

Decomposition analysis of the multidisciplinary coupling in LED System-in-Package design using a DSM and a specification language

E. C. M. de Borst¹ · L. F. P. Etman¹ · A. W. J. Gielen² ·
A. T. Hofkamp¹ · J. E. Rooda¹

Received: 20 March 2014 / Revised: 16 November 2014 / Accepted: 20 November 2014 / Published online: 26 April 2016
© The Author(s) 2016. This article is published with open access at Springerlink.com

Abstract LED System-in-Package (SiP) aims to reduce manufacturing and material costs of LED lighting products through integration of components into one single package, based on semiconductor technology. This introduces multidisciplinary coupling in the system behavior which requires reconsideration of the existing LED design decomposition practice. This paper presents our method to do a decomposition analysis of the multidisciplinary coupling structure for an industry scale LED SiP design problem. The innovative aspect of our method is the use of a specification language to specify the input-output (binary) relationships between design variables, behavioral responses, objective and constraint responses. A design structure matrix, representing the mutual linkage, is automatically generated from the specification. The rows and columns of the DSM are subsequently re-ordered using partitioning and sequencing algorithms to provide insight in the coupling structure. The method is illustrated by means of a simplified example.

Subsequently, the results for a recently manufactured LED SiP design prototype are presented.

Keywords Multidisciplinary design optimization (MDO) · Design structure matrix (DSM) · Specification language · Partitioning · Sequencing · Complex systems

1 Introduction

Decomposition is commonly used in the design of engineering systems. Well-known are the decompositions along the lines of disciplines (aspect decomposition), components (object decomposition), and tasks (sequential decomposition). The mathematical programming approach to decomposition-based design is referred to as Multidisciplinary Design Optimization (MDO) (Alexandrov 2005). A large number of MDO papers has been published during the last decades, see for instance Cramer et al. (1994), Balling and Sobieszczanski-Sobieski (1996), Sobieszczanski-Sobieski and Haftka (1997), Tosserams et al. (2009), Martins and Lambe (2013), and Allison and Herber (2014) for reviews of selected topics. A substantial research focus has been on the mathematical formulation and organization of the coordination problem assuming a particular decomposition structure.

An appropriate decomposition structure, however, is not always easy to find. In the nineties, Wagner and Papalambros (1993) and Michelena and Papalambros (1997) developed an optimal model-based partitioning approach based on mathematical programming. They departed from a functional dependence table (FDT) description of the design problem (rows are associated with objective/constraint functions; columns with design and state variables). They used graph partitioning techniques to identify decomposition

This paper is a revised and extended version of proceedings paper “Decomposition-based analysis of the multi-physical coupling structure in LED System-in-Package design” presented by the authors at the 10th World Congress on Structural and Multidisciplinary Optimization, May 19-24, 2013, Orlando, FL

✉ L. F. P. Etman
l.f.p.etman@tue.nl

A. W. J. Gielen
sander.gielen@tno.nl

¹ Department of Mechanical Engineering,
Eindhoven University of Technology, PO Box 513,
5600 MB Eindhoven, Netherlands

² TNO Technical Sciences, De Rondom 1,
5612 AP Eindhoven, Netherlands

structure. Rogers (1989, 1996) used a directed instead of undirected representation of the optimal design problem by means of a design structure matrix (the DSM displays the relationships between the elements of a system in a square matrix). His focus was mainly on finding decomposition structure associated with sequencing design tasks. Design variables, behavior (state) variables, constraint, and objective function values are viewed as input and output of these tasks. A genetic algorithm was employed to optimize for the sequence of tasks.

