Combining semantic web technologies and KBE to solve industrial MDO problems

Raju Kulkarni, Akshay; Hoogreef, Maurice; La Rocca, Gianfranco

DOI
10.2514/6.2017-3823

Publication date
2017

Document Version
Final published version

Published in
18th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference

Citation (APA)
Raju Kulkarni, A., Hoogreef, M., & La Rocca, G. (2017). Combining semantic web technologies and KBE to solve industrial MDO problems. In *18th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference: 5-9 June 2017, Denver, Colorado* Article AIAA 2017-3823 American Institute of Aeronautics and Astronautics Inc. (AIAA). https://doi.org/10.2514/6.2017-3823

AIAA AVIATION Forum
5-9 June 2017, Denver, Colorado
18th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference

10.2514/6.2017-3823

Combining semantic web technologies and KBE to solve industrial MDO problems

Akshay Raju Kulkarni[*], Maurice F. M. Hoogreef[†] and Gianfranco La Rocca[‡]

Delft University of Technology, Delft 2629HS, The Netherlands,

In order to maintain their competitive edge, manufacturers are constantly looking to shorten their product development time, whilst improving product performance. Multidisciplinary Design Optimization (MDO) is a powerful methodology to improve product design. However, in the industrial context, its potential is not fully exploited yet because of both technical and non-technical barriers. The set-up of any MDO problem is generally a complex and lengthy process, which affects the conventional working procedures within the company and requires expertise that is not always available among the ranks of core discipline experts. The application of MDO in a complex product development process is not straightforward and generally iterative. Different disciplines are considered and tools with different levels of fidelity are used in the various phases of the product development. For the use of MDO in each phase, there is a recurring challenge of selecting an adequate architecture for the MDO problem at hand, i.e. to establish the order of execution of the various disciplines and manage their coupling mechanism. Finally, the implementation of the selected architecture into a simulation workflow involves manual, repetitive and time-consuming tasks which are prone to human-errors.

This research work presents a methodology, based on Semantic Web technologies and Knowledge Based Engineering (KBE), to provide advice on the selection of appropriate architectures for the design optimization problem at hand and automatically formalize and implement it in a simulation workflow. The proposed methodology is demonstrated for a relevant industrial case, namely the optimization of the fin-rudder interface, which is fundamental for the design of the rudder, but also affects the design of the fin. The proposed methodology and its technical implementation proved able to help designers in the selection of a suitable MDO architecture and automate most of the manual, repetitive and time consuming efforts, necessary to set up the simulation and optimization framework. Outstanding set up time reductions, up to 90%,were achieved, whilst drastically reducing the possibilities of human errors. While the KBE application embedded in the MDO problem formulation was specifically developed for the presented use case, the technology to advice, formalize and implement MDO architectures into functioning simulation workflows is fully generic and reusable.

Nomenclature

IDF = Individual Disciplinary Feasible
InFoRMA = Integration, Formalization and Recommendation of MDO Architectures
MDA = Multi-Disciplinary Analysis

---

[*] PhD Candidate, Flight Performance and Propulsion chair, Faculty of Aerospace Engineering, Kluyverweg 1, 2629HS Delft, a.rajukulkarni@tudelft.nl

[†] PhD Candidate, Flight Performance and Propulsion chair, Faculty of Aerospace Engineering, Kluyverweg 1, 2629HS Delft, m.f.m.hoogreef@tudelft.nl

[‡] Assistant Professor, Flight Performance and Propulsion chair, Faculty of Aerospace Engineering, Kluyverweg 1, 2629HS Delft, g.larocca@tudelft.nl

MDF = Multi-Disciplinary Feasible
MDO = Multidisciplinary Design Optimization
MS = Margin of Safety
PIDO = Process Integration and Design Optimization
OWL = Web Ontology Language
RDF = Resource Description Framework
SAND = Simultaneous Analysis aNd Design
SPARQL = SPARQL Protocol And RDF Query Language
XDSM = eXtended Design Structure Matrix
XML = eXtensible Markup Language

I. Introduction

In the development of aircraft rudders, the hinge system, also known as the fin-rudder interface (shown in Figure 1) , is a critical sub-system which affects both the fin and the rudder and, as such, forms the interface between the Tier 1 suppliers and the Original Equipment Manufacturers. The current industrial design approach for the fin-rudder interface does not allow the possibility of performing any multi-disciplinary design optimization owing to its long design lead time.

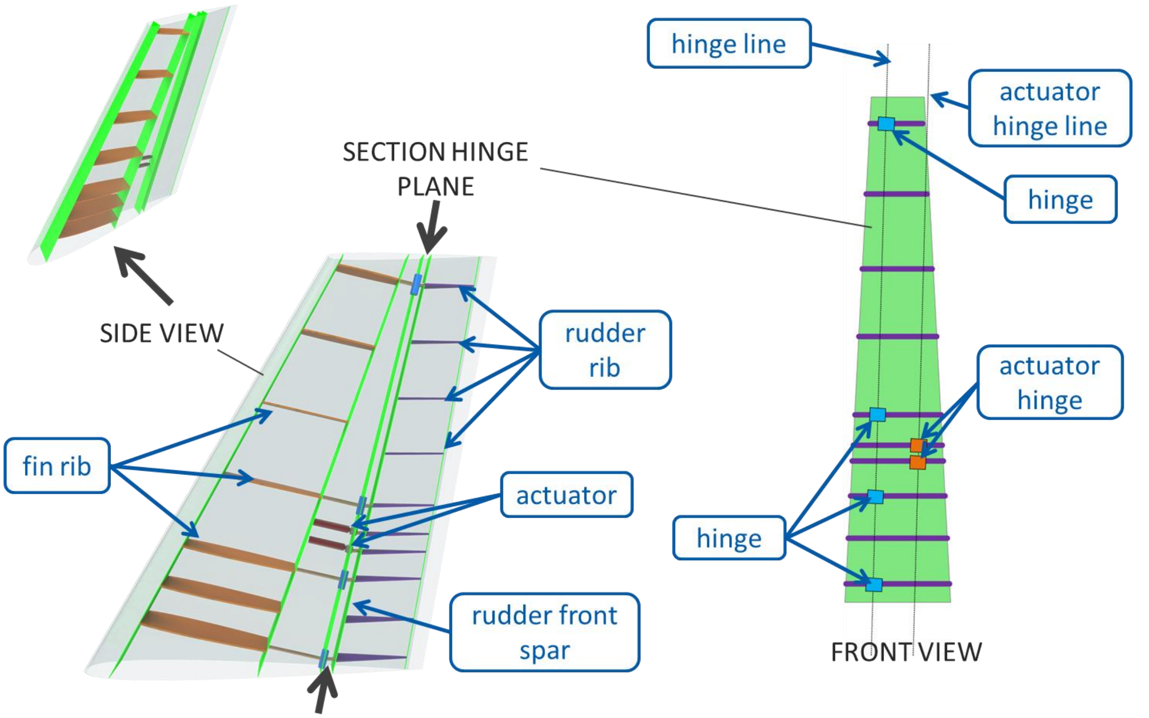


Figure 1. Location of hinge-system in the fin-rudder interface [1].*Isometric view of the vertical tail structure schematic(left). Side view of the vertical tail, with details of the fin-rudder interface (center). Front view of the hinge plane, where both the rotation and actuator hinges are located (right).*

In order to reduce the design lead time, a Knowledge Based Engineering (KBE) tool called Hinge-System Design and Optimization Tool (HDOT) has been developed. KBE technology was chosen because of its ability to capture and systematically reuse product and process knowledge, which aids in automating repetitive and non-creative design tasks. Furthermore, it has been proven able to support MDO in all phases of design process.. [2]

HDOT can quickly and automatically generate a simplified rudder structure based on user defined specifications, generate a quality mesh and carry out structural analysis, using commercial off the shelf tools, to determine the forces acting on the hinges. These forces are in turn used to size the hinge components such as bolts, nuts, bearings and lugs at different hinge locations. HDOT can be seen as an aggregation of multiple design and analysis tools which, in the company, are normally used by engineers/engineering teams specialized in different areas. For example, the geometry generation is the responsibility of CAD specialists, the mesh generation is carried out by FEM experts, and the stress analysis of bolts, nuts, lugs, sleeves etc. is carried out by other system specialists.

