl. engines (32 L) producing 880 kW, 12-cyl. engines (48.7 L) with 1320 kW and 12-cyl. engines (65 L) with 1760 kW. A model of a smaller Volvo Penta engine with 13 L is also generated (590 kW).

Electric generators and motors are functionally modeled with a constant efficiency of 95%. A further 5% loss is incorporated into the electrical system considering conversion losses for the architectures using batteries. The hybrid vessel uses a battery with a capacity of 0.5 MWh (1600 Ah) and the assist variant uses a battery with a capacity of 0.25 MWh (800 Ah). The battery is also modeled in a functional way with a constant internal resistance.

The main simulated performance data are summarized in Figure 4 for the hybrid architecture.

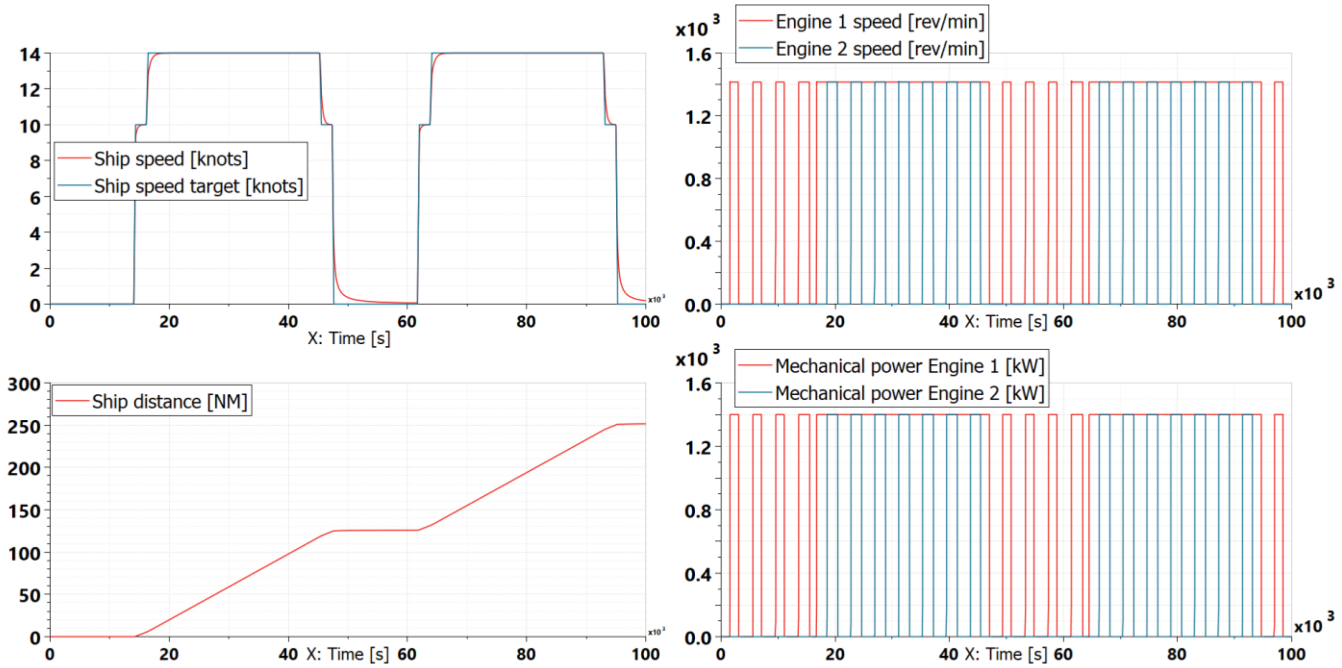


*Figure 4: Performance results for the Hybrid architecture*

It can be seen that the vessel reaches the cruising speed of 14 knots. Below that speed only engine 1 is in operation and the battery is used. The engine starts and stops depending on the SOC of the battery.

The cumulated $CO_2$ emissions over a day of operation are compared for the 4 different architectures in Figure 5.

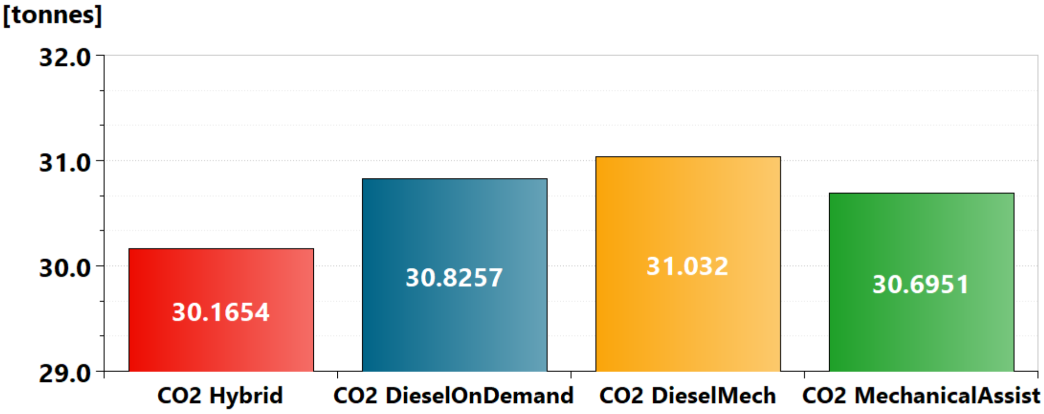


*Figure 5: $CO_2$ cumulated emissions of the 4 architectures*

Finally, an automated optimization was performed to investigate further solutions. The objective of the optimization is to reduce both the OPEX (through fuel consumption) and the CAPEX (through the cost of the propulsion system).

To estimate the OPEX, fuel costs (MGO) are assumed to be 667 USD per metric ton. To estimate the CAPEX, rough assumptions are made for engine cost (1000 USD per liter displacement), battery pack cost (200 USD per kWh) and assist PTO/PTI/Clutch system (8000 USD). These figures are admittedly not documented and can be discussed.

In this design optimization phase, the following parameters and configurations were varied:

- the architecture (Hybrid or Mechanical assist)
- the engine (Volvo Penta D8, Volvo Penta D13, MTU 880 kW, MTU 1320 kW, MTU 1760 kW)
- the battery capacity (from 400 to 1600 Ah)
- the assist activation speed (from 4 to 13 knots)

Two limiting constraints apply to the optimization study:

- The ship speed must be higher than 13.5 knots.
- The battery state of charge must not fall below 5%.

The results are represented by a Pareto front, with the X-axis representing OPEX and the Y-axis representing CAPEX. The goal is basically to be at the bottom left of the graph. Figure 6 shows the feasible configurations for the engine types, battery capacity and architectures.

*Figure 6: OPEX/CAPEX trade-off optimization results*

The best configuration as a result of the optimization study shows a CAPEX improvement of 51% and an OPEX improvement of 37% compared to the baseline design that was used in the Mechanical assist from the initial study.

### 2.2.3 Example B: Energy Management Strategy Optimization

In the early proof-of-concept phase, system integrators face the problem of optimizing a ship's energy management strategy without detailed knowledge of the actual system layout or topology. In such scenarios, virtual system integration provides the possibility to evaluate the benefits or trade-offs of a certain energy management (EM) strategy depending on the given operational power demand profiles.

In the present case, a hybrid topology for an ocean-going LNG carrier vessel was selected and kept constant for all EM strategies investigated (see Figure 7). The system consists of a 2-stroke dual-fuel main engine (>7 MW), four diesel gensets (each 1.5 MW), a 2 MWh battery and a 1500 kW permanent magnet shaft generator. Recorded operational profiles for propulsion and auxiliary load demand were selected from the system integrator's database. The voyage took 42 days.

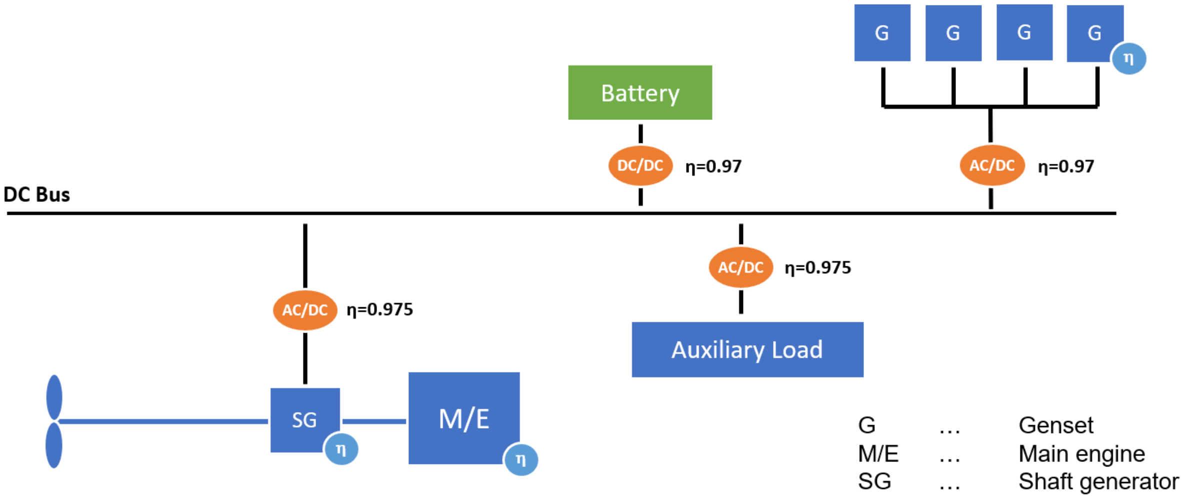


*Figure 7: System topology*

Three significantly different EM strategies – namely, *PTO-High*, *PTO-Low*, and *PTI* (see Table 2) – were used to illustrate the impact of the chosen strategy on overall fuel consumption.

| **Power Management** | **EM Strategy "PTO-High"** | **EM Strategy B "PTO-Low"** | **EM Strategy "PTI"** |
|---|---|---|---|
| Propulsion power request | Completely fulfilled by M/E | Completely fulfilled by M/E | Assuming M/E can only provide limited power, i.e. remaining gap is fulfilled by PTI using energy provided by gensets |
| Auxiliary power request | M/E provides high amount of extra power to support auxiliary load request. Remaining auxiliary load gap is fulfilled by battery or gensets (in order of priority) | M/E provides a lower amount of extra power to support auxiliary load request. Thus, compared to "PTO-High" the remaining auxiliary load gap is bigger | When available extra power from M/E is used to support auxiliary load. However, auxiliary load is mainly covered by gensets. |

*Table 2: EM strategy overview*

A quasi-static modeling approach (see chapter 4.2.3) has been applied. Detailed data (efficiency maps) of the main consumers were available allowing a reliable estimation of the fuel saving potential when shifting load demand from the gensets to the main engine and vice versa. Figure 8 shows an example of how the main engine's power output is distributed between the propulsion power demand and the excess power for the shaft generator. In this context, the following points should be noted: The dual-fuel main engine was mainly operated with natural gas, while the gensets were operated with diesel fuel. Therefore, in addition to the efficiency benefit of the main engine, a further reduction of $CO_2$ emission was achieved solely through the fuel composition itself. Obviously, in this example the engine operates at rather low load corresponding to low engine speed. However, for most of the time the relative engine speed stays above 40%. In this area, PTO operation is generally possible whereas there is an effect on SG efficiency, which for the sake of simplicity has been neglected in this case.

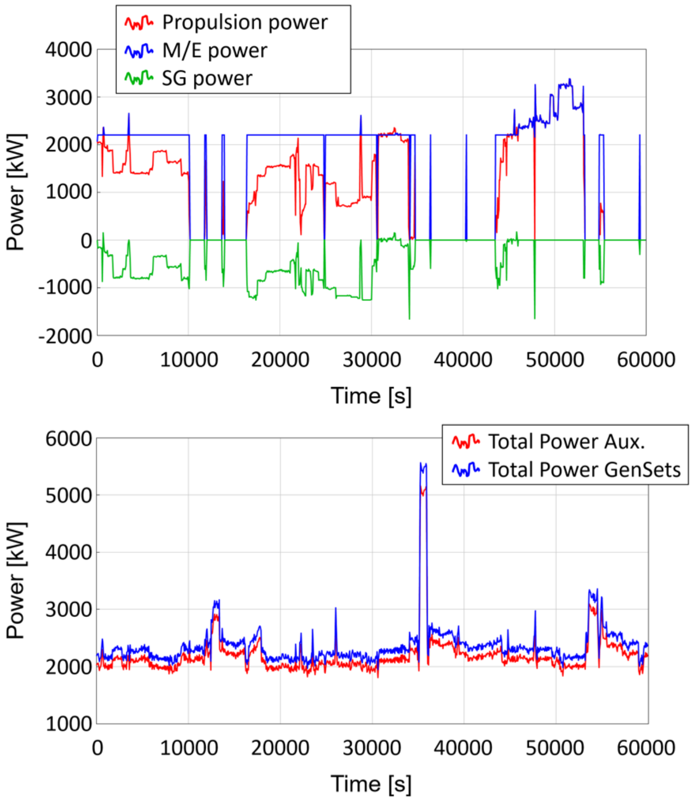


*Figure 8: "PTO-Low" Strategy*

A comparison of the three above-mentioned EM strategies reveals that there are significant differences in terms of overall fuel consumption and $CO_2$ emission – *PTO-High* and *PTI* differ by almost 14% (see Table 3). Of course, these examples represent rather extreme scenarios. They don't provide sufficient information for judging each technology/strategy in general as the different outcomes depend on various factors (such as component sizing, operating profile, auxiliary load demand, etc.). For example, the PTI strategy may not turn out very well in the present case however there might be benefits that are just hidden in such an isolated comparison. For instance, an appropriately sized battery may lead to CAPEX savings by reducing the number of gensets or by a different choice of the main engine. Further, it could also allow for fully electrical maneuvering in port.