It is likely that a decomposition needs to be revised as the system design progresses. In his introduction section Rogers (1989) noted: “for designs based on novel concepts, like large space platforms, the determination of the subsystems, interactions, and participating disciplines is an important task. Moreover, this task must be repeated as new information becomes available or as the design specifications change”. Alexandrov and Lewis (2004) stated that “in realistic MDO environments, it is often difficult to determine a priori whether a chosen (MDO) problem formulation will produce satisfactory results. Reconfiguration may be required.” They observed that this requires the ability to *specify* the MDO components in sufficient detail. They proposed a linguistic approach to MDO problem description, formulation and solution, which they called reconfigurable multidisciplinary synthesis (REMS). Tossierams et al. (2010) also noted that the use of a specification language provides for easy manipulation of the way a system is partitioned, in particular when the system becomes larger. They developed the language Ψ , which is based on a composition paradigm that starts from the definition of individual components (subproblems) that are subsequently assembled into larger systems.

To model the impact of technology or engineering changes, the DSM is frequently adopted. For instance, Clarkson et al. (2004) developed a change prediction method based on the DSM. Chen et al. (2007), Li (2010a, b) developed decomposition analysis techniques for this change prediction DSM method. Suh et al. (2010) presented a delta DSM concept to model the effect of technology changes on the baseline product architecture. Changes may lead to a misalignment between product and organization architecture. Sosa et al. (2004) used a DSM to identify the design interfaces between the components and a second DSM to identify the team interactions developing each of the components. By comparing the two matrices, the misalignment can be analyzed. For a commercial aircraft engine case, they concluded that the likelihood of misalignment is greater across the system decomposition boundaries.

Recently we were asked to analyze the decomposition structure used in a project on the design of a prototype of a new LED lighting system. Several companies were involved in the development of the prototype. The design

is based on a novel manufacturing technology to integrate several components of the LED system, such as, amongst others, driver chip, LEDs, sensors, into a single semiconductor solution, referred to as LED System-in-Package (SiP) (Gielen et al. 2011). The novel manufacturing technology caused major difficulties during the design process. The difficulties were mainly attributed to a different and more pronounced multidisciplinary (structural, thermal, electrical, optical) functional coupling, as well as more interaction between the various companies during assembly when compared to the development of existing LED lighting devices. Although a practical solution was found during execution of the project, there is a need to take into account the multidisciplinary coupling in the design decomposition, at an earlier stage of the development process.

This paper presents our method for decomposition analysis of the multidisciplinary coupling structure, and the industry application to the LED SiP prototype. We analyze the multi-disciplinary coupling structure of the LED SiP by means of a DSM. A DSM representation of the coupling structure is frequently adopted in MDO, see for instance Altus et al. (1996), Lu and Martins (2012), and Jung et al. (2013). To model the multidisciplinary coupling inside the LED SiP, we define as elements of the DSM: the design variables, the behavior/state responses, and the objective/constraint responses (which relates to the parameters of the functional dependency table mentioned earlier). Such a DSM in terms of the MDO variables and responses may be viewed as a parameter-DSM (see Browning (2001)). The matrix is similar to the reduced adjacency matrix presented in Allison et al. (2009), but we use the unreduced square DSM format. We refer to our DSM as the *multidisciplinary* DSM, to clearly distinguish from the (parameter-)DSMs commonly presented in the DSM product development literature, see e.g. Eppinger and Browning (2012), and the extended DSM (XDSM) introduced by Lambe and Martins (2012) to describe multidisciplinary design, analysis, and optimization *processes*.

The scale of the LED SiP application makes it rather elaborate and error-prone to ‘enter’ the zeros and ones in the DSM by hand. The LED SiP case problem consists of hundreds of design variables and responses. An innovative aspect of our method is that we use the Ψ language (Tossierams et al. 2010) to specify the functional relations between design variables, behavior/state responses, and objective/constraint responses. Ψ allows a *local* specification of functional relationships in subproblems and a bottom-up assembly of subproblems in subsystems and systems. From the Ψ specification we automatically generate the DSM.