HDOT, in addition to integrating multiple disciplinary tools, has the ability to carry out a quasi-exhaustive search of all possible hinge components that can be used at each hinge-location. [3] When used within an optimization framework, for example, to identify the position of the hinge lines leading to a minimum cost or weight of the rudder-fin interface, HDOT eliminates all the infeasible designs and carries out the analysis of all feasible designs to come up with a combination of hinge components that results in least cost or least weight at each iteration. This information is then fed back to the optimizer, which can trigger another HDOT execution with a new set of design variables, until convergence.

Despite HDOT's ability to reduce design lead time and facilitate rapid proposal generation [3] , the embedded quasi-exhaustive search capability brings some limitations:

1. The **control** on the optimization process is limited as the optimization algorithms treat HDOT as a black box. The black box system only allows the optimizer to study the design variables and their corresponding objective function evaluation without providing any understanding of the complex coupling variables necessary for the evaluation of the objective function. In other words, the optimizer cannot control or exploit directly the couplings between the various hinge components sizing tools, which are internally computed and managed by HDOT.
2. It is not possible to perform **multi-objective** design optimization studies, as the quasi-exhaustive search method selects components based on a unique objective function, thereby eliminating the possibility of generating cost-weight Pareto fronts, for example.
3. Quasi-exhaustive search methods are not easily **scalable** and become unaffordable when the number of design variables increases. Moreover, the individual disciplinary design and analysis tools inside HDOT are tightly coupled to the quasi-exhaustive search methods. As a result, replacing the existing disciplinary tools with tools of different fidelity levels is not a trivial effort.

To overcome the above mentioned difficulties, it was decided to eliminate the quasi-exhaustive search approach and expose the individual analysis components of HDOT to the direct control of an optimizer. To this purpose, the monolithic HDOT application was re-architected as a set of independent analysis and sizing modules. A commercial Process Integration and Design Optimization (PIDO) tool, Optimus[§] in this case, was selected to handle the couplings and data exchanges between the modules in a so-called simulation workflow (which can be structured by an MDO architecture). At this point, several questions needed to be answered:

1. What is the most convenient architecture to solve such an MDO problem? For example, is a monolithic MDO architecture more convenient than a multilevel one? Is it convenient to use an architecture like MDF that guarantees design consistency at each optimization cycle, or one that exploits the inconsistencies to speed up convergence like IDF?
2. What design parameters should be used as design variables?
3. What would be the most convenient optimization algorithm to handle, for example, a mix of continuous and integer variables?

[§] Optimus R10.18v1 provided by Noesis Solutions N.V.

Answering these questions is a non-trivial exercise, especially in the context of an industrial design problem where discipline specialists do not always have the required in-depth understanding of MDO architectures and optimization algorithms. On the practical side, a complex MDO problem, as the one specifically addressed in this paper, requires also significant implementation efforts in the PIDO tool of choice. The translation of such a problem into a correct representation of an MDO architecture in a simulation workflow requires a lot of manual work and painstaking debugging, which hamper and, in practice, prevent the possibility to test different MDO architectures and optimization setups in general.

To this purpose, a newly developed MDO support methodology, called InFoRMA, was deployed to exploit the potential of combining KBE and MDO in an industrial setting. The InFoRMA methodology and its implementation into a software tool, was developed at TU Delft within the ITEA2 Project IDEALISM[**]. It is a system, based on Semantic Web Technology[††], that (1) enables designers to specify the multidisciplinary problem at hand, (2) advises them on the most convenient MDO architecture to use, and, (3) automatically generates the complete implementation of the selected MDO architecture for the problem at hand as an executable simulation workflow inside a targeted PIDO system (Optimus in this case) [4].

The scope of this paper is to discuss and demonstrate, by means of the aforementioned industrial design case, the potential of combining KBE and Semantic Web technology to support MDO. In particular, it will discuss how the following challenges can be addressed:

1. Expose the correct level of detail of a KBE application to an optimizer, whilst maintaining consistency to exploit the potential of properly formulated MDO architectures.
2. Automatic integration of KBE applications in an MDO architecture optimization workflow using a PIDO system.
3. Allow for the optimization of the hinge assembly for multiple objectives
4. Scalability of the optimization, to include more materials and design variables
5. Smaller tolerances in optimization results

A brief overview of InFoRMA and the approach to support KBE-enabled MDO by means of semantic web technology is provided in section II. In section III, the application of the proposed methodology to a hinge optimization problem is discussed. Section IV provides the results and conclusions.

II. Enhancing MDO support with KBE through Semantic Web technology

The methodology proposed in this research work is generally applicable to industrial problems where MDO can be applied. Furthermore, this methodology aims to aid rapid MDO problem formulation and implementation for large industrial design problems so as to generate ready-to-use simulation workflows. This is automated by eliminating the lengthy and repetitive manual tasks. This methodology combines the MDO support of the InFoRMA methodology with the strengths of KBE to achieve a means to address the challenges listed in Section I.

InFoRMA provides support for the application of MDO to both experienced and inexperienced MDO users, by means of the following three main functionalities:

1. **Advise:**

   Assist the user with the selection of an MDO architecture suited to the specification of the optimization problem (i.e. disciplines, variables, objective and constraints) and according to user specified additional selection criteria (e.g. consistency of intermediate solutions or need for parallel computations).

[**] http://www.idealism.eu/

[††] World Wide Web Consortium (W3C) - Semantic Web; https://www.w3.org/standards/semanticweb/

2. **Formalize:**
   Automatically formalize a user's problem into a selected MDO architecture, according to the formal definition of the selected architecture, which is stored in a knowledge base. The formalization is supported for eight commonly used MDO architectures: AAO, SAND, IDF, MDF (including two iterations schemes), CO, ECO, CSSO and BLISS. The formal definition provides a neutral workflow representation that can be used both for visualization purposes, by means of automatically generated XDSM [5], and to support their automatic integration in a PIDO system.

3. **Integrate:**
   Automatically integrate the formalized MDO problem into an executable simulation workflow inside a PIDO system, while taking care of all software intensive operations. The integration functionality also allows for the automatic coupling of disciplinary analysis (defined in separated simulation workflows) into the simulation workflow representing the selected MDO architecture.

Semantic Web technologies have been used as the backbone of InFoRMA's methodology and the implementation of the prototype system. These technologies address the meaning (semantics) of data, instead of the basic structuring of the data. This allows computers to access, understand and reason on structured information collections. A structured information collection is called an ontology, which is a formal representation of domain knowledge, based on a set of concepts. Ontologies provide a formal vocabulary that can be used to model types of objects or concepts, their properties and the relationships between them. Using an ontology, knowledge within a certain domain can be modeled in a human-readable format that is also suited for machine reasoning. The backbone of InFoRMA consists of ontologies, rules and reasoning functionalities (a combination of SPARQL database queries and RDF/ OWL light reasoning [6]) specific for both MDO architectures and simulation workflows. A complete description of the InFoRMA methodology, including the applied and developed components of Semantic Web technologies, are presented in the dissertation by Hoogreef. [4] The set of InFoRMA functionalities is graphically illustrated in Figure 2 for the case considered in this paper. Details on this use case are provided in section III.