Concluding, by means of this comparison study it has been illustrated that it's necessary to consider and optimize the energy management strategy if the potential of a hybrid system is to be truly assessed.

| EM Strategy | "PTO-High" | "PTO-Low" | "PTI" |
|---|---|---|---|
| ΔGas consumption | +130 ton (+75%) | +63 ton (+36%) | -27 ton (-16%) |
| ΔDiesel consumption | -174 ton (-35%) | -70 ton (-14%) | +52 ton (+10%) |
| ΔCO2* | -190 ton (-9.3%) | -47 ton (-2.3%) | +89 ton (+4.4%) |

* *LNG* → *2.75* $kg_{CO2} / kg_{fuel}$ *Diesel* → *3.15* $kg_{CO2} / kg_{fuel}$

*Table 3: Overall $CO_2$ savings*

### 2.2.4 Example C: Virtual Testing, Verification and Commissioning of a Tourist Boat

The requirements of tourist boats fit well with hybrid propulsion systems. These include silence, vibration-free operation, autonomy and comfort. Hence, it is crucial for a propulsion system supplier to be able to demonstrate the value of a hybrid propulsion system to the shipyard. Simulation is a good way to do this.

The propulsion system supplier must also ensure that the commissioning phase is short, smooth and does not affect the delivery schedule. Virtual commissioning is an advantageous way to shorten commissioning times.

This use case shows how a propulsion system supplier uses simulation to promote the value of its products to the shipyard and reuse the models for virtual commissioning.

The propulsion system is defined by the single-line diagram in Figure 10:

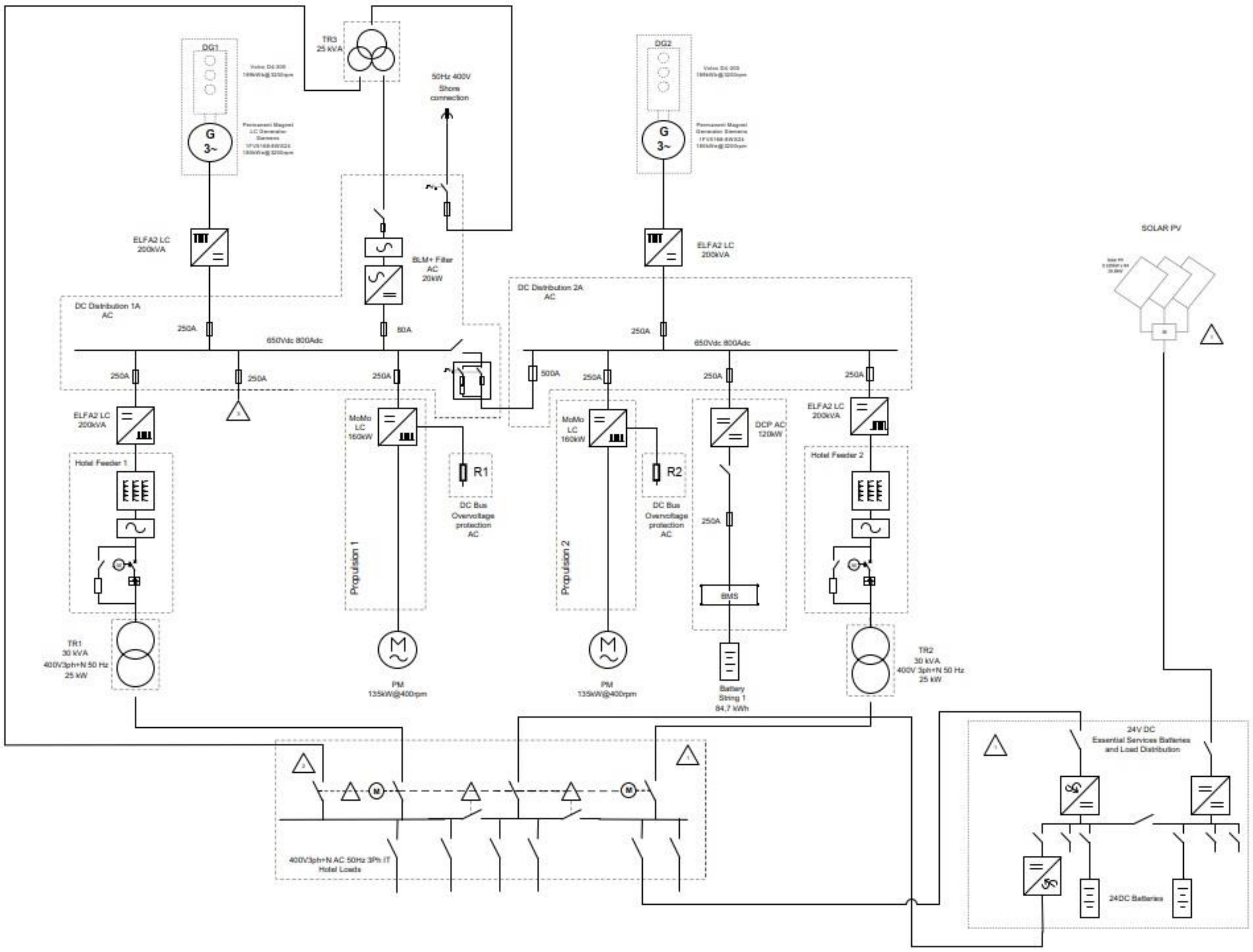


Figure 10: Single-line diagram

The power management system control logic of such a system is shown in Figure 11:

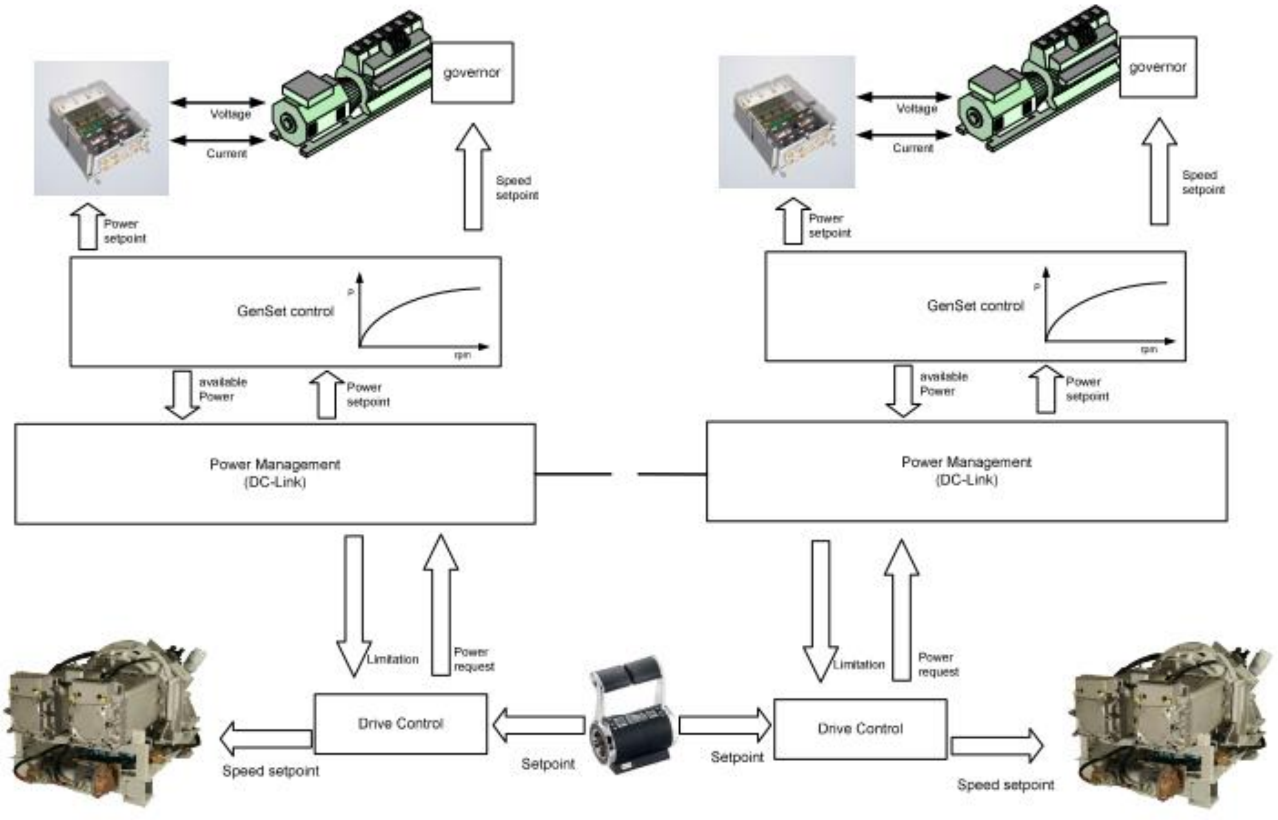


*Figure 11: Power management system layout*

An overview of the MIL and HIL methods is shown in Figure 12:

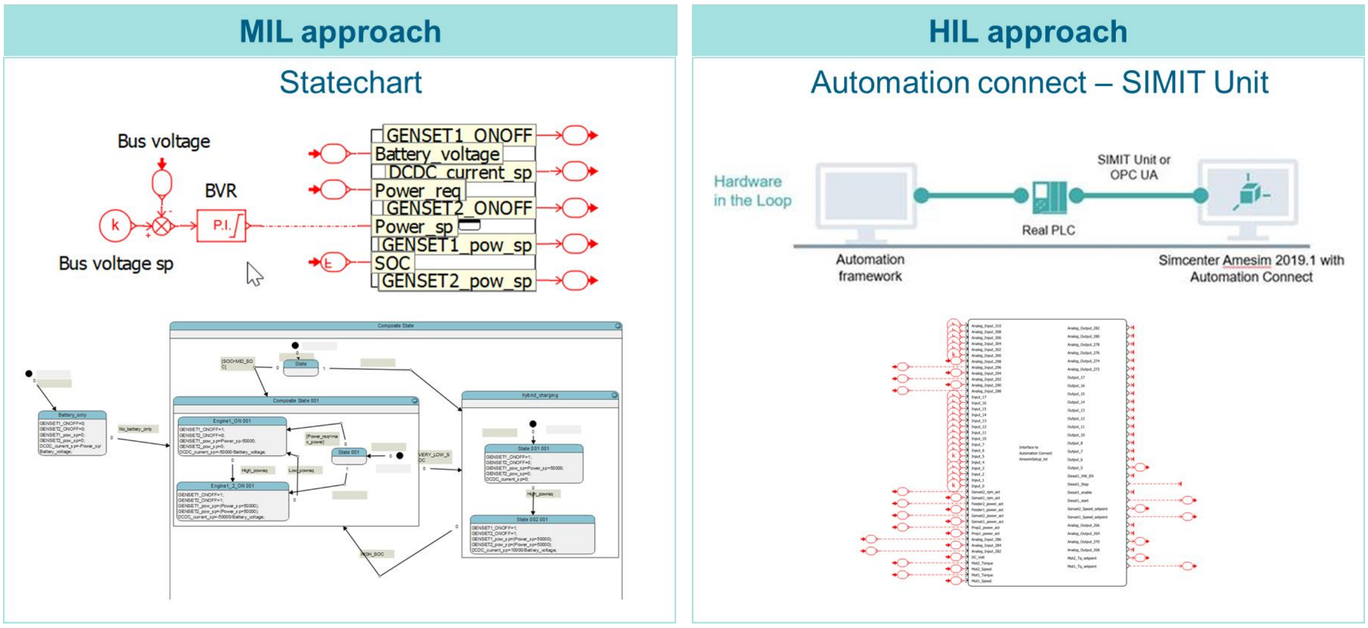


*Figure 12: MIL and HIL controllers embedded into the simulation environment*

In the “request-for-quotation” phase (showing the value of the system to the shipyard), the Model-In-the-Loop (MIL) verification method is used. This means that the simulation model of the propulsion system is connected to a simulation of the power management software functions. The goal is to obtain a functioning control system controlling the virtual system.

Once the potential of the propulsion system is verified and the programmable logic controller (PLC) has been generated, the coupling of the simulation model with the real power management system PLC using can be done. This so-called Hardware-In-the-Loop (HIL) verification method enables virtual commissioning and allows possible problems to be solved even before real commissioning.

Figure 13 shows some results of the battery potential assessed using Model-In-the-Loop:

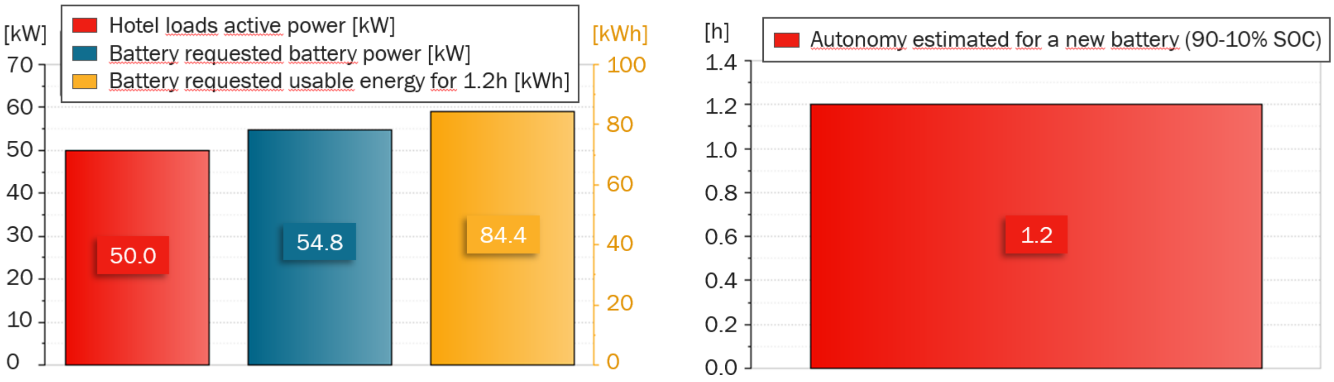


*Figure 13: Battery power, energy and autonomy analysis*

Overall, power management systems play an important role in the performance of hybrid propulsion systems and must be carefully developed. Coupling methods such as MIL and HIL enable simple, continuous and reusable verification of software functions. This can both reduce the time-to-sea and improve the propulsion system’s performance.

# 3 Hybrid Terminology

## 3.1 Definition of Hybrid System

There are several attributes that can help describe a hybrid system (e.g., the presence of an energy storage system or the use of alternative energy sources), but according to the findings of CIMAC Working Group 20, there should be only one general attribute that classifies a hybrid system, and that is:

> **“The ability to perform a specific function (power production or propulsion) based upon at least two different and independent technologies”**

In this concise definition, the terms "different" and "independent" play a particularly important role.