The multidisciplinary DSM generated from the Ψ specification is subsequently partitioned using computational partitioning algorithms (Thebeau 2001; Dhillon et al. 2007)

minimizing a weighted criterion of partition sizes and partition interactions. Also a combination of an exhaustive sequencing approach and the Dulmage-Mendelsohn decomposition algorithm (Dulmage and Mendelsohn 1958) is employed to obtain a matrix with minimal feedback-coupling. The re-ordered matrix serves as a means to analyze the SiP coupling structure, and to compare with the existing design decomposition practice in LED lighting product development.

The paper is organized as follows. In the next section the LED SiP concept and multidisciplinary LED SiP design problem is explained in further detail. Subsequently, in Section 3 the DSM, reduced adjacency matrix, and FDT are explained, as well as the purpose of partitioning and sequencing these matrices. Section 4 presents our proposed method. Section 5 gives an illustrative example to demonstrate the method. In Section 6, the method is applied to the design of the LED SiP prototype. Section 7 offers concluding remarks.

2 LED system-in-package

LED System-in-Package (Gielen et al. 2011) is a novel lighting concept that aims to reduce the costs of the next generation LED lighting products. The LED SiP can integrate LED chips with driver chips, sensors, radio-frequency interfaces, and possibly other components into a single semiconductor solution, see Fig. 1. The integrated and miniaturized package reduces manufacturing and material costs. Due to integration in the system, the coupling in the product design becomes more prominent (de Borst et al. 2012). Unlike traditional devices built on printed-circuit-boards, the interaction between the different functional components goes beyond the electrical domain: heat generation, electromagnetic fields, optical interference, and mechanical loads between the functional components must be taken into account to obtain a LED SiP with a good performance and lifetime.

During the LED SiP product design, mechanical (structural reliability), thermal, electrical and optical aspects have to be taken into account. In particular thermal management

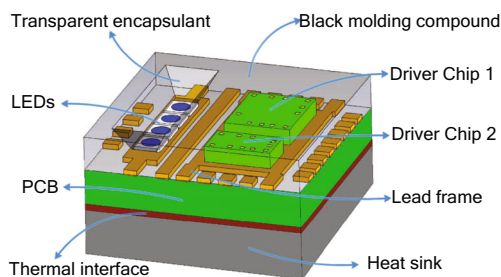


Fig. 1 LED SiP design concept

plays a key role, since the performance and lifetime of components is heavily affected by high temperatures within the package. For example, the actual light output of high brightness LEDs depends on the junction temperature in the LED. As the temperature in the LED rises, the light output decreases (Hechfellner and Landau 2009). Also the quality of the light deteriorates under increasing temperatures, since LEDs tend to shift wavelength at higher temperatures, resulting in off-color light output (Hong and Narendran 2004). On a longer time horizon, due to high temperatures and high electrical currents, driver and lenses tend to degrade resulting in lumen depreciation and color shift (Tarashioon et al. 2011; Narendran and Gu 2005; Koh et al. 2011). The LED SiP can also fail catastrophically, when one of the components in the LED SiP instantly fails. Such catastrophic failures may be electrically or thermo-mechanically induced (Chang et al. 2012).

Typical design decisions for the LED SiP are associated with, amongst others, the type, dimensions and mutual placement of the various components that make up the SiP, the routing of interconnects, and the packaging. The main design objectives are: performance, lifetime, and costs. The performance is described by the luminous efficacy, i.e. the amount of light (lm) per amount of electrical energy (W), and the quality of the light output. The lifetime of the LED SiP is a function of failure due to lumen depreciation and catastrophic failure of components. The costs relate to material and manufacturing costs.

Several teams of engineers, typically from different companies, are involved during the design of a LED SiP product. Each team has a specialization in one particular discipline or component of the SiP. In product design of existing LED lighting systems, the design of the electrical circuit is usually performed first, followed by the design of the components and the design of the package. Ultimately proper functioning of the full system is analyzed. However, the coupled nature due to the SiP concept complicates prediction of the system behavior and thus the process of decision making. The project team involved with the development of a first LED SiP prototype experienced that the new manufacturing integration concept introduced multidisciplinary coupling that requires a different design decomposition. We carried out a decomposition analysis of LED SiP case using a parameter DSM to represent the multidisciplinary coupling and a specification language to model the various functional relationships.