While InFoRMA can take care of supporting the integration of a generic MDO system, KBE technology can be used to support the development of specific design automation applications, such as, in this case, the development of modeling and analysis tool for hinges. KBE systems like ParaPy[‡‡], the one adopted in this study, help engineers in building advanced rule-based parametric models of complex products, by automating all geometry generation and manipulation activities, while taking care of complex software intensive tasks such as runtime caching, dependency tracking and lazy evaluation [7]. This enables the generation of so called generative models (i.e. models that can automatically generate themselves based on a set of provided input data), that can be easily integrated inside an MDO framework. The optimizer can feed these models with a set of design variables. The knowledge (rules) formalized in these models will guarantee the automatic generation of consistent designs with relative analysis data.

In other words, KBE tools help formalize domain specific knowledge and allow the automation of design process to support MDO. In particular, the generative modeling approach enabled by KBE allows exposing only those product model parameters that must be managed by the optimizer. The dependency tracking and lazy evaluation mechanism cascade the changes made to those exposed model parameters by the optimizer down to the various parts and modules of the KBE application.

[‡‡] https://www.parapy.nl/

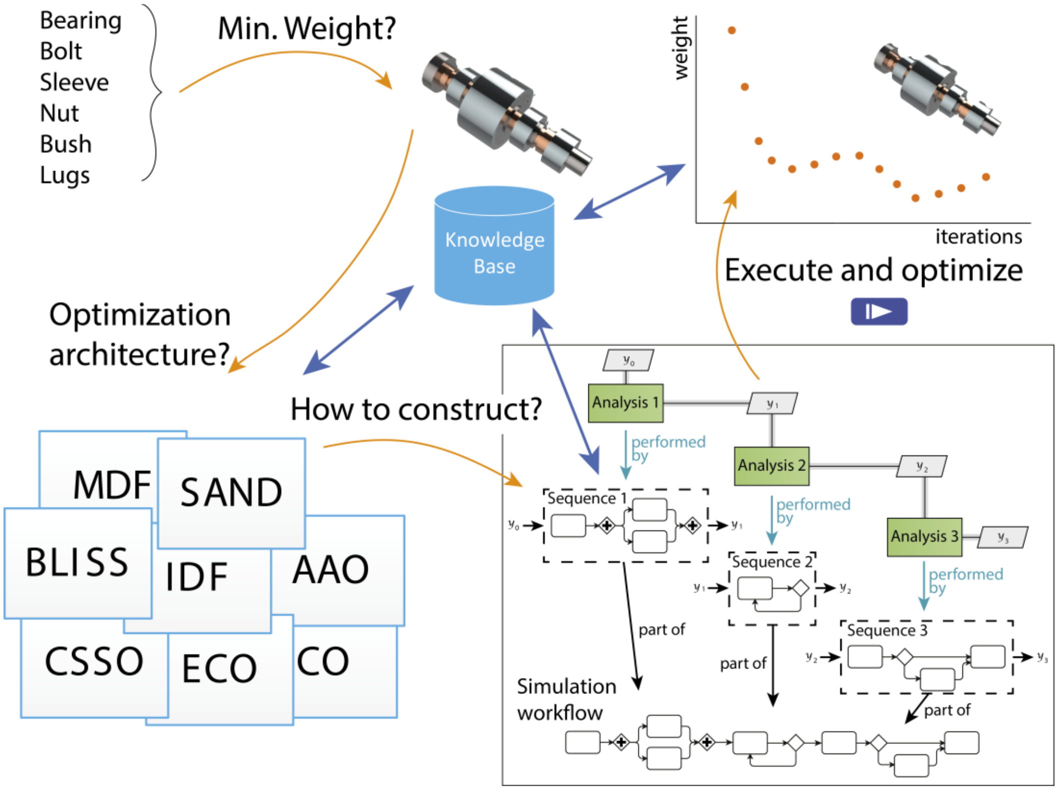


Figure 2. Illustration of the three functionalities of InFoRMA (advise, formalize, integrate) for the case study of the hinge assembly optimization. *Top left illustrates the optimization problem to be solved, in this case consisting of the components of the hinge assembly. The optimizaton problem, for a given objective (e.g. minimizing the weight of the hinge assembly), requires the selection of an appropriate MDO architecture, suited to the problem at hand (bottom left). The problem must be correctly formalized according the selected MDO architecture, such that it is represented by a neutral, formal specification for a simulation workflow (bottom right). The simulation workflow can then be translated according to the format of the PIDO system of choice (in this case Optimus), such that the actual optimization problem can be solved (top right). All information in structured and stored in a central knowledge base, structred by ontologies and applying other Semantic Web technologies, such as reasoning. The knowledge base provides a single source of truth for the optimization problem formulation, formalization (according to the definitions of MDO architectures) and the integration of the optimization problem.*

The selection of the parameters to expose to the direct control of the optimizer (or designer) and those to keep “hidden” inside the KBE application, controlled by the rules coded in the application itself, strongly depends on the design case at hand. As discussed in the introduction of this paper, one can decide to expose a minimum amount of parameters and use the KBE application as a large black box, or expose most of the parameters that couple the various KBE application modules, thus breaking the large black box into multiple simple KBE tools. In the first case one can use the KBE application as a single design competence within a monodisciplinary optimization system; in the second case, one can use the various modules as a set of disciplinary tools inside an MDO system. In the design case addressed in this paper, the KBE tool HDOT, originally developed to perform (black box-wise) the sizing of a complete hinge system, was split into a set of separate modules (e.g. to select the bearing, to size the bolt, the bush, the lug, etc.). The inputs and outputs of each module were then used to create micro analysis workflows using a PIDO tool (e.g. simple sequences composed of input, tool, and output block). Finally, all the disciplinary analysis workflows were combined using InFoRMA into an MDO system. A schematic is shown in Figure 3 and the interconnections between semantic web technologies, KBE and InFoRMA are detailed.

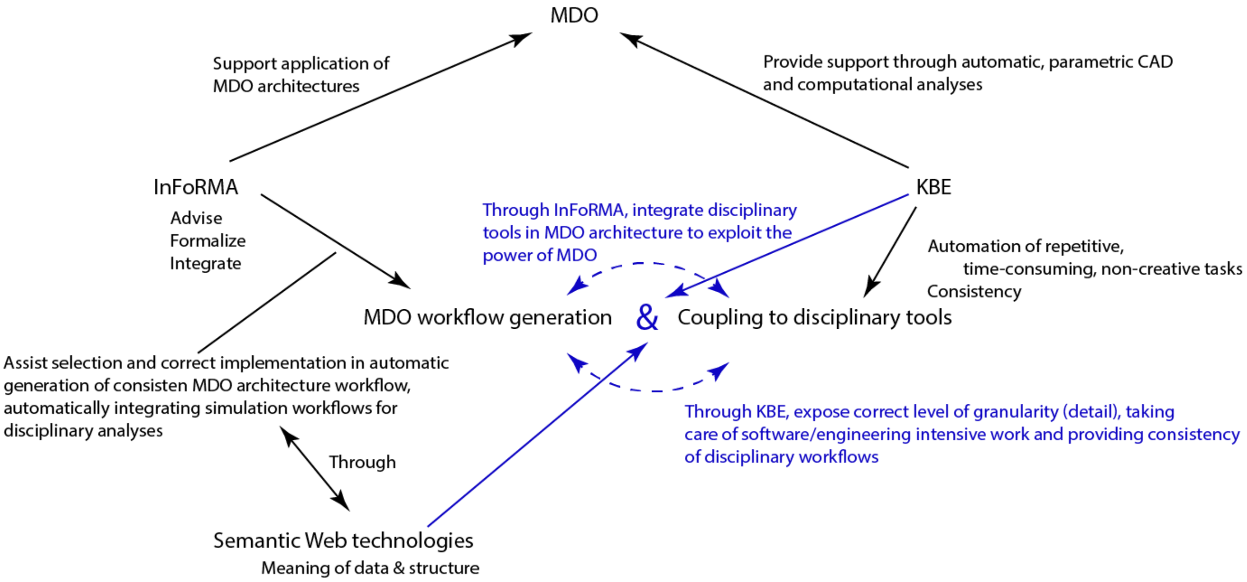


Figure 3. Methodology for rapid problem formulation and formalisation using KBE tools and InFoRMA.