‘Different’: *Two or more systems are different from each other if their underlying core energy conversion processes are different.*

‘Independent’: *An independent system consists of two or more technologies that are capable of operating independently from each other and while serving the intended specific function (e.g., propulsion or power generation). The specific function has to be fulfilled even in case the other hybrid technology is not available.*

Detailed and further considerations are contained in the CIMAC Guideline ‘Maritime Hybrid Systems’, which deals with the definition of hybrid systems and also gives examples.

## 3.2 Operating Modes

A hybrid system can be operated in different operating modes based on the load request and the state of the energy storage. The operating modes are optimized to achieve low fuel consumption, fulfil transient and steady-state load request and ensure low emissions. Following hybrid operation modes are listed according to energy consumption:

- Propulsion of the ship
    - Propulsion power is provided by M/E only
    - Propulsion power is provided by e-motor (battery electric, fuel cell electric) only
    - Hybrid system supports M/E during steady-state operation (load shift down)
        - To reach better engine efficiency
        - To discharge the battery
        - To ensure higher overall power
        - To have lower emission or noise
    - Hybrid system supports the ICE during dynamic operation (load shift down)
        - To have faster engine load built up
        - To cover short load peaks higher than steady-state engine load
        - To have lower emission during load built up
- Electric power supply for hotel load
    - Electric power is provided by ICE only
    - Electric power is provided by battery of fuel cell only
    - Electricity is provided by ICE supported by battery

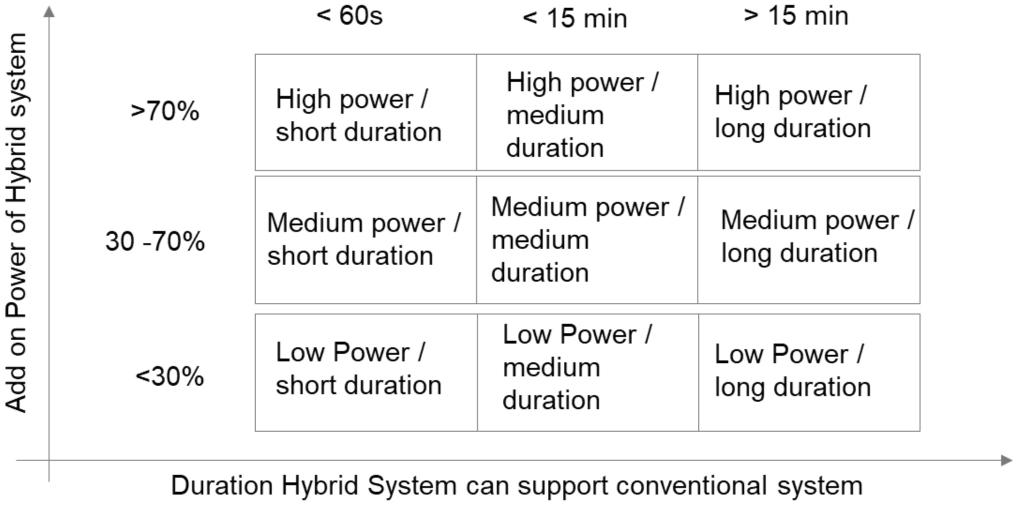
- Charging the hybrid storage by increasing the engine output higher than required for propulsion and hotel load
    - Balancing of SOC
    - Shifting of engine operating point to higher load for better efficiency
    - Shifting of engine operating point to higher load to ensure high efficiency of exhaust gas aftertreatment system

## 3.3 Naming Convention

Within this chapter a naming convention for hybrid ship applications is investigated. The objective of a common naming convention is that with a short and clear naming convention the base properties of a hybrid ship layout could be defined. The naming is based on the Px naming convention used in the automotive industry. For automotive P0 to P4 are used to describe parallel hybrid systems. The number is used to define the position of the E-motor in the powertrain. For the powertrain it is assumed that, in addition to the engine, there is a clutch, a gear box and one driven axle. The 1 stands for e-motor between engine and clutch, the 2 between clutch and gear box, 4 means an e-motor on an axle not mechanically connected to engine. This naming convention could not be readily adopted because marine applications have a less fixed propulsion line configuration consisting of engine, clutch, gearbox and propeller and further configurations with more than one engine are common.
The naming convention for marine hybrid applications can be done based on different categories:

- Power the hybrid system can deliver additionally to the conventional power supply
- Duration the hybrid system can provide full hybrid system power
- Propulsion layout
- Plug-in capability
- Energy storage type
- Electric grid type and layout

The following naming convention covers the 4 most important items on the list. To define the hybrid add-on power, the 3 levels low, medium and high are defined. It is defined as the ratio of the add-on power of the hybrid system to the power of the conventional power system. Figure 14 shows the definition of the power level.



| | < 60s | < 15 min | > 15 min |
|---|---|---|---|
| >70% | High power / short duration | High power / medium duration | High power / long duration |
| 30 -70% | Medium power / short duration | Medium power / medium duration | Medium power / long duration |
| <30% | Low Power / short duration | Low Power / medium duration | Low Power / long duration |

*Figure 14: Definition of hybrid power and possible support duration*

In addition to power, the duration of how long a hybrid system can operate at full hybrid power and thus the capacity of the energy storage system is important. Like power, the duration is also characterized with 3 levels, see Figure 14. Short duration means below 60s and so the hybrid assistance is mainly used to compensate the delay during the built-up of the engine load. In the medium level, the electric part of the hybrid system can provide support for up to 15 min compensating for sudden load increase till a second engine has started and reached full power. In the long duration level, the electric drive can be operated for a longer time and switching off the engines even at high power demand is possible, as it is the case in zero emission zones.

The base layout of a hybrid system is defined with 7 different layouts named from L0 to L6, see Figure 15. The different layouts differ mainly by the e-motor, engine and gear box position.

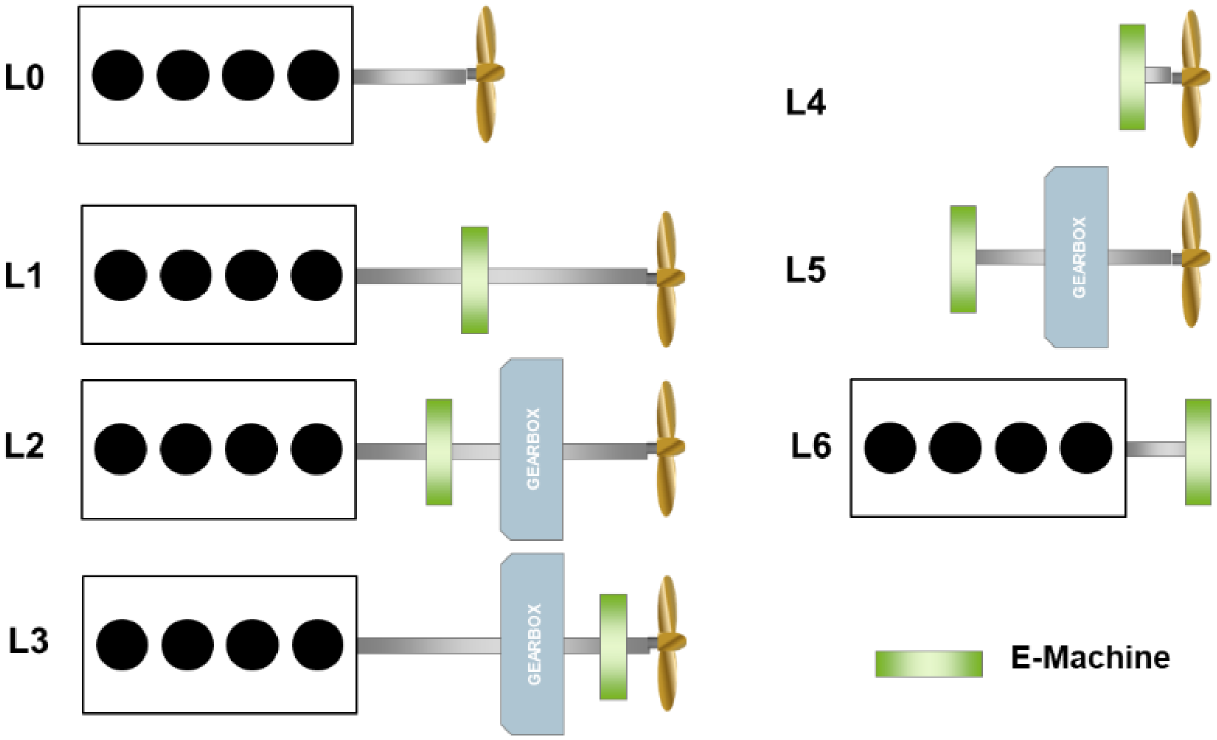


Figure 15: Definition of Hybrid base layout from L0 to L7

In order to have also the possibility of a clutch in the powertrain, a second number is introduced to indicate whether a clutch is located before or after the e-motor, see Figure 16.

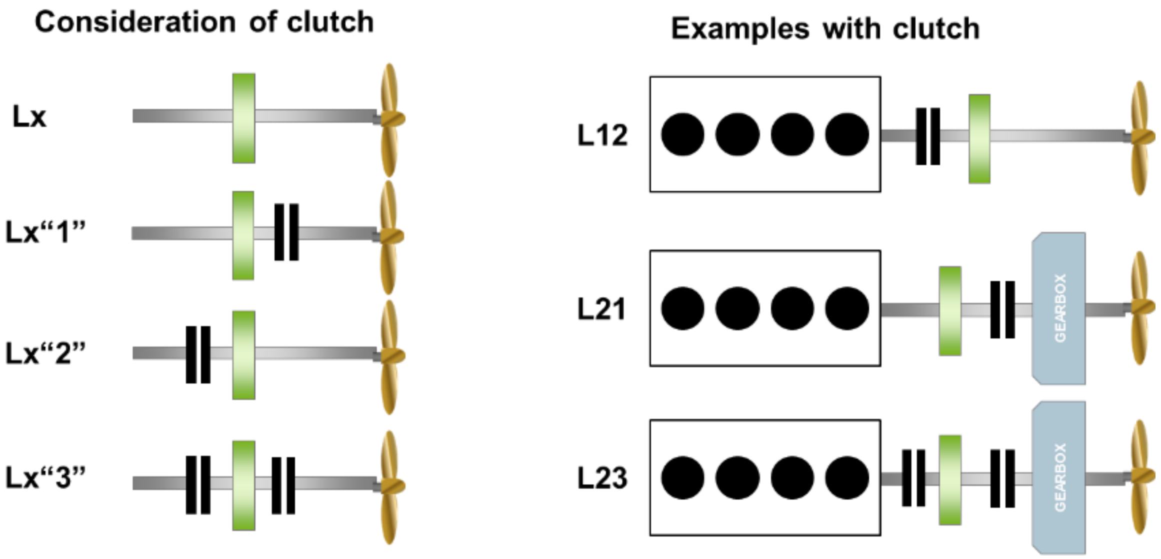
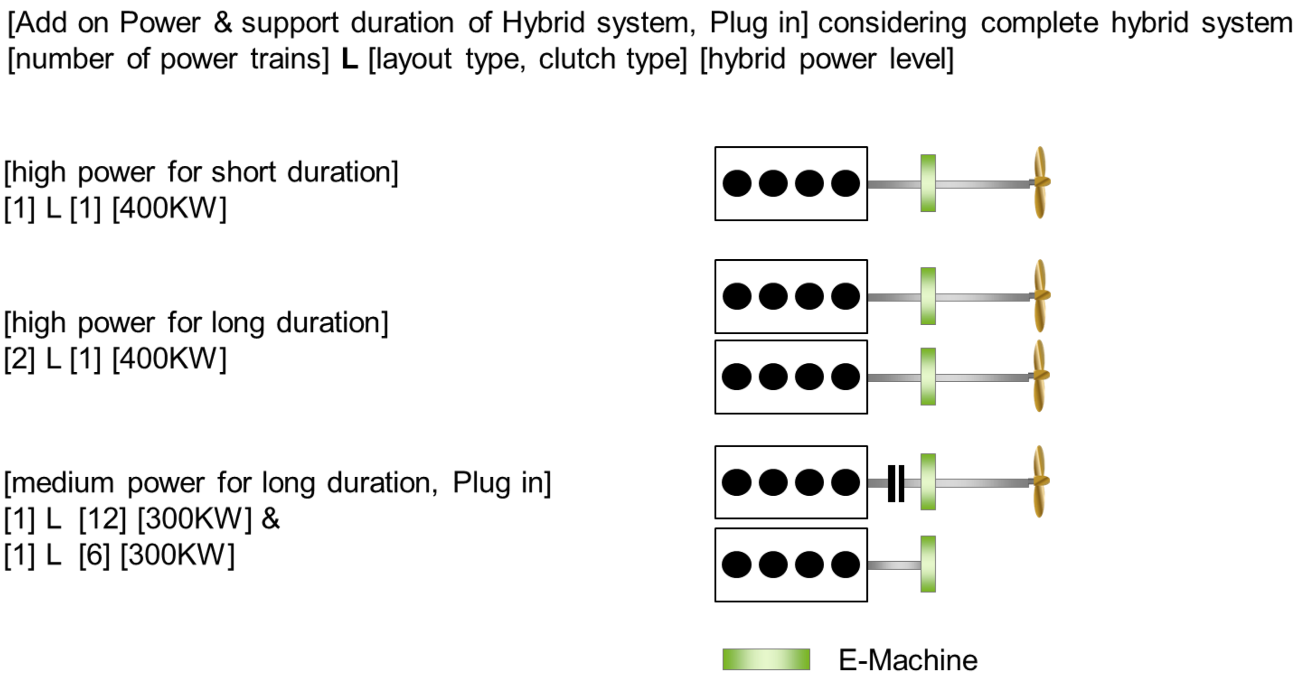


*Figure 16: Consideration of clutch in hybrid system layout*

To name the hybrid configuration in a first line the power and duration level is given as well as it as the possibility of plug-in charging. In following lines serial and parallel engines are named. Behind the L also the maximum shaft power of the powertrain is given.
Examples are illustrated in Figure 17.



[Add on Power & support duration of Hybrid system, Plug in] considering complete hybrid system
[number of power trains] **L** [layout type, clutch type] [hybrid power level]

[high power for short duration]
[1] L [1] [400KW]

[high power for long duration]
[2] L [1] [400KW]

[medium power for long duration, Plug in]
[1] L [12] [300KW] &
[1] L [6] [300KW]

*Figure 17: Definition of hybrid ship naming convention with 3 examples*

In addition to specifying these designations to characterize the hybrid system, the use of single-line diagrams is also recommended for complex systems.