3 Matrix representation and reordering

In this section the design structure matrix (DSM) concept is summarized, as well as two matrix reordering techniques that can be applied to the DSM. The functional dependence

table (FDT) and the reduced adjacency matrix are two related matrix concepts commonly used in MDO modeling. The three matrices use MDO parameters and functions as matrix elements. Since they are closely related, the FDT and the reduced adjacency matrix are presented here as well.

3.1 Functional dependence table

The *functional dependence table* (FDT), is an $m \times n$ representation matrix, aimed at the decomposition of model-based design optimization problems (Wagner 1993; Michelena and Papalambros 1997). Figure 2 shows an FDT. Each row of the FDT represents a design function, either an objective function or a constraint function. Each column represents a design variable or a state variable. Table entry (i, j) is non-empty if the i^{th} function depends on the j^{th} variable.

3.2 Design structure matrix

The *design structure matrix* (DSM) originates from Steward (1981), who proposed a matrix-based technique to manage the design of complex systems, in particular regarding the interactions between elements of the system. The main purpose of a DSM is to illuminate structure and aid in the design of products, processes and organizations (Browning 2001; Eppinger and Browning 2012).

A DSM is a square $N \times N$ matrix with identical row and column labels, representing the elements of the modeled system. These elements can be, amongst others, system components, parameters, tasks, and activities. An off-diagonal mark represents linkage between two elements. Besides binary marks, also weighted interactions can be presented in a “numerical” DSM, for example to differentiate between strong and weak dependencies.

The DSM may be symmetrical or asymmetrical. In a symmetrical DSM directionality of interactions is not considered. In an asymmetrical DSM, the input-output direction of the linkage is taken into account. In this regard, two conventions exist. In the DSM convention due to Steward, a

	x_1	x_2	x_3	x_4	x_5	x_6
f_1	X			X		
f_2		X			X	
h_1	X	X				
h_2	X	X	X			
h_3				X		X
h_4	X			X		
h_5		X			X	
h_6			X			X

Fig. 2 Functional dependence table

cross at row i , column j means that element j is input to element i . However, the opposite convention, element i is input to element j , is also frequently used (Eppinger and Browning 2012). In this paper we adopt Steward’s convention. Figure 3 shows an example of a DSM, representing the interactions between seven elements, labeled A through G, by means of cross-marks. For instance, reading across row D, shows that element D has inputs from elements B, C, E, and G, represented by the cross marks in row D, and columns B, C, E, and G.

3.3 Reduced adjacency matrix

Allison et al. (2009) use the DSM to model the input-output properties in design problems with multidisciplinary coupled analyses. They use analysis functions a , and design variables x , as DSM elements. Since design variables are independent quantities the corresponding matrix rows are zero. These rows may be omitted without loss of information. Allison et al. (2009) refer to this condensed matrix as the *reduced adjacency matrix*. They use this matrix as basis for simultaneous partitioning and coordination strategy decision making.

Figure 4 shows an example of a reduced adjacency matrix. Similar to Steward’s DSM convention, a non-zero element in the i^{th} row and j^{th} column indicates that element j is input to element i . The matrix is organized such that analysis functions appear before design variables in order to make a visual distinction between them.

3.4 Partitioning

With *partitioning* a matrix we mean identifying blocks of strongly coupled elements in the matrix. Strongly coupled elements should be considered together in the design or optimization. Figure 5 gives an illustration of partitioning a DSM. Note that in the context of time or task based (directed) DSMs, partitioning may be used to transform the DSM into block angular form to identify the blocks that are without coupling (Meier et al. 2007). Partitioning an FDT (which is undirected) aims to reveal patterns that allow

	A	B	C	D	E	F	G
A						X	
B	X		X	X			X
C				X			X
D		X	X		X		X
E	X			X		X	
F	X				X		
G		X	X	X			

Fig. 3 Design structure matrix

	a_1	a_2	a_3	a_4	x_1	x_2	x_3	x_4
a_1	0	1	1	0	1	0	0	0
a_2	0	0	1	0	0	1	0	0
a_3	1	1	0	0	0	0	1	0
a_4	1	0	0	0	1	1	1	1

Fig. 4 Reduced adjacency matrix

decomposition of the optimization problem (Wagner and Papalambros 1993).