Based on the capabilities of InFoRMA, KBE tools and the semantic web technology, a methodology is proposed which provides the steps that must be used to successfully combine KBE and semantic web technologies. The proposed methodology is broken down into six main phases as follows (refer to Figure 3):

1. **Generation of tools:** In this phase, the disciplinary experts develop design and analysis tools that can be used as a part of simulation workflows

2. **Generation of simulation workflows for disciplinary tools:** The tools developed in step 1 cannot be directly used as a part of the overall optimization. A simulation workflow for the execution of each tool must be created. Manual generation of these simulation workflows is laborious and often prone to human errors. To solve these problems, essential elements of the disciplinary tools that are necessary to generate a simulation workflow are recorded in an XML file based on a well-defined XSD schema. These XML files are used by InFoRMA to automatically materialize the simulation workflows, which are then stored in a database in the form of triples. [8]

3. **MDO problem definition using N^2 chart:** In this phase, the user is required to model the problem inside an N^2 chart by providing the order of execution of various disciplinary tools detailed in step 2 and their respective inputs and outputs. All the information needed to complete the N^2 chart forms the basis for automated formalization and integration and is readily available with the discipline experts/designer. This is relatively straightforward, can be created without being burdened by the complexity of optimization algorithms and architecture.

4. **Advice/Selection of MDO architecture:** For this phase, a user without in-depth MDO knowledge can request advice based on the characteristics of the problem under consideration, while a more experienced MDO user can specify the desired MDO architecture.

5. **Formalization of the MDO problem according to MDO architecture:** The formalization phase takes care of automatically translating the problem definition from the N^2 chart to the formal definition of the selected MDO architecture. This is achieved through the semantic data model, which contains the templates and rules to accomplish this task.

6. **Integration of simulation workflow representing MDO architecture:** The final phase of this methodology performs the integration of the formalized neutral definition as an executable simulation workflow in a PIDO tool. This requires the translation of the neutral formalization to the specific implementation of a optimization problem definition in the PIDO tool. This can be performed automatically thanks to the applied Semantic Web technologies.

In order to demonstrate the efficacy of the proposed methodology of combing Semantic Web technologies with KBE, InFoRMA is used in the recommendation, formalization and integration of an MDO architecture for the fin-rudder interface design optimization problem. The advised architecture is then integrated into executable simulation workflow inside a PIDO tool with the KBE tool HDOT. To measure the lead time gains, the integration is carried out both manually and using InFoRMA. Furthermore, the results of quasi-exhaustive search are compared with the results of optimization carried out by the PIDO tool to study the improvements in product performance.

III. Combining Semantic Web technologies and KBE: Applied to hinge optimization problem

Using the MDO strategy proposed in the preceding section, two studies were carried out. In the first study, the lead time reduction obtained due to the use of InFoRMA was quantified for hinge optimization problem. In the second study, the validity of InFoRMA's advice and its applicability to the hinge optimization problem was investigated and the results of this investigation were further compared with the results of quasi-exhaustive search of HDOT. The optimization problem used for both case studies i.e. hinge optimization is detailed below.

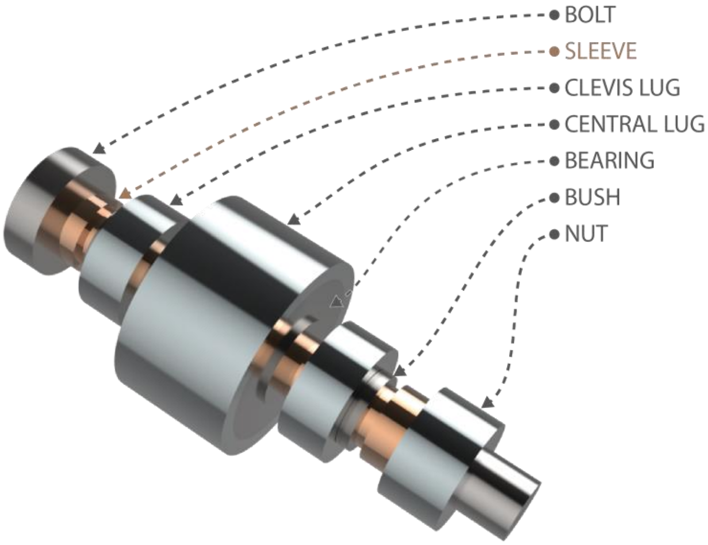


Figure 4. Visualization of hinge assembly generated by HDOT.

Seven components of a sliding hinge namely bolts, nuts, bearings, clevis-lugs, central lug, sleeves and bushes are to be selected and assembled together, a visualization of this is shown Figure 4. Of the seven components, bearings and nuts are standard parts and the rest are machined parts. For each of these components, a material must be chosen and dimensions be determined such that the weight of the hinge assembly is minimum. This is subject to three conditions:

1. The materials of components coming in contact with one another must not react/corrode
2. Each of the components must satisfy the margin of safety requirements for the forces acting on the hinge-assembly
3. The hinge components should fit well with one another to make a consistent design

The problem described involves ten design variables, of which six are integer variables (e.g. part or material ID's) and four are continuous. Because HDOT's internal quasi-exhaustive search immediately eliminates unfeasible combinations of bearing and sleeve materials, a single design variable represents these combinations. All possibilities of the integer variables together (including the bearing-sleeve combination) result in approximately 205e6 possible combinations, for any given set of values for the continuous design variables. The objective of the problem is to minimize the weight of the system, while respecting all margins of safety and geometrical constraints (such that the hinge fits inside the leading edge of the rudder).

A. Problem formulation

The hinge-assembly optimization problem that is considered in this paper is presented below:

minimize:

$$\boldsymbol{W_{hinge} = W_{bearing} + W_{central-lug} + W_{sleeve} + W_{bush} + W_{clevis-lug} + W_{nut} + W_{bolt}}$$

with respect to:

Integer design variables	***bearing-sleeve combination ($\boldsymbol{ID_{bearing-sleeve}}$)***
	nut ID ($\boldsymbol{ID_{nut}}$)
	central-lug material ($\boldsymbol{mat_{CeLug}}$)
	bush material ($\boldsymbol{mat_{bush}}$)
	clevis-lug material ($\boldsymbol{mat_{CLLug}}$)
	bolt material ($\boldsymbol{mat_{bolt}}$)
Continuous design variables	***thickness factor ($\boldsymbol{t_f}$)***
	sleeve outer diameter ($\boldsymbol{d_{o_{sleeve}}}$)
	central-lug outer diameter ($\boldsymbol{d_{o_{CeLug}}}$)
	clevis-lug outer diameter ($\boldsymbol{d_{o_{CLLug}}}$)

subject to:

$$\boldsymbol{MS_{bearing} \geq 0}$$
$$\boldsymbol{MS_{central-lug} \geq 0}$$
$$\boldsymbol{MS_{sleeve} \geq 0}$$
$$\boldsymbol{MS_{bush} \geq 0}$$
$$\boldsymbol{MS_{clevis-lug} \geq 0}$$
$$\boldsymbol{MS_{nut} \geq 0}$$
$$\boldsymbol{MS_{bolt} \geq 0}$$
$$\boldsymbol{geometry\ constraints \geq 0}$$

The problem modeling for InFoRMA requires a definition of an N^2 chart for all disciplines, including a specification of the inputs and outputs of each of these disciplines. In this case, the disciplines that are considered are the separate components of the hinge-assembly, following the logic of the traditional hinge design process. However, the order of these disciplines is not yet determined and it may have a significant influence on the actual optimization time.