# 4 Modeling Fidelity

## 4.1 State of the Art

“The predictive power of a model depends on its ability to correctly identify the dominant controlling factors and their influences, not upon its completeness.” This is an adaptation of Occam’s Razor to modeling by Oberkampf and Roy.

“Model building is the art of selecting those aspects of a process that are relevant to the question being asked.” – Holland, JH (1995) Hidden Order. Addison-Wesley, New York, USA.

These quotes highlight the importance of clearly identifying and understanding the scope of a given model. The first part of this chapter aims at defining what are the different metrics to characterize modeling fidelity.

A model is defined by several attributes that are listed here, based on definitions found in (Ponnusamy, 2019):

- **Abstraction** – The process of selecting the essential aspects of a real-world system to be represented in a model or simulation while ignoring those aspects that are not relevant to the purpose of the model or simulation.
- **Accuracy** – The degree to which a parameter or variable or set of parameters or variables within a model or simulation conform exactly to reality or to some chosen standard or referent.
- **Capacity** – The number of instances of an object or detail that are simultaneously represented by a model or simulation.
- **Error** – The difference between an observed, measured, or calculated value and a correct value.
- **Fidelity** – The methods, metrics, and descriptions of models or simulations used to compare those models or simulations to their real-world referents or to other simulations in such terms as accuracy, scope, resolution, level of detail, level of abstraction and repeatability. Fidelity can characterize the representations of a model, a simulation, the data used by a simulation (e.g., input, characteristic or parametric), or an exercise. Each of these fidelity types has different implications for the applications that employ these representations. (SISO)
- **Fitness** – Providing the capabilities needed or being suitable for some purpose, function, situation or application.
- **Precision** – 1. The quality or state of being clearly depicted, definite, measured or calculated. 2. A quality associated with the spread of data obtained in repetitions of an experiment as measured by variance; the lower the variance, the higher the precision. 3. A measure of how meticulously or rigorously computational processes are described or performed by a model or simulation.
- **Sensitivity** – The ability of a component, model or simulation to respond to a low-level stimulus.
- **Tolerance** – The maximum permissible error or the difference between the maximum and minimum allowable values in the properties of any component, device, model, simulation or system relative to a standard or referent. Tolerance may be expressed as a percent of nominal value, plus and minus so many units of a measurement, or parts per million.
- **Validity** – 1. The quality of being inferred, deduced, or calculated correctly enough to suit a specific application. 2. The quality of maintained data that is found on an adequate system

of classification (e.g., data model) and is rigorous enough to compel acceptance for a specific use. 3. The logical truth of a derivation or statement, based on a given set of propositions.

- **Resolution** – 1. The degree of detail used to represent aspects of the real world or a specified standard or referent by a model or simulation. 2. Separation or reduction of something into its constituent parts; granularity.

The modeling fidelity can be characterized as per the following scheme (Figure 18) according to (Roza, 2005):

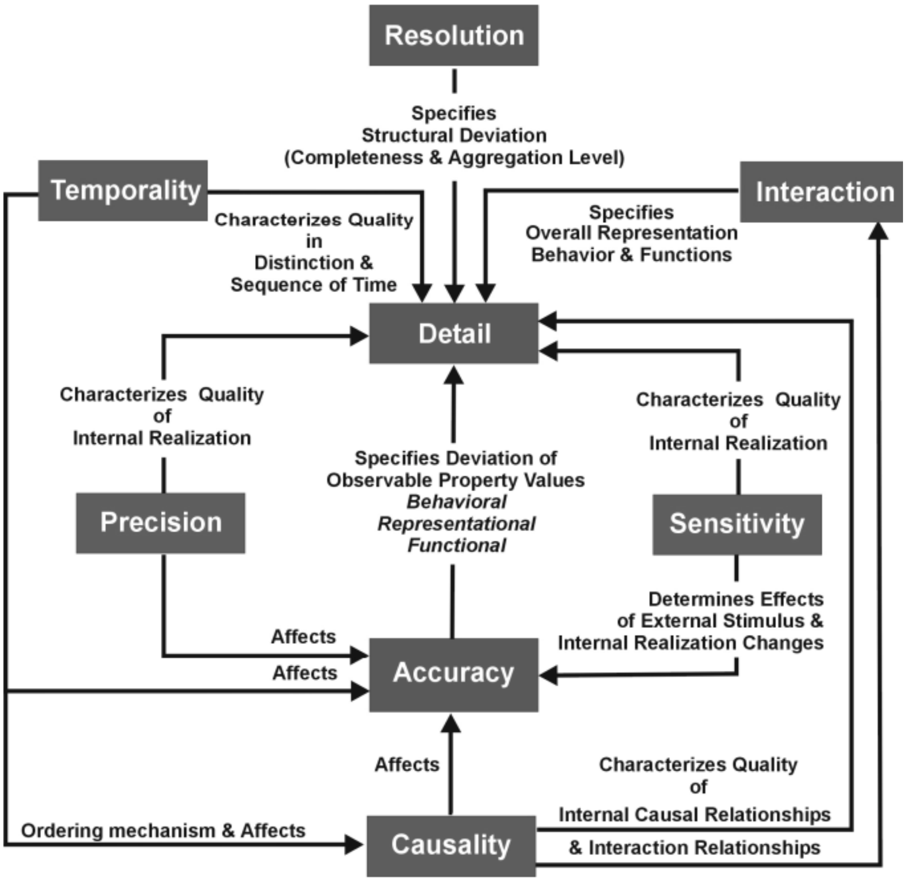


*Figure 18: Fidelity characterization concepts scheme*

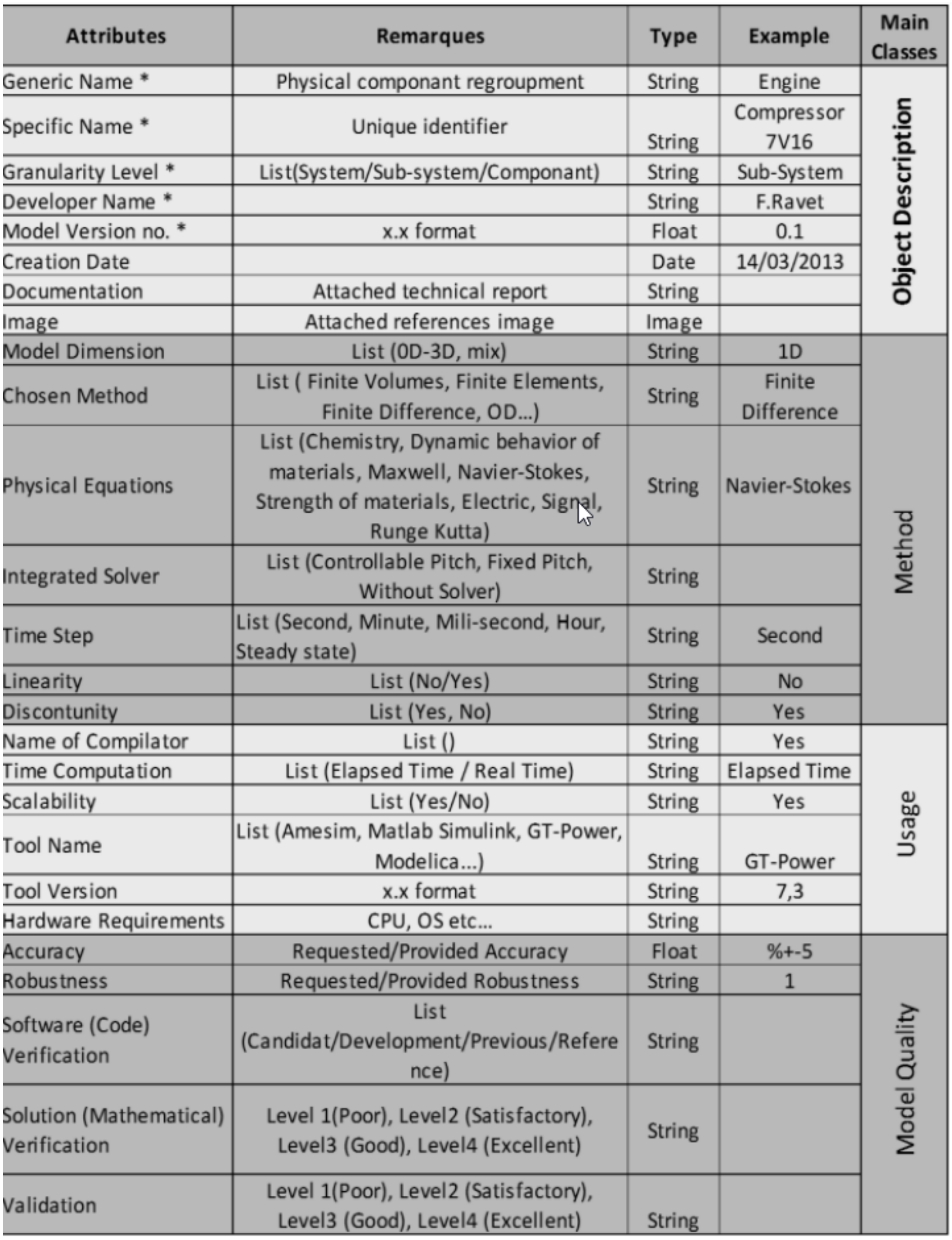
It can be seen that modeling fidelity may vary in numerous dimensions which might also explain why there is no clear consensus on how to characterize modeling fidelity levels in general. It is not uniformized in any industry. For illustrative purposes, below are three examples of modeling fidelity level definition:

According to the **NASA** (Roe, 2019), modeling fidelity is defined by these 3 criteria:

1. Abstractions (including simplifications) in the model, e.g., physical laws or processes ignored or adjusted.
2. Basis of empirical or phenomenological model of Real World Scenario, as opposed to that of physical law or explanatory model; observed behaviors mimicked vs. detailed processes described.
3. Complexity of the model, e.g.:
    - Complexity of excitation equations.

- Mathematical submodels used to complement sets of equations in the main model, e.g., analytical equations, ordinary differential equations (ODEs) and partial differential equations (PDEs) for constitutive properties of materials and fluids, PDEs for fluid turbulence modeling.
- Estimated level of temporal and/or spatial discretization needed to achieve defined modeling and simulation objectives and requirements. Model testing is required to confirm the adequacy of these initial estimates.
- Deterministic or non-deterministic specifications

**Renault SA** uses a so-called “Model Identity Card” (Göknur, Paredis, Yannou, Coatanéa, & Landel, 2015) which includes much more metadata listed below (Figure 19):



| Attributes | Remarques | Type | Example | Main Classes |
|---|---|---|---|---|
| Generic Name * | Physical componant regroupment | String | Engine | Object Description |
| Specific Name * | Unique identifier | String | Compressor 7V16 | |
| Granularity Level * | List(System/Sub-system/Componant) | String | Sub-System | |
| Developer Name * | | String | F.Ravet | |
| Model Version no. * | x.x format | Float | 0.1 | |
| Creation Date | | Date | 14/03/2013 | |
| Documentation | Attached technical report | String | | |
| Image | Attached references image | Image | | |
| Model Dimension | List (0D-3D, mix) | String | 1D | Method |
| Chosen Method | List ( Finite Volumes, Finite Elements, Finite Difference, OD...) | String | Finite Difference | |
| Physical Equations | List (Chemistry, Dynamic behavior of materials, Maxwell, Navier-Stokes, Strength of materials, Electric, Signal, Runge Kutta) | String | Navier-Stokes | |
| Integrated Solver | List (Controllable Pitch, Fixed Pitch, Without Solver) | String | | |
| Time Step | List (Second, Minute, Mili-second, Hour, Steady state) | String | Second | |
| Linearity | List (No/Yes) | String | No | |
| Discontunity | List (Yes, No) | String | Yes | |
| Name of Compilator | List () | String | Yes | Usage |
| Time Computation | List (Elapsed Time / Real Time) | String | Elapsed Time | |
| Scalability | List (Yes/No) | String | Yes | |
| Tool Name | List (Amesim, Matlab Simulink, GT-Power, Modelica...) | String | GT-Power | |
| Tool Version | x.x format | String | 7,3 | |
| Hardware Requirements | CPU, OS etc... | String | | |
| Accuracy | Requested/Provided Accuracy | Float | %+-5 | Model Quality |
| Robustness | Requested/Provided Robustness | String | 1 | |
| Software (Code) Verification | List (Candidat/Development/Previous/Reference) | String | | |
| Solution (Mathematical) Verification | Level 1(Poor), Level2 (Satisfactory), Level3 (Good), Level4 (Excellent) | String | | |
| Validation | Level 1(Poor), Level2 (Satisfactory), Level3 (Good), Level4 (Excellent) | String | | |

*Figure 19: MIC classes and their attributes*

Finally, the USA Department of Defense Modeling and Simulation Enterprise (MSE) proposes a Verification Validation and Accreditation documentation where 3 types of modeling fidelity descriptions are proposed:

1. **Short descriptions** of simulation fidelity, including qualitative labels such as “high,” “medium,” or “low” fidelity. Such dimensionless characterizations tend to have more public relations utility than technical value in that they frequently lack the information content necessary to support technical decisions about simulation fitness.
2. **Shorthand descriptions** of simulation fidelity, including checklists, indicate that a simulation satisfies multiple, bundled attributes. For example, the Federal Aviation Administration's “Level D Flight Simulator” certification requires satisfaction of more than 100 specific attributes.
3. **Long descriptions** of simulation fidelity typically describe simulation fidelity in terms of multiple explicit attributes. The number and kinds of attributes considered varies with the construct being employed for simulation fidelity. Most constructs consider either the scope of the simulation's treatment of significant factors in the application domain (this usually involves some kind of enumeration), the quality of treatment of factors within the simulation (as indicated by parameter accuracy, resolution, etc.), or both.

Since there is no uniformized standard for modeling fidelity description, CIMAC will propose its own description based on shorthand descriptions. This description will be defined through examples presented in the next sections of this chapter.