In partitioning, there are two competing goals: minimizing the size of partitions and minimizing the number of interactions between partitions. Typically a weighted criterion of partition size and partition interaction is used. Partitioning algorithms may directly operate on the matrix or alternatively use a graph representation.

3.5 Sequencing

Sequencing is the reordering of the DSM's rows and columns such that the new DSM arrangement minimizes some criterion regarding the presence of feedback marks (Fig. 6). An often used criterion is the number of super diagonal marks (in Stewards convention), but many other sequencing objectives have been defined (see e.g. Meier et al. (2007) for a short overview). A system without feedback coupling can be sequentially designed. Feedback coupling implies design iterations in the system design process. Note that if the partitioning of the DSM provides a block angular form, the sequencing can be restricted to the elements within the respective blocks.

4 Method

We denote \mathbf{x} to be the vector of design variables, \mathbf{r} the vector of behavioral responses (intermediate responses), and \mathbf{F} the vector of objective and constraint responses (end responses). We denote one objective/constraint function f by $F = f(\mathbf{x}, \mathbf{r})$. We denote one response function \mathbf{a}_i by $\mathbf{r}_i = \mathbf{a}_i(\mathbf{x}, \mathbf{r}_{i \neq j})$. The inputs of response function \mathbf{a}_i are

	A	B	C	D	E	F	G
A	█					X	
B	X	█	X	X			X
C			█	X			X
D		X	X	█	X		X
E	X			X	█	X	
F	X				X	█	
G		X	X	X			█

	B	C	D	G	A	E	F
B	█	X	X	X	X		
C		█	X	X			
D	X	X	█	X			X
G	X	X	X	█			
A					█	X	
E			X		X	█	X
F					X	X	█

Fig. 5 Partitioning a DSM into two partitions

	A	B	C	D	E	F	G
A	█		X				
B		█	X	X			
C	X		█		X		
D	X			█			X
E					█		
F			X		X	█	X
G		X	X				█

↑ Feedback coupling

	E	A	C	B	D	G	F
E	█						
A		█	X				
C	X	X	█				
B				█	X		
D		X			█	X	
G			X	X		█	
F	X	X			X		█

Fig. 6 Sequencing of a DSM

design variables \mathbf{x} and behavioral response output from other response functions denoted by $\mathbf{r}_{i \neq j}$. Figure 7 illustrates the coupling between design variables \mathbf{x} , responses \mathbf{r} , and objective and constraint responses \mathbf{F} . The response functions represent the relations that have to be considered to model the multidisciplinary coupling of the various components in the system. Our main interest is to identify which input-output relations are present. A binary representation matrix is used, which describes the existence of interactions but which does not quantify the actual relations.

4.1 Representation matrix

The coupling between design variables \mathbf{x} , behavioral responses \mathbf{r} , and objective and constraint responses \mathbf{F} is represented by the multidisciplinary DSM, as shown in Fig. 8. The elements of the DSM are $\mathbf{F}_1, \dots, \mathbf{F}_m, \mathbf{r}_1, \dots, \mathbf{r}_p, \mathbf{x}_1, \dots, \mathbf{x}_n$. Note these elements are the input and output variables of the functional relationships