In order to bring together all the disciplinary tools, engineers/engineering teams from different disciplines need to agree on the order of execution of the various disciplinary tools. For a large industrial problem such as the fin-rudder interface design optimization, the determination of the execution order can often become difficult. To aid this decision, two main parameters are used, the execution time of every disciplinary workflow and the number of feedbacks a given arrangement of disciplines generates. For the fin-rudder interface design problem, the level of fidelity of tools used resulted in most tools having comparable execution time and the nature of the considered problem allowed the arrangement of disciplines such that feedbacks can be eliminated (this is illustrated in Figure 5 where the original HDOT process and the rearranged process for optimization are shown). This aided the easy determination of the order of execution of the disciplines. Such a process was not used earlier in the industrial context because:

1. engineers followed a legacy design process that guarantee certification,
2. expert MDO knowledge to formulate an MDO problem was missing
3. long preparation time was needed to formalize the problem.

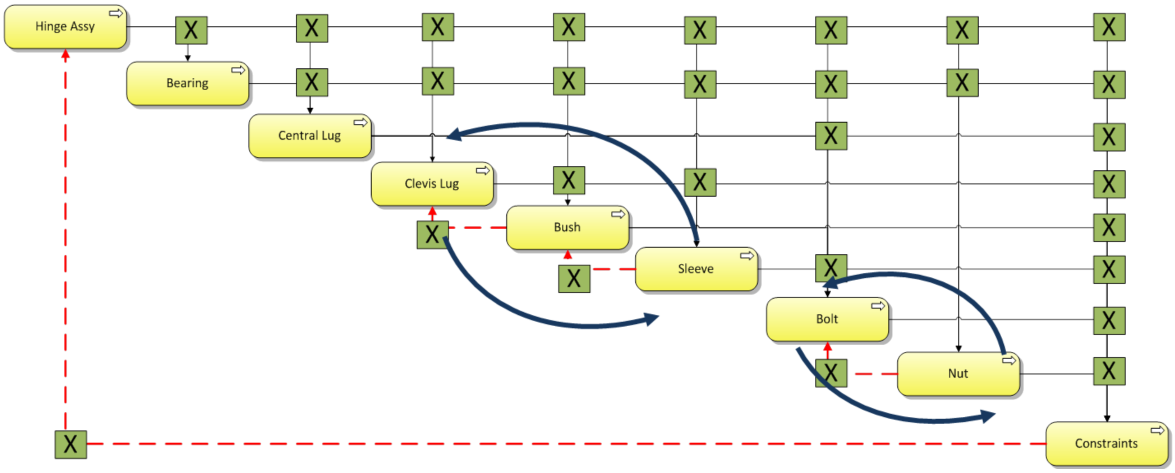


Figure 5a. Original organization of components of the hinge-assembly.

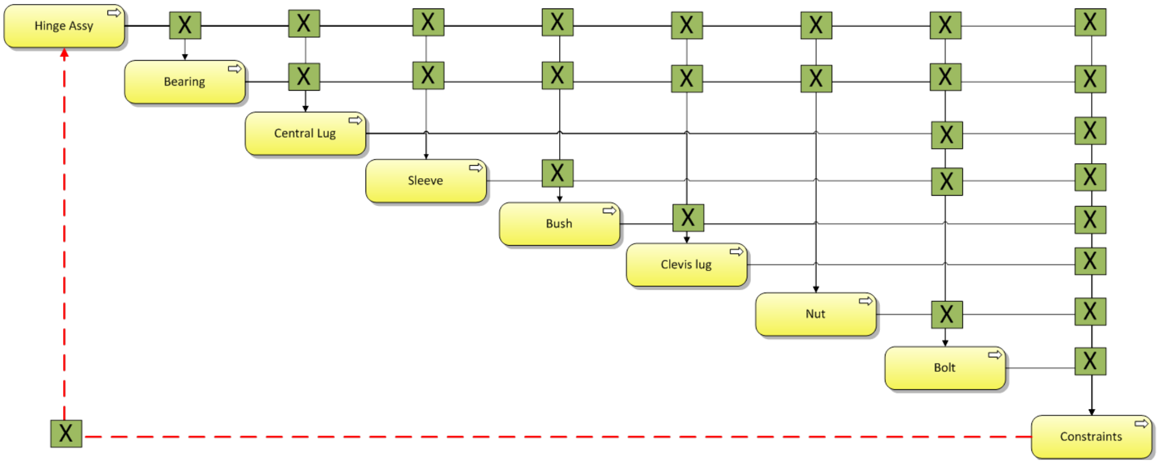
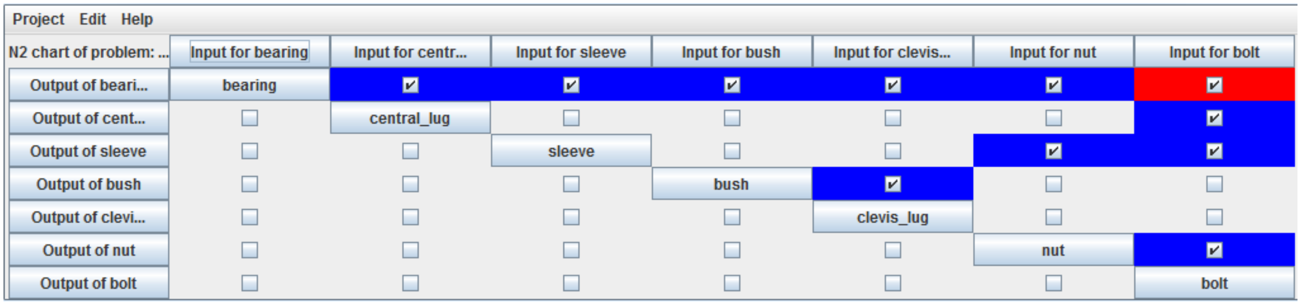


Figure 5b. Rearranged organization of components, resulting in a process without feedback loops.

Once the order of disciplines is decided, the disciplinary experts must determine the inputs and outputs required for each of the disciplinary tools. Based on these inputs and outputs, the design variables are chosen. The inputs, outputs and the order of execution of disciplines are then filled into a N^2 chart using the graphical user interface provided by InFoRMA (shown in Figure 6). In case the disciplinary variable names are not the same as the variable names used in the N^2 chart, InFoRMA allows engineers to map the disciplinary variable names with the names used in the N^2 chart. Such mapping is used by InFoRMA to correctly integrate the various disciplines during the integration phase.



N2 chart of problem: ...	Input for bearing	Input for centr...	Input for sleeve	Input for bush	Input for clevis...	Input for nut	Input for bolt
Output of beari...	bearing	☑	☑	☑	☑	☑	☑
Output of cent...	☐	central_lug	☐	☐	☐	☐	☑
Output of sleeve	☐	☐	sleeve	☐	☐	☑	☑
Output of bush	☐	☐	☐	bush	☑	☐	☐
Output of clevi...	☐	☐	☐	☐	clevis_lug	☐	☐
Output of nut	☐	☐	☐	☐	☐	nut	☑
Output of bolt	☐	☐	☐	☐	☐	☐	bolt

Figure 6. Graphical user interface for N^2 chart in InFoRMA.