## 4.2 Common Modeling Approaches

### 4.2.1 Overview

Table 3 gives an overview of the previously mentioned CIMAC model fidelity levels. In this overview the model characteristics are simply clustered into three model fidelity levels only – aiming to provide guidance specifically tailored to the topic of virtual system integration. Note that due to this pragmatic segmentation classification may not always be unambiguous.

### 4.2.2 Surrogate Model Approach

*Description*

Surrogate models are a widely used modeling approach that is also referred to as meta or black box models. Especially, the latter term indicates that the focus of surrogate model lies solely on the input-output relationship of a given system whereas detailed knowledge about the actual inner model mechanics is not given. Popular examples of surrogate model approaches are response surface, artificial neural network, Bayesian network or random forest models.

*Application*

Surrogate models are often utilized for optimization purposes (Yuan, Teng, Sun, & Huei, 2013) (Zhang & Xiong, 2015) (Peng, He, & Xiong, 2016). Further recent application examples are artificial neural networks which are deployed for a variety of cases, e.g., for engine emission/performance prediction (Uslu & Celik, 2018), general aspects of ship operation (Lazakis, Raptodimos, & Varelas, 2018) (Beşikçi, Arslan, Turan, & Ölçer, 2016), or in combination with more detailed model approaches for mitigation of computational hurdles (Zhang, Xu, Zhong, & Bai, 2020).

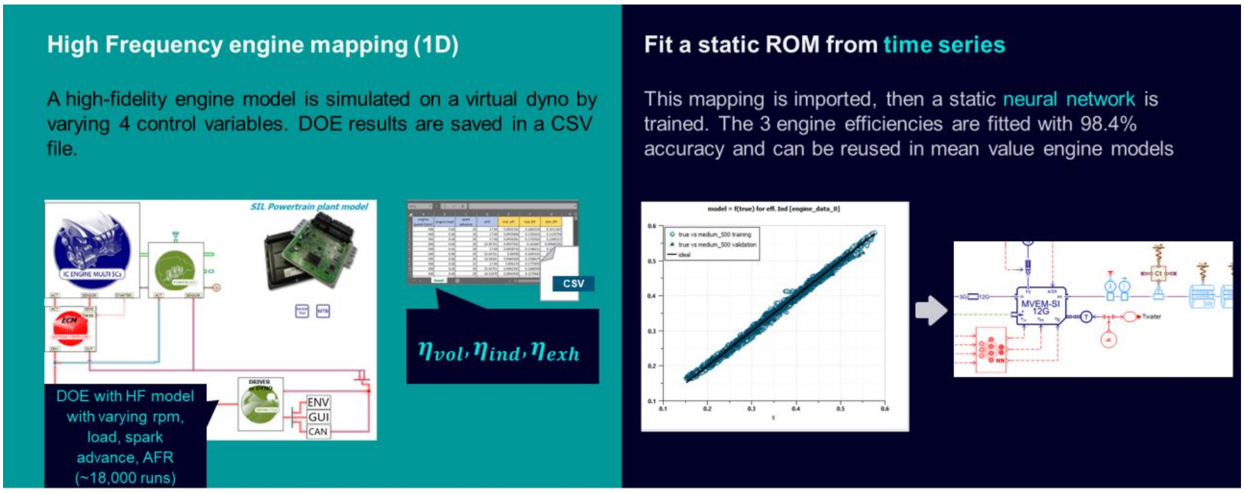
| **Model Fidelity Level** | **Description** | **Application** | **Simulation Direction** | **Level of Predictivity** |
|---|---|---|---|---|
| Surrogate Model | Data based models (meta or black box). Example: Response surface, Artificial Neural Networks | Early concept phase and also later phases: Optimal Control, plant models, or digital twins. I.e. applicable throughout whole development process | Backward, forward | Low, extrapolation of results is rather difficult |
| Quasi-static Model | Data based models mainly constituted of maps or correlations. Example: Driveline models | Proof of concept, CAPEX/OPEX analysis. Usually applied in the early design phase/topology optimization | Typically backward | Low (to mid), extrapolation of maps is difficult however additional correlations may enhance predictivity |
| Dynamic Model | Based on phenomenological models replicating physical correlations. Example: 1D-CFD flow calculations | Full system integration and verification, detailed scenario analysis, digital twins.<br>Suitable for later development stages, after topology selection | Typically forward | Mid to high, since models try to replicate physical phenomena extrapolation is possible |

*Table 3: Overview of modeling fidelity approaches*

In general, with respect to the process of virtual system integration surrogate models may be applied in cases when run time speed and simplicity are more important than accuracy or traceability of results. Since the benefits of hybridized propulsion systems are particularly evaluated during operation (i.e., during various maneuvers over several hours of time) run time speed plays an even more important role. In the end, deployment of surrogate models may actually be reasonable at different stages of the integration process, for instance:

- in the early proof-of-concept phase when no detailed project data is available and a potential analysis requested (e.g. fuel reduction potential for selected maneuvers/operational profiles)
- in later stages of controls development, optimization, and validation when computational speed is the main issue.
- Finally, surrogate models may also be suitable for the deployment in the final product like, for example, in a model predictive control or a digital twin model.

Typically, such surrogate models are derived from already existing high-fidelity models by utilizing model order reduction techniques (e.g., training of a neural network). In below diagram such conversion process is exemplarily depicted.



**High Frequency engine mapping (1D)**

A high-fidelity engine model is simulated on a virtual dyno by varying 4 control variables. DOE results are saved in a CSV file.

DOE with HF model with varying rpm, load, spark advance, AFR (~18,000 runs)

**Fit a static ROM from time series**

This mapping is imported, then a static neural network is trained. The 3 engine efficiencies are fitted with 98.4% accuracy and can be reused in mean value engine models

*Figure 20: Surrogate model generation process*

*Challenges*

The challenges of surrogate modeling are mainly connected to the quality and provision of the underlying data base. For instance, measurement data is not always covering the data range needed so physical simulation models may have to be utilized for creating more data. In this regard, design of experiment approaches may provide huge amounts of data which, however, are already introducing a first modeling error. Furthermore, the data generated will have to be replicated by means of a simplified mathematical correlation, e.g., a mathematical response surface or an artificial neural network (ANN). During the corresponding conversion process information may be lost and model accuracy may suffer. However, for state-of-the-art surrogate model approaches (like e.g., ANN) this accuracy loss may be smaller than 1%. Thus, the accuracy of the surrogate model is rather affected by the quality of the underlying data base.

*Level of predictivity*

As introduced in the *Challenges* part, the level of predictivity of the surrogate modeling approach is strongly linked to the quality and variety of the data base. If the prediction must be done within the data base boundaries, i.e. interpolation, then the level of predictivity is high. However, if the model must give results with inputs that are out of the training data boundaries, i.e. extrapolation, then level of predictivity is typically rather low.

### 4.2.3 Quasi-static Model Approach

*Description*

In comparison to surrogate models the quasi-static modeling approach bears a certain degree of phenomenology. In general, a quasi-static model is constituted of maps or equations describing a predetermined I/O correlation. Those maps or equations are typically embedded into a simplified framework of mostly thermodynamic, mechanical, or electrical relationships while allowing for comparatively small time steps (e.g. in the range of one combustion cycle). For example: An internal combustion engine could be represented by a number of maps (for power output, fuel consumption, friction, emissions, heat rejection, etc.) which are defined for a range of engine torque or speed points, or any other kind of dependency (e.g. temperatures, pressures, etc.). Amended by further maps (for transmission and drive shaft) or marine engineering equations (for ship resistance, hull, and propeller modeling) it is relatively simple to create a quasi-static model for longitudinal ship propulsion simulation (Pustode et al., 2020).

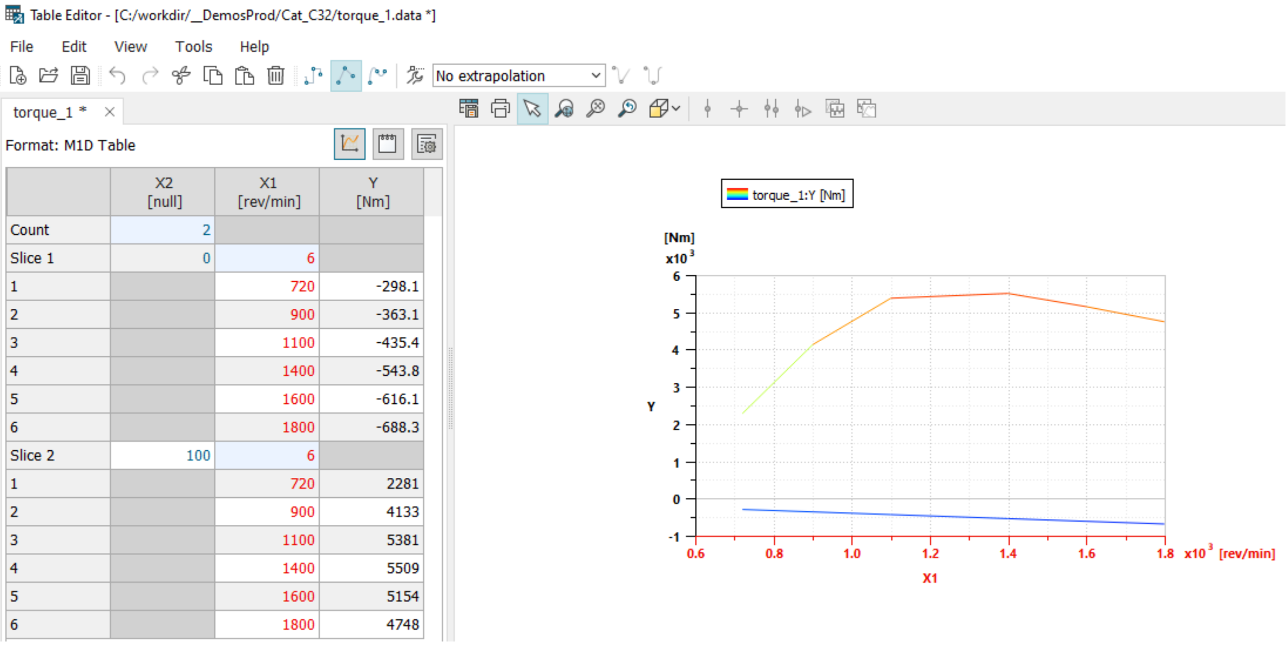


Table Editor - [C:/workdir/__DemosProd/Cat_C32/torque_1.data *]

No extrapolation

torque_1 *

Format: M1D Table

| | X2 [null] | X1 [rev/min] | Y [Nm] |
|---|---|---|---|
| Count | 2 | | |
| Slice 1 | 0 | 6 | |
| 1 | | 720 | -298.1 |
| 2 | | 900 | -363.1 |
| 3 | | 1100 | -435.4 |
| 4 | | 1400 | -543.8 |
| 5 | | 1600 | -616.1 |
| 6 | | 1800 | -688.3 |
| Slice 2 | 100 | 6 | |
| 1 | | 720 | 2281 |
| 2 | | 900 | 4133 |
| 3 | | 1100 | 5381 |
| 4 | | 1400 | 5509 |
| 5 | | 1600 | 5154 |
| 6 | | 1800 | 4748 |

Figure 21: Example of Caterpillar C32 engine torque curve

*Application*

When conducting proof of concept studies and high-quality surrogate models are not available, the quasi-static model approach appears to be the right choice. With respect to virtual system integration, this, particularly, applies to the investigation of the system topology for a given ship type or operational profile. In this regard, the CAPEX/OPEX trade-off can be determined through optimizing component choice and high-level energy management control strategy. Main criteria are usually fuel consumption, and greenhouse gas and pollutant emissions.

This approach offers good accuracy at a reasonably high computational speed – multiple concepts can be tested and compared for a variety of operational profiles. Thus, virtual system integration may support the decision making in the early concept phase – provided the decision is purely derived from a CAPEX/OPEX perspective. However, hybridized propulsion systems may also offer functionalities that only become apparent in transient system operation and in conjunction with a more detailed controls modeling (e.g., transient M/E support, dynamic positioning, spinning reserve/redundancy, etc.) – for this purpose, the so-called dynamic model approach seems to provide a more viable solution.

*Challenges*

Similarly like the surrogate models, the quasi-static approach builds upon pre-defined data for forming the above-mentioned maps or equations. Therefore, providing high data quality is one of the main challenges – especially since collecting the data maps or equations for each component type and variant can become difficult. This also touches upon an inherent issue of system integration, that is, the knowledge or data sharing among the component suppliers. Establishing trusted ecosystems of partners may facilitate the data exchange but at the same time limit the CAPEX/OPEX optimization potential. In this regard, a component-agnostic approach might be more expedient.

Furthermore, when optimizing fuel consumption via a quasi-static approach high-level energy management plays an important role (as it can be seen in chapter 2.2.3). Varying operational profiles, physical component limits, or battery aging are additional influencing factors. Thus, development and integration of an energy management control is another key challenge.

*Level of predictivity*

The level of predictivity of quasi-static models is usually low. Even though the input data used in the model are of high quality, a lot of side phenomenon are usually neglected in quasi-static models. For example, an engine quasi-static model cannot “predict’ the turbocharger lag effect; this must be tuned manually based on measurements or high-fidelity simulation results.

### 4.2.4 Dynamic Model Approach

*Description*

In contrast to surrogate or quasi-static models, the dynamic model approach attempts to represent the physics of a given system. To this end, it employs theories like fluid mechanics, thermodynamics, or chemical reaction kinetics. For example, an internal combustion engine is typically modeled in a (quasi-) 0D/1D-CFD domain while building upon the conservation of momentum or energy, the continuity equation, or the Courant criterion. The arguably biggest advantage of dynamic models is their predictivity. After calibration, a dynamic model should be capable of predicting the model behavior even outside of the calibrated range – at least up to a certain extent. Depending on the level of integration (e.g., system controls coupling) and the degree of predictivity of the system- and its submodels even the transient system response may be replicated. Of course, such modeling depth requires noticeable computational effort – i.e., typically, such dynamic model approaches may not run at real-time or faster. However, since run-time speed varies depending on the requested time step size real-time capability may be achieved at the expense of result accuracy.