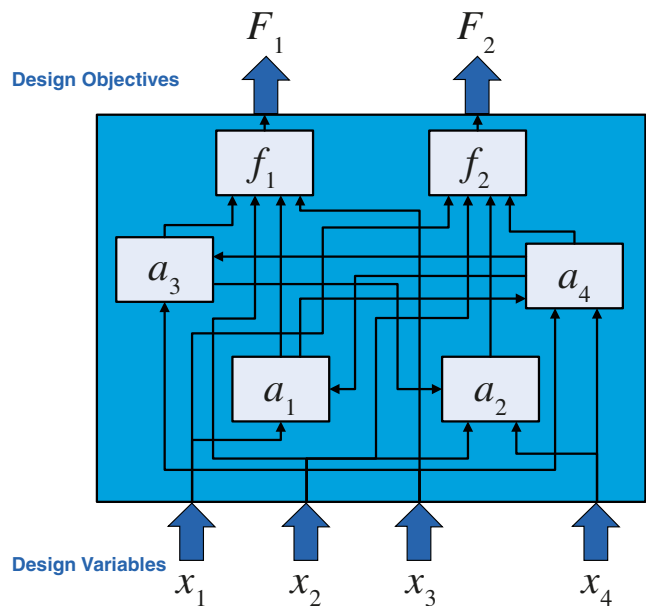


Fig. 7 Illustration of the coupling between design variables, response functions, design objectives

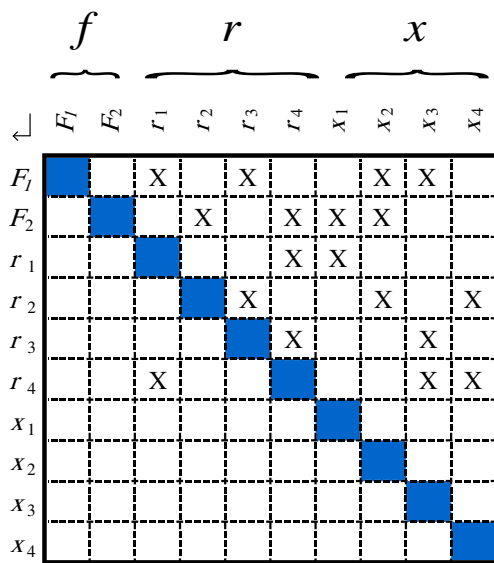


Fig. 8 Multidisciplinary DSM showing the coupling between design variables, behavioral responses, and objective/constraint responses

f and **r**. Also note that we use vector notation for the elements of the DSM to express that an element may be represented by multiple variables that should stay together during the partitioning and sequencing. The rows corresponding to the design variables are all empty (see Section 3.3). Similarly, the columns corresponding to the objective and constraint responses are empty. In contrary to the reduced adjacency matrix of Section 3.3, we keep the zero rows and columns to retain the square matrix, such that DSM partitioning and sequencing methods can be used.

In industrial applications the size of the multidisciplinary DSM may grow to hundreds or thousands of elements. In the DSM literature, the identification of interactions between elements is mainly based on design documentation and interviews with designers. This interaction modeling is not straightforward, and is quite difficult both for larger systems, and in the early design phase, see Tilstra et al. (2010), Dong (2002), and Schmitz et al. (2011). An interesting domain where one builds large DSMs is the analysis of software applications, see for instance Sangal et al. (2005). The DSM can be computer generated from the inputs and outputs to the subroutines and files in the software code. As a result, the DSM can become exceptionally large. The LED SiP application falls into the first category: the relations between design variables **x**, behavioral responses **r**, and objective/constraint responses **F** are to be specified using literature and knowledge available with engineers.

4.2 Specifying interactions

We notice, however, that we do not need to manually fill the DSM, checking entry by entry of the matrix. Instead, we

propose to use a language to specify the variables and their interactions, and subsequently generate the matrix automatically. We have adopted the Ψ specification language (Tosserams et al. 2010) to demonstrate the advantage of such an approach.

The Ψ specification language was originally intended as a linguistic software tool for specification of partitioned problems in decomposition-based design optimization. The language elements include variables, objective functions, constraint functions, and response functions. The language provides an intuitive environment for compact specification of partitioned optimization problems. Subproblems can be defined, and subsequently assembled into larger subsystems and systems. This assembly feature presents a significant advantage for specifying the interactions in large-scale design problems: the specification of variables and their interactions can be carried out locally for subsystems of the system design; the coupling between subsystems can be specified at a higher system level. Typically, these subsystems are associated to a certain physical component or disciplinary aspect of the system design.