B. Architecture advice

For the hinge-assembly optimization problem, the advice provided by InFoRMA yielded the MDF architecture, based on the desire to have a consistent design at every iteration (a feature guaranteed by the coordination loop inside the MDF architecture). Additionally, the no-feedback process that resulted from the re-organization of the optimization problem yields a problem statement that is mathematically equivalent to the problem statement of MDF:

$$\begin{array}{ll} \text{minimize} & f_0(\mathbf{x},\mathbf{y}(\mathbf{x},\mathbf{y})) \\ \text{with respect to} & \mathbf{x} \\ \text{subject to} & \mathbf{c}_0(\mathbf{x},\mathbf{y}(\mathbf{x},\mathbf{y})) \geq 0 \\ & \mathbf{c}_i\ (\mathbf{x}_0,\mathbf{x}_i,\mathbf{y}\ (\mathbf{x}_0,\mathbf{x}_i,\mathbf{y}_{j\neq i}))\ \geq 0 \quad \boldsymbol{i} = 1,2,...,N \end{array}$$

C. Problem formalization

Through the ontologies describing the formal model of MDO architectures and simulation workflows, an automatic formalization of the hinge-assembly optimization problem can be made with just the input provided in the N^2 diagram and the selection of a particular MDO architecture (MDF, in this case). The ontologies provide a template description and structure of the MDF architecture, including all required connections between the various components of the architecture. The N^2 problem model describes the actual variables existing in the problem and the connections that must be made between the physical components of the hinge-assembly. This formalization of the problem contains the relevant information for a translation to a simulation workflow (which can be translated to a PIDO system), as well as visualized in an XDSM for easy inspection by the user. The XDSM for this particular problem is shown in Figure 7. In this figure, it should be noted that although there is an MDA coordinator present along the diagonal, the lack of feedback in the N^2 chart results in a system that is consistent after the first run of the sequence of disciplinary components. Hence, there is no feedback to and no output from the MDA coordinator.

.

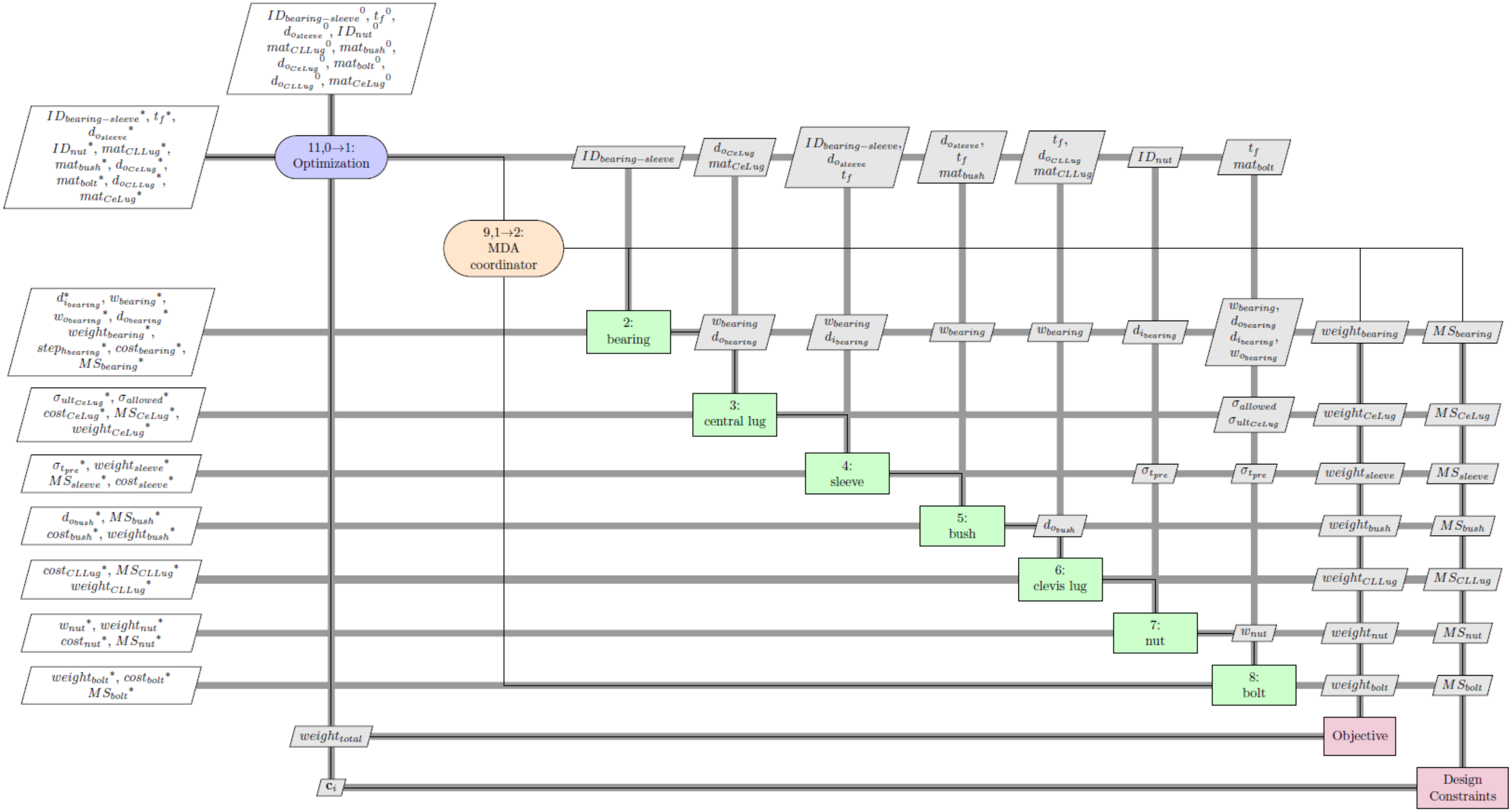


Figure 7. XDSM representation for the MDF architecture of the hinge-assembly optimization problem generated by InFoRMA

The sequential nature of this no-feedback problem means that the Gauss-Seidel iteration schema is ideal for this particular application of the MDF architecture. However, in case there is a desire to perform the disciplinary computations (of each of the hinge's components) in parallel, the Jacobi iteration scheme could be used. In that particular case, there would be feedback to the MDA coordinator and the system would not be consistent at the first run, because the feedforward connections would also be broken to allow for parallel computations. However, this scheme converges more slowly and for this problem, the single-run consistency due to the lack of feedback outweighs the benefits that would be obtained from a parallel execution inside a Jacobi convergence loop

D. Simulation workflow integration

The formalization yields a neutral representation of the optimization problem according to the MDF architecture. However, in order to generate a simulation workflow integrated in the PIDO system Optimus, additional information is required. For the formalization, the specification of the analysis software performing the disciplinary computations was not yet required. However, an executable workflow requires a specification of these analyses, in addition to the specification of the optimization algorithm (in this case a genetic algorithm to cope with the combination of integer and continuous variables), initial values for design variables and bounds of the design variables. Also, this step requires a definition of the objective function and the constraints for the margins of safety (equations relating values to larger/equal zero), both can be defined through an InFoRMA user interface. The geometric constraints are handled internally by the KBE application.