*Application*

In the last decade, increasing computational power has led to a broad deployment of the dynamic model approach within the scope of combustion engine or hybrid system simulation (Baratta et al., 2013; (Strasser, Hrauda, Wurzenberger, Roduner, & Valero-Bertrand, 2015); Winke et al., 2015; (Strasser, Schönbacher, & Flagmeier, Optimizing Marine Hybrid Propulsion Systems by Multi-Domain System Simulation, 2019)). This development may have been supported by the current trends of front loading and digital twinning, and, of course, by the recent trend of virtual system integration. In this regard, surrogate or quasi-static approaches provide enough model fidelity for conducting CAPEX/OPEX analyses whereas the dynamic approach further helps to reveal the hidden benefits of a hybrid propulsion system (e.g., support of transient engine behavior, dynamic positioning, spinning reserve, port operation, power boosting, etc.). Furthermore, it is only the dynamic approach that will enable an early verification of system operability within a virtual setup – there are different commercial or joint industry project platforms that facilitate such investigations (e.g., Open Simulation Platform, Siemens Simcenter Amesim, GT xLink, AVL Model.CONNECT$^{TM}$, etc.). Conclusively, the deployment of surrogate, quasi-static, and dynamic models is not mutually exclusive but rather complementary.

*Challenges*

Since the dynamic model approach requires detailed technical information for creation of the hybrid propulsion system including its submodels (e.g., technical drawings, controls models, model libraries, etc.) the system integrator typically struggles to collect such data from the different component suppliers. Usually, concerns about the protection of intellectual property are hindering an open exchange of data and/or models (see also chapter 7). And even if models are safely shared, technical experience of each element in the system is still mandatory for their proper integration. For example, a sudden load request is responded differently by a diesel or a gas engine, their limits and boundaries are depending on ambient conditions and on the system's topology. Similarly, batteries are facing restrictions based on its SOC or temperature management. Thus, developing a truly integrated system such details have to be known, understood, and accordingly considered.

Other more technical issues are related to the increased computational effort, the coupling of different subsystems and the corresponding inefficiencies when it comes to bug fixing, interfacing, or result analysis.

*Level of predictivity*

As mentioned above, the inherent predictivity of dynamic models is one of their key advantages. Therefore, compared to surrogate or quasi-static models predictivity is generally given and can reach different levels depending on how much phenomenology is incorporated within the system's submodels. The more phenomena are considered in these models the more cross-effects may be captured resulting in an overall improved system behavior prediction.

### 4.2.5 Backward and Forward Approach

When embarking on the topic of modeling fidelity also the dimension of calculation direction has to be discussed. Generally, there are two calculation directions to be considered, namely the 'backward' and 'forward' simulation (omitting here any further discussion on the so-called 'forward-backward'). Various publications (Chan et al., 2009; Delavaux et al., 2010; Hofman et al., 2011) are already addressing this matter so in the following only a brief overview is given.

The 'backward' approach calculates from a certain (vessel/vehicle) speed request back to the actual power that the propulsion system has to deliver. On the contrary, the 'forward' method calculates from the engine or respectively the energy source forward to an output torque and eventually to the vessel/vehicle speed. Due to this setup and for the purpose of following the requested vessel speed profile a 'driver' controller has to be applied.

With respect to virtual system integration, both approaches may further utilize the acting forces (e.g., from air or water resistance) to determine a propulsion torque request or directly impose the requested torque to the propulsion system. The latter is recommended if no detailed vehicle or ship model is available.

Principally, the direction does not define the modeling fidelity itself though there might exist combinations that appear to be more reasonable. This might be particularly the case in the maritime industry where 'drive cycles' are not existing but rather operational profiles for various maneuvers and voyages. The longer the operational profile the bigger is the importance of fast run time speed. Thus, typically a backward approach is combined with surrogate or quasi-static models within the scope of concept studies, topology optimization, or optimal control/dynamic programming. When moving towards more dynamic controls phenomena (e.g., PTI/PTO control, dynamic positioning, peak shaving, etc.) the forward modeling approach in conjunction with dynamic (physical) models may be a suitable choice.

# 5 Approaches for Simulation of Overall Systems

The decision on which approach to choose depends on various aspects, such as the complexity of the system to be described by the simulation, the capabilities of the modeling and simulation tools, the degree of interdisciplinary and cross-company collaboration and others. Therefore, defining the approach to perform the simulation tasks as efficiently as possible is an important step in the planning phase of a simulation project.

It should be taken into account that most suitable tools are used with which the components and domains can be well described and with which the simulations can be reliably carried out. An advantage is an approach in which all partners involved can use their existing infrastructure without having to rely on additional proprietary and commercial software.

In order to be able to exchange models easily, a standard and interfaces must be defined. Ideally, all available models should be able to be integrated into the simulator. This requires that the individual tools support a defined standard. In the following, different approaches are described that are characterized by the use of different types of tools, which are:

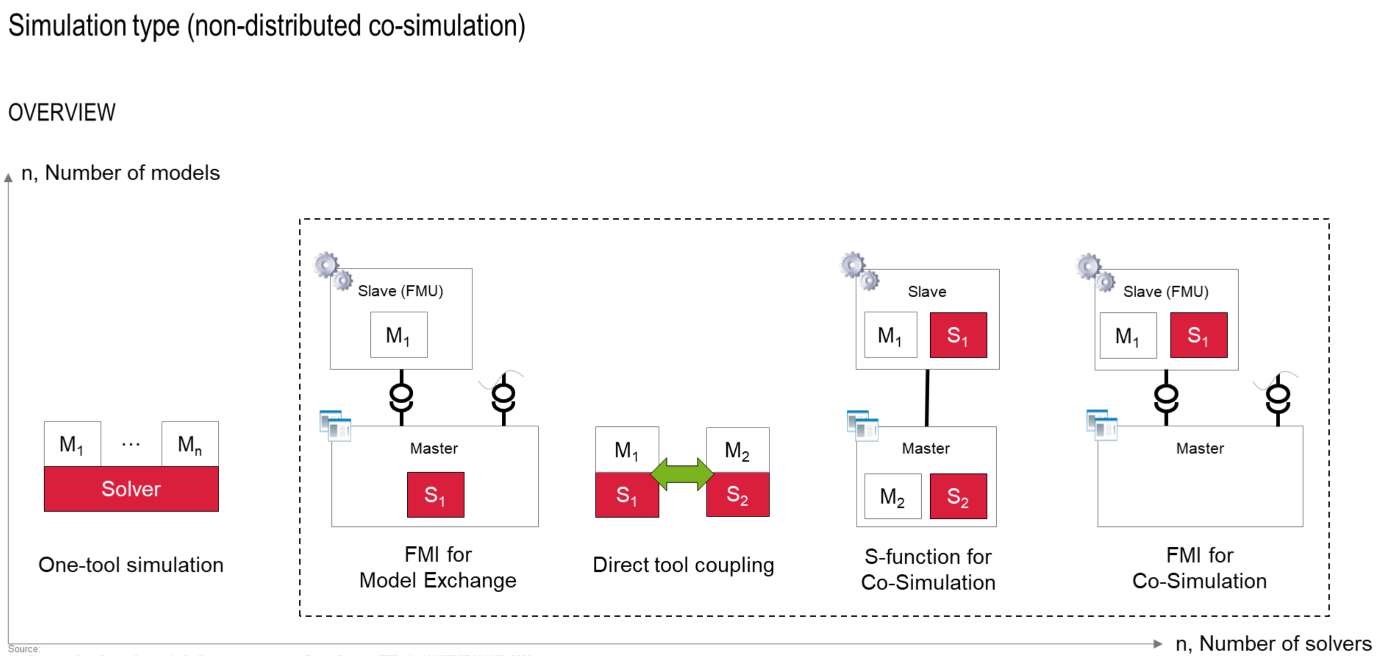
**Modeling tools**: Software for the purpose of creating simulation models.

**Simulation tools**: Software for the purpose of carrying out simulations.

(Most modeling and simulation software contain both classes of functionality, and the terms are therefore sometimes used interchangeably.)

**Co-simulation platforms**: Software for the purpose of facilitating co-simulations.

Based on the way the different tools are used and the requirements, a distinction can be made between different methods of system simulation.



Source:
- Towards a Classification Scheme for Co-Simulation Approaches in Energy Systems, IEEE, 10.1109/SEDST.2015.7315262
- Cosimulation of intelligent power systems: Fundamentals, software architecture, numerics, and coupling, IEEE
- Wesentliche Eigenschaften von Co-Simulationsumgebungen und deren Vergleich, KIT
- FMI/FMU R&D projects (e.g. ConneCDT), FEV/VKA
- Final report, ACOSAR

*Figure 22: Simulation type (non-distributed co-simulation)*

The consistent and seamless application of virtual system integration in the development process with different degrees of virtualization helps to save development time and costs and to increase safety and quality. Using the example of an engine development, a functioning engine control system is ideally already available when the performance & emission development is to be started on the test bench. For maritime hybrid applications, this means that system integration begins well before construction and is completed before the ship is launched, rather than having to adjust and calibrate controls during sea trials.

The main reason for using a one-tool solution is usually simplicity. By having just one tool users need only to be trained once and get more confident with increased usage. In general license costs are lower (except if individual libraries are charged separately) and purchasing need to work with only one supplier. Additionally, maintenance is also simpler for the IT department. Especially if one tool is already established at a company it will usually be much faster to get started with the existing infrastructure than introducing additional tools or processes.

If a tool is designed for multi-domain simulations, the solvers are tightly coupled with integrated stability and time stepping schemes which leads to faster and more robust solutions.

The main downside of using just one tool is that not all domains might be equally well covered, and some specialized features might be missing.

Generally, there is a trend of tool suppliers becoming vendors of complete simulations platforms. This trend is supporting the “one tool solution” whereas on the other hand system integration has an inherent tendency to co-simulation.

## 5.1 Co-Simulation

The reasons for conducting a co-simulation-based approach are manifold. In general, and in contrast to the other simulation approaches co-simulation allows to use a tailored simulation software for each individual application need, in terms of the model’s level of fidelity, the applicability of the solver type and finally the preference and the existent experience of the nominated engineer(s).

### 5.1.1 Direct coupling approach

In addition to that the direct coupling approach addresses the fine adjustment of the interface between the mated simulation software, resulting in a superior solver robustness and simulation result quality on the one hand, but implying a predefined convention for the interface, e.g., regarding the type of application or the exchanged variables. In general, one may expect a time overhead with respect to a compliant setup of an already existent or the development of a novel co-simulation interface.

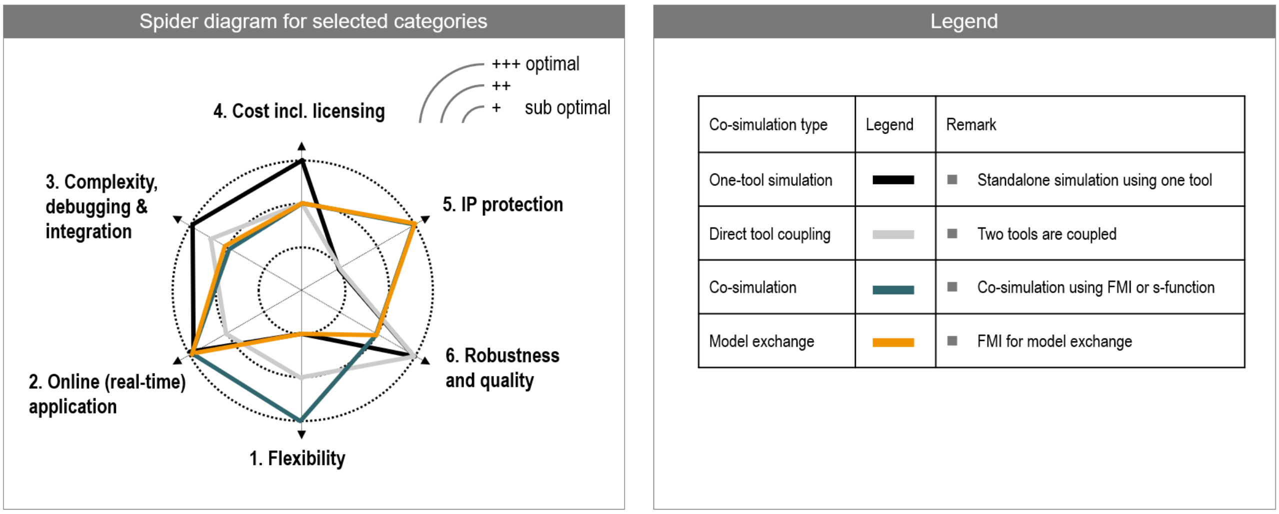
It is a common practice that one simulation software is assigned the task to lead the co-simulation process (initialization of the interface, call to the mated simulation software, handshake, variable value exchange, event messages, etc.)

### 5.1.2 FMI for Co-Simulation

FMI for Co-Simulation (FMI-CS) connects the master solver component with one or more slave solvers. The simulation tools are coupled in a co-simulation environment. In this specification, both the model and a solver are encapsulated inside the FMI-CS. The data exchange between the different subsystems is restricted to discrete communication points; the subsystems are solved independently from each other by their individual solver. The data exchange between subsystems, as well as the synchronization of all simulation solvers (slaves), are controlled by the master algorithms. With FMI-CS, each domain can be ideally covered by its specialized tool. However, the number of licenses necessary for the simulation part is relevant and in case of loosely coupled solvers, data exchange may occur only for major time steps. Despite the sensible increase of efforts for integration and setup of the multi-domain environment, the flexibility obtained is enhanced. Real-time application with this interface is allowed.

## 5.2 Model Exchange

With FMI for Model Exchange (FMI-ME), C code of a dynamic system model is generated by a modeling environment to be utilized by other modeling and simulation environments. It consists of the model description interface and the model execution interface. It connects the external model component with the solver component. The environments importing FMI-MEs need to provide an integrator, or an ODE solver that integrates the dynamics of the model, but the imported model, distributed in one zip-file called FMU (Functional Mock-up Unit), is independent of the target simulator. FMI-ME is tool-independent, and its standardized approach facilitates the re-use of models providing less efforts and good quality. Despite the loss of the advantages related to the usage of a specific solver, when models are to be shared between different organizations with commercial interests, model information protection is granted. When parts of a model are hidden, or protected, information required to keep the model usable can increase. Particular attention is required in case of mismatch between model complexity and solver capability. For FMI-ME, real-time application is also possible, but with some drawbacks in terms of flexibility.



| Co-simulation type | Legend | Remark |
|---|---|---|
| One-tool simulation | ⬛ | ▪ Standalone simulation using one tool |
| Direct tool coupling | ⬜ | ▪ Two tools are coupled |
| Co-simulation | 🟦 | ▪ Co-simulation using FMI or s-function |
| Model exchange | 🟧 | ▪ FMI for model exchange |

*Figure 23: Advantages and disadvantages of co-simulation approaches*

**Categories in Spider diagram:**

**1. Flexibility:** Flexibility around model integration regarding different solver types and definition of interfaces.

**2. Online (real-time) application:** Capability to embed and run the model in real-time applications.

**3. Complexity, debugging & integration:** Task Complexity around the model preparation, know-how build-up, documentation and troubleshoot.