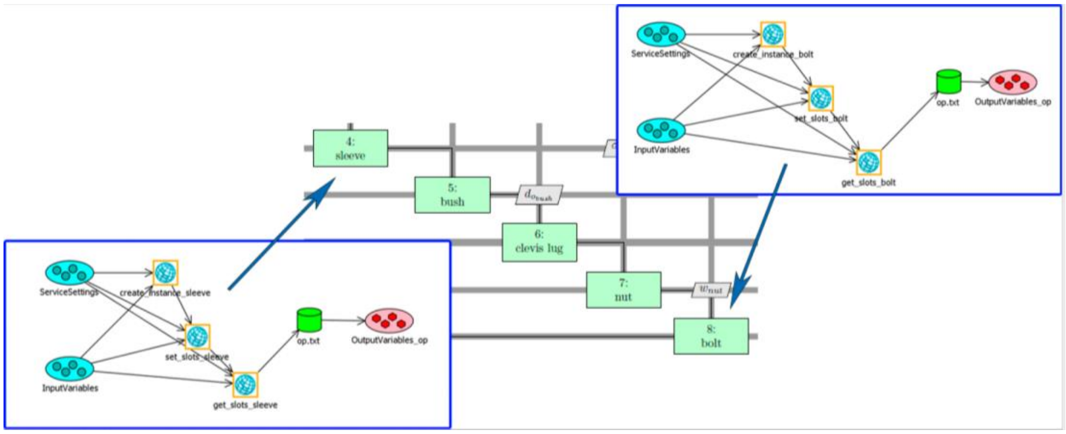


Figure 8. Snapshot of the XDSM for the hinge optimization problem with illustrations of disciplinary workflows for the sleeve and bolt. [5]

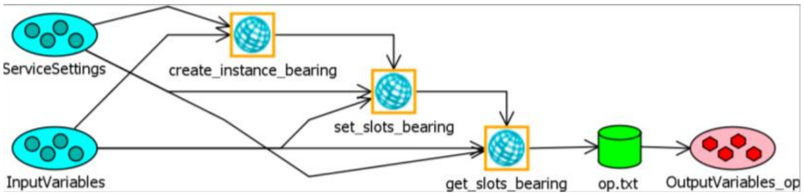


Figure 9. Workflow for bearing component in HDOT consisting of three web-requests to create an instance of the bearing class in the KBE application, set its inputs and retrieve outputs (computed slots). [5]

At this integration step, KBE is directly combined with Semantic Web technology. The meaning of disciplinary analyses is described in both MDO architecture context and simulation workflow context in the respective ontologies. Through a unique-name-assumption, the disciplinary analyses can be related to individual computational components (Figure 8), described according to a formal specification. These disciplinary analyses are in fact components of the product model in the KBE application, which are separately exposed as web services (Figure 9).

The MDO architecture describes how these different web services should be connected to each other to represent the MDF architecture (Figure 10). Hence, a set of loosely coupled web services, integrated in a KBE application handling geometrical constraints and dependency tracking, is automatically, and properly integrated into a simulation workflow representing the MDF architecture (Figure 11).
The product model from the KBE application exposes the different inputs and outputs of each component, which are linked on a simulation workflow level, according to the MDO architecture, to allow the optimization algorithm to assess the effects of the design variables on the different components. Hence, the algorithm can exploit the couplings to optimize the objective.

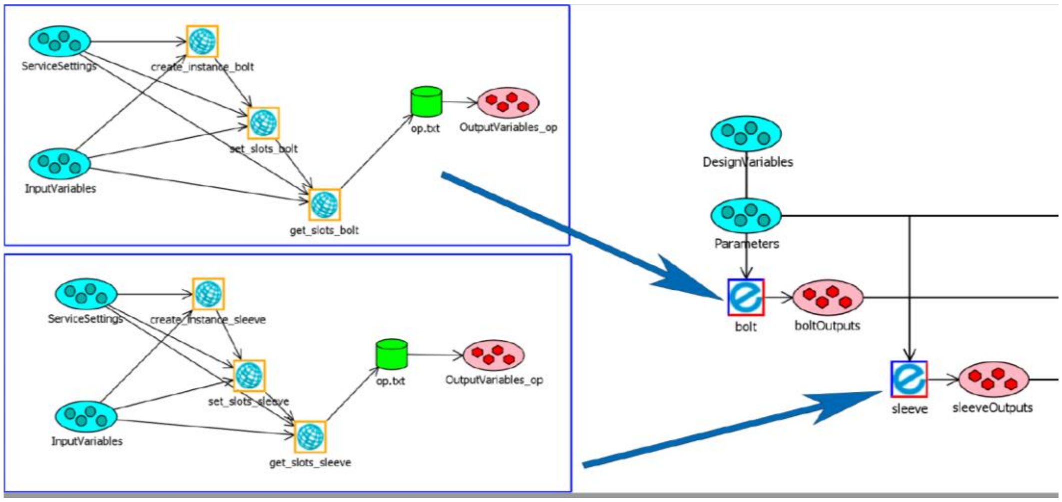


Figure 10. Illustration of the integration of disciplinary workflows for the hinge optimization problem. [5]

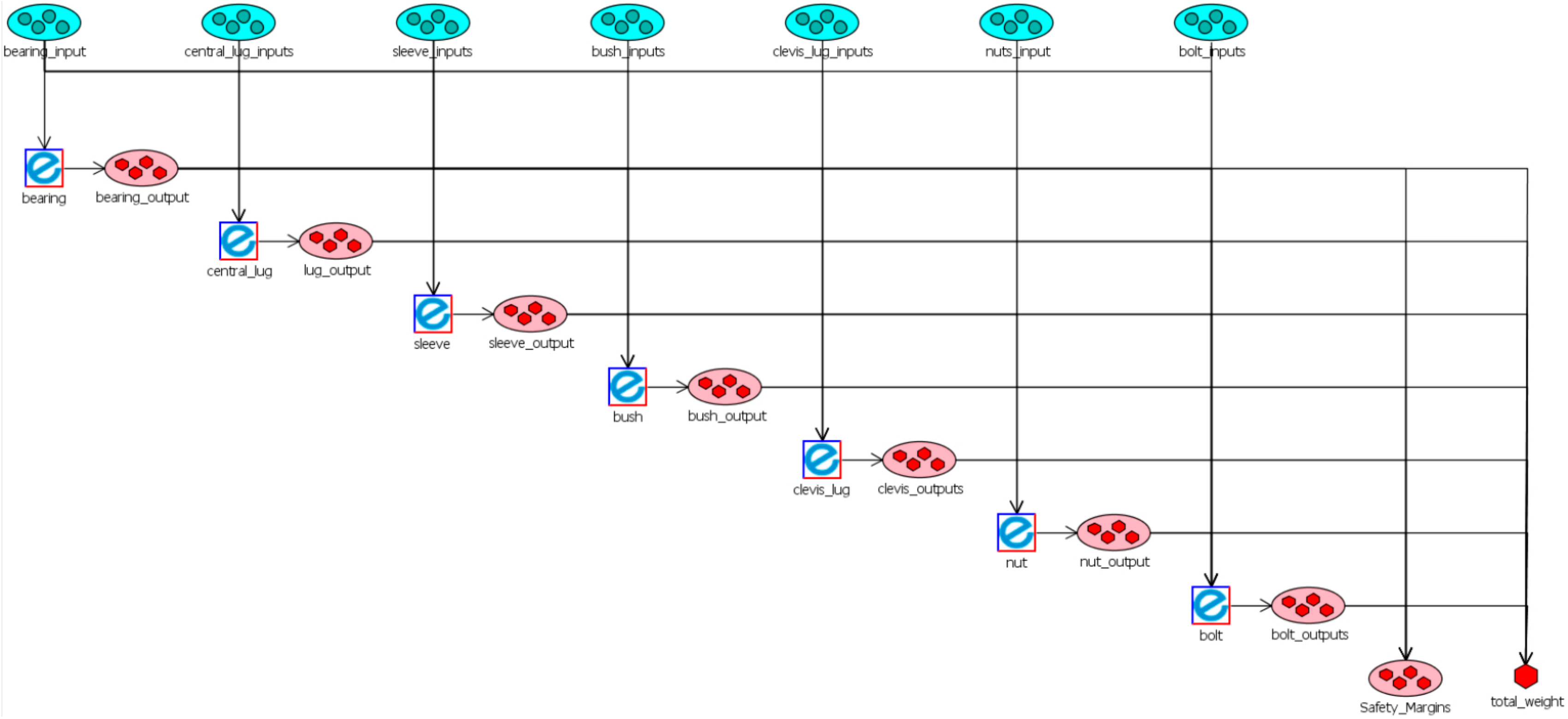


Figure 11. Automatically generated MDF Optimus simulation workflow by InFoRMA for HDOT 5

IV. Results and Conclusions

CASE-STUDY 1: Problem formalization lead time gains

In the first study, the benefits derived from the automation activities shown in Figure 3, i.e. the generation of disciplinary and multi-disciplinary simulation workflow, are quantified. For this, the order of execution of disciplines is decided as shown in Figure 6. Furthermore, the MDO Multidisciplinary Feasible (MDF) architecture is used because of the match of the problem definition to the mathematical definition of MDF and the fact that all components should fit well with one another at every iteration (hence requiring consistency at every iteration). Based on this advice, all the disciplinary and multi-disciplinary simulation workflows are manually created and also automatically instantiated using InFoRMA, the time required for the two are tabulated in Table 1. In this case, a reduction of more than 90% in setup time was achieved. Important to note is that this does not even include the time lost due to failed runs of flawed manual workflow implementations.