**4. Cost incl. licensing:** License costs for the generation and application of FMUs both for offline and online application.

**5. IP protection:** Protection of intellectual property regarding model content and the solution approach.

**6. Robustness and quality:** Robustness of solution approach and quality of results which results from the programmatical harmonization.

## 5.3 Model-in-the-Loop, Software-in-the-Loop

Since in complex systems - such as hybrid applications - the optimal interaction of the components and the enabling of the optimal operating strategy for a given operating scenario play a major role, the development of control systems is becoming increasingly important.

Model-in-the-Loop (MiL) as well as Software-in-the-Loop (SiL) are frontloading methods that are already state of the art in the automotive sector and are increasingly finding their way into other areas as well.

In contrast to the simulation approaches described at the beginning, they are characterized by different degrees of virtualization, as shown in Figure 5. Nevertheless, they should be mentioned here, especially since they are also types of virtual system integration and the models used in pure system simulation can be reused. The intention is to reduce physical testing by frontloading the development to the virtual environment.

MiL and SiL are mainly used in control system development and numerous tasks can be performed virtually, such as function development, algorithm development, control strategy development, software check or even pre-calibration. Controls development is performed using appropriate development platforms (e.g. Matlab Simulink). Therefore, the modeling tool should either be able to provide the plant models in a suitable form (e.g. Simulink S-functions, FMUs) in order to integrate them into the development environment, or vice versa: the control models / control software can also be integrated into the system simulation environment.

Special attention should be paid to the correct dynamic behavior of the plant models in order to develop the control loops in the best possible way. In addition, the actuator and sensor channels (input and output channels, sample frequency) of the plant models must be available according to the requirements of the control models. The interface definition is of particular importance here.

## 5.4 Hardware-in-the-Loop, Virtual Testbed

Besides controls development, the HiL approach is mostly used for calibration tasks (drivability/operability, base calibration, emission calibration, diagnostics and monitoring calibration, robustness & tolerance calibration, final validation).

The key components for the virtualization of the testing environment (see Figure 24) are real-time capable simulation models, the automation system and exchangeable tools and test-procedures between virtual and physical lab.

Ideally, those models that are used in MiL & SiL environment can also be used on virtual testbeds or HiL test systems. However, especially the integration of hardware components into the simulation environment is a special challenge for the models: Whereas for MiL investigations real-time capability is not a mandatory requirement, the models which should run on the HiL system must be real-time capable when operated with hardware components in order to avoid real-time violations and breakdown of the HiL system. It is imperative that the calculation of each time step is completed as fast as or faster than the physical time period so that the control unit reliably receives signals corresponding to the specified sampling frequency. The need for real-time capability should definitely be considered when determining modeling depth and modeling fidelity. It must be ensured that all those effects and phenomena are represented by the models which are of importance for the given task. A (multi-domain) modeling tool is required for the creation of models which fulfil the requirements in view of model fidelity and real-time capability and which can be compiled for the real-time target. Multi-core real-time PCs and coupled virtual testbeds enable multitasking and parallel simulation of models (engine, aftertreatment system, battery, fuel cell, etc.) to take advantage of the HiL methodology, especially for complex systems.

The consistent and seamless application of virtual system integration in the development process with different degrees of virtualization helps to save development time and costs and to increase safety and quality. Using the example of an engine development, a functioning engine control system is ideally already available when the P&E development is to be started on the test bench. For maritime hybrid applications, this means that system integration begins well before construction and is completed before the ship is launched, rather than having to adjust and calibrate controls during sea trials.

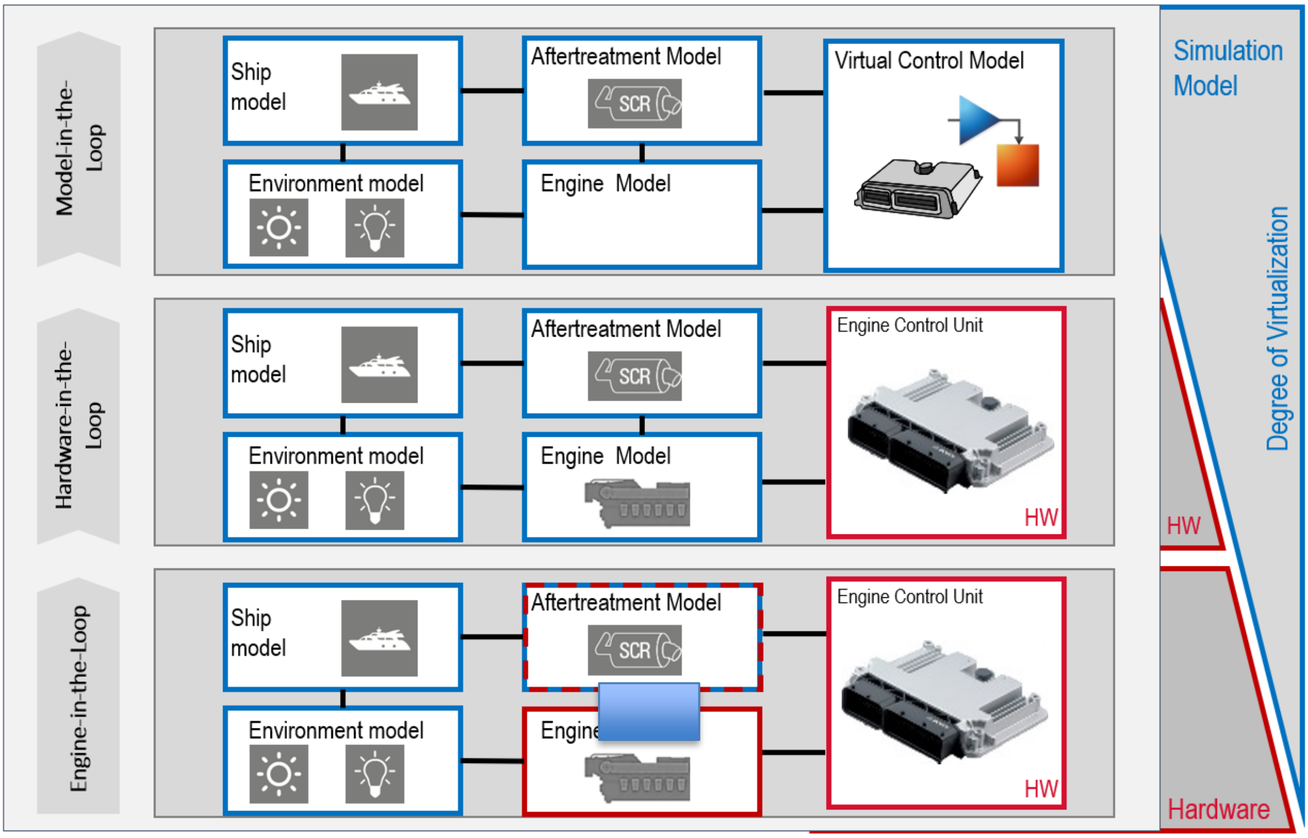


*Figure 24: Degree of virtualization (example: engine development)*

# 6 Model Interfaces

Since systems generally consist of several components, units or subsystems, there is the need for discussion about model interfaces resulting from the different ways of simulating complex systems. The question is how the individual models communicate with each other, what the signal flow looks like and which input and output channels the individual models have to provide. Both administrative and technical aspects must be considered.

Interfaces must be carefully defined especially if

- different teams and persons are involved in the modeling process of components or if the simulation should be performed as a cross-company collaboration.
- multi-domain system simulation is required.
- several suppliers and stakeholders are providing submodels which are to be incorporated into the overall system model.
- virtual and real components should be coupled (e.g. control-system development by means of hardware-in-the-loop investigations, virtual testbed operation).
- advanced co-simulation should be performed.

The parties performing co-simulation need to agree on

- Modularization of systems (definition of system boundaries of subsystems)
- Signal structure (definition of input and output channels which have to be provided by the single sub models)
- Naming convention for input and output channels allowing easy handling of the models and the coupling process
- Coordinate systems
- Units (standardization, e.g., agreement on strict usage of a universally valid unit system such as SI unit system)
- Embedding of control and monitoring systems

It is recommended to nominate a responsible person in the project team coordinating the definition of the interfaces and checking the compliance with the agreed model interface definition. The use of appropriate forms for the individual models are very helpful in this respect.

**System simulation and monitoring of energy and propulsion system of ships**
A wide range of data is measured and collected to monitor the performance and condition of components and subsystems on a ship. It would be advantageous if this collection of measurement and operational data would take into account the requirements of system simulation. There are signal flows, energy flows and information flows between the individual components of the ship, which are also important for the model-based consideration of the overall system and whose measurement can be made usable for the simulation. The availability of corresponding measurement and operating data supports the simulation in two respects:

- Model creation: The measurement and operating data can be used for model parameter calibration in order to best represent the physical properties of the components and systems.
- Validation of models: The comparison of measured with simulated data can be seen as a measure of quality assurance. The predictive accuracy of the models can be increased if

attention is paid to model validation. This will allow improved models to be available for further simulation tasks and also for follow-up projects.

The definition of the available data channels as well as the definition of the channel names, the sampling rates and units should be aligned with the requirements regarding data availability for system simulation. Besides, data harmonization simplifies the prerequisites for certain use cases of model-based approaches, such as virtual sensors, simulation-based operational optimization, or the use of model-based assistance systems on board ships and in fleet control centers.

At this point reference should be made to the Guideline “Monitoring Systems for Marine Hybrid Propulsion Systems” of CIMAC WG20. The Guideline provides an overview regarding the implementation of performance monitoring systems for Marine Hybrid Propulsion Systems. With system integration gaining importance for the industry to improve energy efficiency, the objective of the publication is to contribute to the development and promulgation of multi-source energy system design optimizations for ships and land-based power plants.

## 6.1 FMI Standard

Different possibilities have been used in the past for coupling models and for performing co-simulation. One standard has emerged as particularly useful and has therefore been successfully used in various industries for some time, such as in the automotive industry and should also be advantageous for large engine and marine systems: The Functional Mock-up Interface (FMI) is a free standard that defines a container and an interface to exchange dynamic models using a combination of XML files, binaries and C code zipped into a single file.

One big advantage is that FMI is supported by more than 100 simulation tools and it is under constant development, maintained as a Modelica Association Project. Detailed information can be found at https://fmi-standard.org/. The CIMAC partners can check the compatibility of their used simulation tools with the FMI standard (export & import of FMU models).

**Description and properties of FMI standard:**

- A model which implements FMI is called Functional Mock-Up Unit (FMU).
- FMU is an archive file (zip format) consisting of
  - model code for one and more platforms (C or binary),
  - a description of the interface data (xml format) and
  - optional documentation and metadata.
- FMI standard specifies the APIs that must be implemented by the model code.
- FMI for co-simulation is based on master/slave model of communication and control (sub-simulators are slaves controlled by a master algorithm).
- FMI specifies how the co-simulation software interacts with the models; it is not a simulation software.
- FMI 1.0 was released in 2010. This version includes the basic FMU concepts such as co-simulation and model exchange. FMI 2.0 was released in 2014 with major enhancements and additional features (e.g., parameters can be changed during simulation; the complete FMU state can be saved, restored and serialized; directional derivatives with respect to states and inputs can be computed, the structure of the partial derivatives with respect to states and inputs can be given). FMI version 3.0, released in 2022, offers new features that enable the use of FMI in important new use cases: advanced co-simulation, virtual Electronic Control Units (vECUs), the next generation of Digital Twins, artificial intelligence, and autonomous driving applications.

**Benefits**

- There is no need for recompilation
- FMI standard facilitates model sharing and co-simulation.
- The FMI standard is completely open and free to use.
- It supports a large and growing number of simulation tools.
- FMI is continuously and actively refined.
- FMI can also represent interfaces to hardware (sensors, actuators, human-machine interface).
- It is already standard for co-simulation in some areas (e.g. automotive industry) and well approved.

**Limitations**

- Sub-simulators (FMUs) do not have any information neither about each other nor about the simulation environment.
- There is no knowledge about control or about to which sub-simulators they are coupled (data routed by master algorithm).
- FMI does not specify how the sub-simulators are time synchronized.
- FMI does not specify in what format data are transferred between the sub-simulators.
- Error control is not addressed by FMI.
- How can discontinuous models be encoded as FMUs?
- Non-deterministic and unexpected behavior of models is theoretically possible.
- FMI does not enforce logical checks.
- Physical connections and acausal connections or not yet supported.

## 6.2 Documentation of Model Properties and Interfaces

Sufficient documentation and description of models is essential for effective project work and to minimize the effort involved in coupling models in a simulator or co-simulation environment. Such a model description should be handed out together with the simulation model and should contain all relevant information needed for the application of the model.

**Necessary components of a documentation and model description**

- Program version of modeling tool
- Field of application of model / intended use case
- Operating areas in which the model can be operated and is also validated.
- Proof of validation available
- Description (main data) of system/subsystem/component covered by the model
- Description of degree of modeling depth/modeling fidelity/modeling approach
- Simulation step size
- Topology of simulation model
- Channel description (input and output channels)
    - Names
    - Units
    - Range
    - Additional comments
    - …
    - …
- …

| No. | Input Signal | Unit | Recommended Range | Comment |
|---|---|---|---|---|
| 1 | T_Amb | °C | -40 – 50 | Ambient temperature |
| 2 | P_Amb | bar | 0.8 – 1.1 | Ambient pressure |
| 3 | ModeTorqueSpeed | - | 0 / 1 | Mode = 0 („Torque control mode“): Load is the input to the system. (The speed is calculated under consideration of engine torque and inertia of the system.)<br>Mode > 0 („Speed control mode“): Speed is the input to the system. The engine torque is calculated from the injected fuel at the demand speed. |
| 4 | SpeedDemand | rpm | 30 – 120 | Input of engine speed in „Speed control mode“ |
| 5 | Load | Nm | 0 – 1200000 | Input of engine load in „Torque control mode“ |
| 6 | Fuelling | g | 20 – 70 | Fuel mass per cycle to be injected |
| 7 | Eta_CAC | - | 0.5 – 1 | Hot effectiveness of the charge-air cooler: The outlet temperature of the charge-air cooler is calculated from the inlet temperature and the hot effectiveness. |

*Table 4:* Example of input channel description of engine model (excerpt)

| No. | Output Signal | Unit | Comment |
|---|---|---|---|
| 1 | Speed | rpm | Actual averaged engine speed |
| 2 | BMEP | bar | Brake mean effective pressure |
| 3 | Power | kW | Averaged power of engine |
| 4 | Torque | Nm | Averaged torque of engine |
| 5 | BSFC | g/kWh | Brake specific fuel consumption |
| 6 | IMEP | bar | Indicated mean effective pressure |
| 7 | FMEP | bar | Friction mean effective pressure |
| 8 | MassFlowAir | kg/h | Overall air-mass flow |
| 9 | MassFlowAir_1 | kg/h | Air-mass flow of intake branch of turbocharger 1 |
| 10 | MassFlowAir_2 | kg/h | Air-mass flow of intake branch of turbocharger 2 |
| 11 | MassFlowFuel | kg/h | Averaged fuel-mass flow of engine (6 x cylinder 1) |
| 12 | Lambda_SoC | - | Excess-air coefficient of exhaust gas |
| 13 | Lambda_Testbed | - | Excess-air coefficient calculated from overall air-mass flow and total fuel-mass flow |
| 14 | FuelMassPerStroke | g | Fuel mass per cycle of one cylinder |
| 15 | SoC | deg | Crank angle at start of combustion |
| 16 | MFB5% | deg | Crank angle at mass fraction burned 5 % |
| 17 | MFB50% | deg | Crank angle at mass fraction burned 50 % |

*Table 5:* Example of output channel description of engine model (excerpt)

**Documentation of model quality:**

- Report of passing quality gates
  - Simulation of steady-state operation
  - Simulation of transient and dynamic behavior (e.g., load steps)
- Graphical comparison of measured and simulated data
- Confidence intervals, hypothesis tests

# 7 Challenges of Co-Simulation & Advanced System Simulation for Complex Systems

Hybrid marine propulsion systems are typically characterized by high system complexity due to multiple energy systems and different domains. Multi-physical system simulation is generally a very useful methodology for propulsion concept definition of hybrid systems for all vessel types as the interactions between the various subsystems can be investigated in an ideal way; however, the complexity of the systems is also reflected by the complexity of the simulation environment and there are many challenges. In order to use co-simulation successfully for tasks regarding system integration some aspects should be considered before starting to set up a simulation environment and to create models. Based on experience with virtual system integration several aspects and items are listed in the following. These items have a great impact on the simulation approach, the modeling concept, the procedure and the execution of a project. They are - depending on the respective application - more or less worthy of consideration. By observing these points, the procedure and simulation approach should be chosen correctly, additional coordination and simulation loops should be avoided and the efficient, technical achievement of objectives should be ensured:

- **Coordination and planning of cooperation:** Since many partners are involved in the modern shipbuilding process, several partners (e.g., shipyard, OEMs, propulsion-system suppliers, battery suppliers, system integrator, control-system provider) are also involved in the implementation of virtual system integration. The effort for the coordination and planning of the cooperation should not be underestimated. It has to be checked which component models can be made available by the partners in time. In addition, the model requirements and model interfaces need to be defined.
- **Model quality & simulation stability:** The relevant domains and subsystems need to be described by appropriate models, such as the propulsion line, ship hull, engine thermodynamics, electric domain, cooling and lubrication system, hydraulic domain and control system. To ensure the quality of the models and the simulation stability, the component and subsystem models should be carefully validated and tested before used in the simulation environment. „Over-optimistic“ models and models with insufficient accuracy should be avoided; as the models would affect the overall system performance and might lead to wrong results and wrong decisions. Hence, the models should be validated by comparing the simulated performance with the test results of component tests if available. Besides, it should be checked whether the available models are suitable for the intended application regarding modeling depth, significance of relevant component behavior and the required operating range.
- **Control domain:** In most cases, the physical behavior of a subsystem is sufficiently well represented by a model, but it is often forgotten to consider the control characteristics of a component specific control system. The data basis for the creation of models is mostly based on steady-state measurement results; however, especially for transient operation the component’s control strategy may influence the component’s and, therefore, the system’s behavior significantly. This is done primarily for component protection or to comply with emission limits under given boundary and operating conditions (e.g., smoke limiter of engine-control system, thermal protection of battery by battery-management system). Reliable control system models of components are sometimes missing, but it is mandatory to consider

important control strategies and algorithms as the simulation of transient system performance is one objective of system simulation.

- **IPR protection:** It should be noted that models of partners may also contain information that is confidential and secures competitive advantages. Therefore, on the one hand, one should be aware that models may not be made available by partners for reasons of IPR protection. Even if models are black boxes (e.g., FMUs or other models where code is compiled), it would be possible to draw conclusions by reverse engineering. Sensitive handling of this topic is advisable and non-disclosure agreements shall be signed.
- **Holistic versus causal approach:** In many cases, with proprietary modeling tools a subsystem can be assembled from predefined model components under consideration of the mechanical, electrical, or logical connections. The differential equations generated at simulation start allow the holistic description of the system behavior. If, on the other hand, the model is built as a cause-effect chain, then interactions between subcomponents might not be considered. Therefore, it is essential to ensure during modeling that all relevant energy, heat or information flows between the individual component models are mapped and that algebraic loops that affect behavior are avoided. Even if models are linked together as part of a co-simulation, there is a risk that important interactions are not considered. Therefore, the interface definition for the single models should carefully consider which energy, mass, heat and information flows are of importance and the corresponding input and output channels should be provided.
- **Computational effort versus modeling depth / real-time capability:** The system complexity as well as the modeling depth influence the computational effort and duration of the simulation. If long ship voyages and extensive operational profiles are to be simulated, the computing time should also be taken into account. The question arises, which modeling depth is required to perform the given tasks and which added value results from higher model depth. In case real-time capability is mandatory for virtual testing and hardware-in-the-loop (HiL) runs a special focus has to be set on the simulation speed as hardware (e.g., control units) must receive reliable signals from the simulation after each time step (frequency of 0.1 - 10 Hz dependent on application). Crank-angle resolved engine simulation and simulation of electric system in frequency domain might be challenging and an appropriate computer hardware and HiL equipment is needed. For merchant ships with main engines running at speeds below 130 rpm, real-time capability of crank angle resolved simulation for HIL application can be easily achieved, while real-time capability for high-speed engines depends on model complexity.
- **Coupling of mechanical subsystems:** In case mechanical subsystems are not coupled directly via mechanical connections, some properties of the mechanical system might not be treated in the correct manner. One example is the consideration of the rotational inertia when coupling mechanical subsystems, such as engine, generator, shaft, or propeller: It must be ensured that the principle of angular moment is solved under consideration of the correct inertia. Especially when submodels (e.g., FMUs) are coupled and simulated in a co-simulation platform, this fact must be considered. By analogy, electrical impedances and thermal inertias must also be considered carefully when direct electrical or thermal connections are not possible.
- **Effort for model creation and setup of co-simulation environment:** The effort for performing system simulation can seem quite high, however, this should be contrasted with the high benefit (e.g., specification of components tailored for given operating profile, reduction of hardware testing effort, optimization of operating strategy). Added value should be created by using system simulation and model-based development approaches

consistently and seamlessly throughout the entire development process and beyond by model re-use.

- **License management:** One decision criterion is the license costs incurred. Licenses are usually required to execute the simulation of models created by means of proprietary software tools. If co-simulation with models from multiple sources should be performed, considerable costs can be incurred. There are also often costs associated with executing the simulation of FMUs and it should be checked if the execution of FMUs compiled by software tools are license-free.
- **Numerical treatment of co-simulation**: The integrated co-simulation system should work fast and accurately but integrating the models from different domains in a consistent co-simulation system is highly complex. Different software techniques such as advanced coupling & synchronization techniques, error elimination, parallelization, optimal time-step/solver management and smart scheduling have to be combined and should be supported by the chosen co-simulation platform.
- **Documentation**: As in complex co-simulation multi-disciplinary teams and many engineers are involved, the availability of sufficient documentation and model description is strongly recommended.
- **Troubleshooting:** When numerous submodels are embedded into a co-simulation environment, it can be difficult to debug single submodels and FMUs. Hence, the submodels should be validated and checked in a release process by means of appropriate tools to support the debugging in an early phase. This can avoid the need for troubleshooting when errors or unexpected behavior occur. This can be done by means of a release process with quality gates or to entrust one team member with the quality check and the model validation.
- **Clear arrangement and structure of models and simulation environment:** It might be a challenge to keep track of large system simulations with complex overall system topology. Therefore, a clear arrangement and structure of the models is recommended.
- **Software & hardware requirements:** An appropriate infrastructure (software, hardware, data lines and connections) is needed for successful simulation project execution. Fast simulation, support of multi-threading, support of distributed co-simulation are criterions which define the requirements for the simulation infrastructure.
- **Experienced simulation engineers:** Even with the very best simulation tools, good results can only be achieved if you have experienced simulation engineers on your team. The choice of input data, the making of necessary assumptions and the definition of boundary conditions determine the quality of the results and should therefore be carried out by experienced engineers. The results should always be questioned with common sense.

# References

Beşikçi, E. B., Arslan, O., Turan, O., & Ölçer, A. I. (February 2016). An artificial neural network based decision support system for energy efficient ship operations. *Computers & Operations Research 66*, S. 393-401.

Göknur, S., Paredis, C. J., Yannou, B., Coatanéa, E., & Landel, E. (2015). A Model Identity Card to Support Simulation Model Development Process in a Collaborative Multidisciplinary Design Environment.

Lazakis, I., Raptodimos, Y., & Varelas, T. (March 2018). Predicting ship machinery system condition through analytical reliability tools and artificial neural networks. *Ocean Engineering, Volume 152*, S. 404-415.

Olle Sundström, L. G. (2009). A Generic Dynamic Programming Matlab Function. Saint Petersburg: 18th IEEE International Conference on Control Applications.

Peng, J., He, H., & Xiong, R. (January 2016). Rule based energy management strategy for a series–parallel plug-in hybrid electric bus optimized by dynamic programming. *Applied Energy 185*.

Ponnusamy, S. S. (2019). Simulation product fidelity : a qualitative & quantitative system engineering approach.

Roe, R. R. (2019). *NASA handbook for models and simulations: an implementation guide for NASA-std-7009a.* NASA.

Roza, Z. C. (2005). Simulation fidelity theory and practice.

Simona Onori, L. S. (2016). *Hybrid Electric Vehicles - Energy Management Strategies.* Springer.

Strasser, R., Hrauda, G., Wurzenberger, J., Roduner, C., & Valero-Bertrand, D. (2015). XiL-Application and Model-Based Development for Large Engines – The Future? Graz: Institute of Internal Combustion Engines and Thermodynamics, Graz University of Technology.

Strasser, R., Schönbacher, M., & Flagmeier, N. (2019). Optimizing Marine Hybrid Propulsion Systems by Multi-Domain System Simulation. Vancouver: CIMAC Congress 2019.

Uslu, S., & Celik, M. B. (December 2018). Prediction of engine emissions and performance with artificial neural networks in a single cylinder diesel engine using diethyl ether. *Engineering Science and Technology, an International Journal*, S. 1194-1201.

Yuan, Z., Teng, L., Sun, F., & Huei, P. (2013). Comparative study of dynamic programming and Pontryagin's minimum principle on energy management for a parallel hybrid electric vehicle. *Energies, 6(4)*, (S. 2305-2318).

Zhang, S., & Xiong, R. (2015). Adaptive energy management of a plug-in hybrid electric vehicle based on driving pattern recognition and dynamic programming. *Applied Energy, 155*, S. 68-87.

Zhang, Y., Xu, S., Zhong, S., & Bai, X.-S. (November 2020). Large eddy simulation of spray combustion using flamelet generated manifolds combined with artificial neural networks. *Energy and AI, Volume 2*.

# List of Abbreviations

| | |
|---|---|
| ANN | Artificial neural network |
| BMEP | Brake mean effective pressure |
| BSFC | Brake specific fuel consumption |
| CAPEX | Capital expenditures |
| DWT | Deadweight tonnage |
| EATS | Exhaust gas aftertreatment system |
| EM | Energy management |
| EMS | Engine Management System |
| FC | Fuel cell |
| FMI | Functional mock-up interface |
| FMI-CS | FMI for co-simulation |
| FMI-ME | FMI for model exchange |
| FMU | Functional mock-up unit |
| GHG | Green house gas |
| GT | Gas turbine |
| HIL | Hardware-in-the-loop |
| ICE | Internal combustion engine |
| LNG | Liquified natural gas |
| M/E | Main engine |
| MGO | Marine gas oil |
| MIL | Model-in-the-loop |
| ODE | Ordinary differential equation |
| OPEX | Operational expenditures |
| PDE | Partial differential equation |
| PTI | Power take-in |
| PTO | Power take-off |
| SG | Shaft generator |
| SOC | State of charge |
| VSI | Virtual system integration |

# Imprint

CIMAC e. V.
Lyoner Strasse 18
60528 Frankfurt
Germany

President: Prof. Dr. Donghan, Jin
Secretary General: Peter Müller-Baum

Phone +49 69 6603-1567
E-mail: info@cimac.com

## Copyright

CIMAC, the International Council on Combustion Engines, was founded in 1951 as a global non-profit association to promote technical and scientific knowledge exchange in the field of large internal combustion engines. CIMAC consists of National and Corporate Members from the Americas, Asia and Europe, and is supported by engine manufacturers, engine users, technical universities, research institutes, component suppliers, fuel and lubricating oil suppliers, classification societies and other interested parties. CIMAC's mission is to promote large engine technology power solutions that are efficient, reliable, safe, sustainable and of benefit to society, in pursuit of the transition to a low-carbon future.

For further information about CIMAC please visit http://www.cimac.com.