Table 1: Time difference in optimization problem formulation- Manual process vs. InFoRMA

Formalization methodology	**Time required to prepare the complete MDF simulation workflow**
Manual MDF formalization	77 hours
InFoRMA MDF formalization	2 hours

The complexity of MDO application is reduced through the problem definition in the N^2 chart and the associated automatic formalization, also resulting in a significant reduction in setup time. Although these results are primarily indicative, since an application by a different engineer might result in a different manual setup time, the automatic formalization and integration is guaranteed to be correct at every application and can be reused for different MDO architectures. Moreover, through the automatic generation and integration of the disciplinary workflows, changes in the problem definition can be easily and automatically reflected in the formalized problem definition.

CASE-STUDY 2: Effectiveness of InFoRMA's simulation workflow

In the second case study, the effectiveness of the MDF simulation workflow generated by InFoRMA is investigated by comparing the results of the optimization of InFoRMA's simulation workflow with the quasi-exhaustive search carried out by HDOT. For this study, the forces acting on the hinge assembly were considered as input parameters for the optimization and the objective function was the minimization of the weight of the hinge assembly. For InFoRMA's generated MDF simulation workflow, a differential evolution algorithm provided by the Optimus (PIDO tool) is used as an optimization algorithm. Table 2 shows the weight of hinge assembly obtained from HDOT's quasi-exhaustive search and InFoRMA's optimization as a fraction of the weight of the HDOT design. The results are normalized due to the confidentiality of the data involved.

Although the run-time of the MDF problem formulation is significantly larger, it is able to find a better design. A large fraction of the overhead is caused by the operating system used for this case, and can hence be reduced.
For both cases, the initial point is the same feasible hinge and the studies are performed using the same machine. Due to the tolerances in the quasi-exhaustive search and biases built-in to the bi-section algorithm that is included, a higher weight design is found by the HDOT application. This quasi exhaustive-search also excludes some combinations of components, which have a slightly heavier component upstream of the design process which leads to the selection of slightly lighter downstream components, thereby resulting in the loss few lighter hinge assemblies due to the fixed design process. The MDO application with MDF allows exploiting the couplings between the different components.

Table 2: Comparison of results of hinge assembly optimization for a given material database and forces acting on the hinge assembly

Components	$\mathbf{Weight_{HDOT}}/\mathbf{Weight_{HDOT}}$	$\mathbf{Weight_{MDF}}/\mathbf{Weight_{HDOT}}$
Bearing weight	1.00	1.00
Lug weight	1.00	0.23
Sleeve weight	1.00	0.74
Bush weight	1.00	0.80
Clevis weight	1.00	0.25
Nut weight	1.00	0.35
Bolt weight	1.00	0.63
Hinge assembly weight	1.00	0.58
Run time	2:30 hrs:min	9:16 hrs:min

Conclusions

For the hinge-assembly optimization problem, KBE tools coupled with the MDO architecture recommended by InFoRMA have found better design solutions as compared to the designs recommended by the quasi-exhaustive search of HDOT. The quasi-exhaustive search itself comes very close to the optimum but is limited by its convergence tolerances. The use of MDO architectures, in combination with the exposition of the correct level of granularity from the KBE model allowed for a reduction of the tolerances introduced by the quasi-exhaustive search of HDOT.

Up to 90% reduction in setup time was achieved for hinge assembly problem. InFoRMA shows reduction in lead time by automating manual, repetitive and MDO knowledge intensive tasks thereby effectively reducing the human errors. This lead-time reduction allows engineers to try out different MDO architectures for the same problem and based on the outcome of the architecture trade-off study, most suitable architecture can then be selected for use in a higher level optimization problem.

Since the KBE application can be automatically integrated through disciplinary workflows in the MDO architecture, InFoRMA can automatically substitute one micro analysis workflow with another depending on the fidelity and accuracy of the generative model desired, thereby, improving the scalability of KBE applications. When a different optimization objective is defined, the use of the formal problem definition ensures that the problem formalization and integration is automatically corrected to represent this change. This provides engineers an opportunity to study the impact of their design decisions on different design objectives and perform multi-objective optimization.

The studies carried out in this research form the preliminary demonstration of the potential of combining KBE with semantic web technologies. While semantic web technologies play a crucial role in integrating, formalizing and advising on industrial MDO problems (demonstrated using InFoRMA), KBE enables the formalization and automation of complex engineering tasks involved in the design process (demonstrated using HDOT). Without semantic web technologies, the formalization and implementation of MDO problems becomes a rigorous and challenging problem and without KBE, application of MDO in industry would be limited even with the availability of InFoRMA like tools. This effectively creates a symbiotic relationship between KBE and semantic web technologies which can be exploited to efficiently solve complex industrial MDO problems.

Acknowledgments

The research leading to these results was performed within the European ITEA2 project IDEaliSM (#13040) as part of the Eureka cluster programme. The authors would like to express their gratitude to the IDEaliSM partners for their support and contributions.

References

1 Raju Kulkarni, A. *Development of Knowledge Based Engineering tool to support fin-rudder interface design and optimization;* Master thesis; Master of Science thesis-Delft University of Technology: Delft, 2015.

2 La Rocca, G., "Knowledge based engineering: Between AI and CAD. Review of a language based technology to support engineering design.," *Advanced engineering informatics,* Vol. 26, No. 2, 2012, pp. 159-179.

3 Raju Kulkarni, A., van Dijk, R. E. C., van den Berg, T., and La Rocca, G. "A Knowledge based engineering tool to support front-loading and multi-disciplinary design optimization of the fin-rudder interface [To Be Published]," *CEAS*, Romania, 2017.

4 Hoogreef, M. F. M. *Advise, Formalize and Integrate MDO Architectures - A Methodology and Implementation;* PhD thesis; To be published: Delft, 2017.

5 Lambe, A. B. and Martins, J. R. R. A., "Extensions to the design structure matrix for the description of multidisciplinary design, analysis, and optimization processes," *Structural and Multidisciplinary Optimization*, Vol. 2012, No. 46, 2012, pp. 273-284.

6 Hoogreef, M. F. M. and La Rocca, G. "An MDO advisory system supported by knowledge-based technologies," *16th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference*, Dallas, Texas, 2015.

7 La Rocca, G. *Knowledge Based Engineering techniques to support aircraft design and optimization [PhD Thesis]*, Delft, 2011.

8 Hoogreef, M. F. M., D'Ippolito, R., Augustinus, R., and La Rocca, G. "A multidisciplinary design optimization advisory system for aircraft design," *5th CEAS Air and Space Conference "Challenges in European Aerospace"*, Delft, 2015.