4.3 Partitioning

The multidisciplinary DSM is automatically generated from the Ψ specification. Subsequently, structure can be identified by re-ordering of the rows and columns of the matrix. For this we use a partitioning algorithm, to group together design variables, behavioral responses, and objective/constraint responses that are strongly coupled. The partitioning of the DSM is an optimization problem for which specialized algorithms have been developed. For reasonably small problems we have good experience with the Matlab DSM partitioning algorithm of Thebeau (2001). The algorithm implements simulated annealing search as optimization method to minimize cost function:

$$C_{total} = \underbrace{\sum (A(i, j) + A(j, i)) \cdot S_{partition(y)}}_{\text{Interactions within partitions}} + \underbrace{\sum (A(i, j) + A(j, i)) \cdot S_A}_{\text{Interactions outside partitions}} \tag{1}$$

Herein, A is the design structure matrix, $A(i, j)$ denotes matrix entry i, j , S_A denotes the size of the full matrix A , and $S_{partition(y)}$ denotes the size of partition y . The first term in this expression represents the cost associated with the interactions that are within partitions. The second term represents the cost due to interactions outside the partition blocks, that is, the cost due to interactions between partitions.

For larger problems we opted for the graph-based partitioning algorithm Graclus (Dhillon et al. 2007). Graclus

computes a partitioning for a given *undirected* weighted graph and requires as input the graph and the desired number of partitions. The cost function that is minimized is a relative measure (normalized cut) of the weight of interactions between the partitions. Graclus is able to compute partitions of unequal size, unlike Metis (Karypis 2011) for instance. Since Graclus is intended for undirected graphs, direction information (i.e. the asymmetry of the matrix) is accounted for by introducing edge weights to the graph. We assign a connection in one direction a weight of 1; otherwise the connection is assigned a weight of 2.

4.4 Sequencing

Subsequently, sequencing is carried out without affecting the partitioning outcome. As the partitioning outcome is in our case not block angular, we seek the optimal sequence of the partitions such that the amount of feedback coupling between partitions is minimized. For our LED SiP case problem we found the number of partitions to be reasonably small which allows an exhaustive enumeration search procedure. That is, we subdivide the DSM into n^2 sub-matrices, with n the number of partitions. We calculate the sum of the entries of each sub-matrix and place them in an $n \times n$ matrix. We generate this matrix for all possible sequences and calculate the sum of the entries above the main diagonal. The sequence with the lowest sum of entries above the main diagonal presents our solution. Finally, a Dulmage-Mendelsohn decomposition is applied within each partition, which is a well known matrix transformation technique (available for instance in Matlab) to re-order a matrix towards lower-triangular form (Dulmage and Mendelsohn 1958).

We acknowledge that coupling exists between the partitioning and sequencing tasks. However, the combined problem becomes computationally demanding in particular

for large problems. With algorithmic solutions available for the separate problems, we opted for a sequential procedure.

4.5 Procedure

A summary of our approach is as follows:

- 1 Specify the \mathbf{x} , \mathbf{r} , \mathbf{F} interactions using the Ψ language
- 2 Generate the multidisciplinary DSM containing the \mathbf{x} , \mathbf{r} , \mathbf{F} interactions
- 3 Partition the matrix to find the strongly coupled parts
- 4 Find an ordering with minimal number of feedback interactions, by sequencing the partitions, and sequencing within each partition

Figure 9 gives an illustration of the outcome of the procedure for the sample problem presented in Fig. 7. Variables x_2 , x_4 , r_2 , x_1 , and F_2 are grouped together in Partition 1; variables x_3 , r_1 , r_4 , r_3 , F_1 are grouped together in Partition 2.

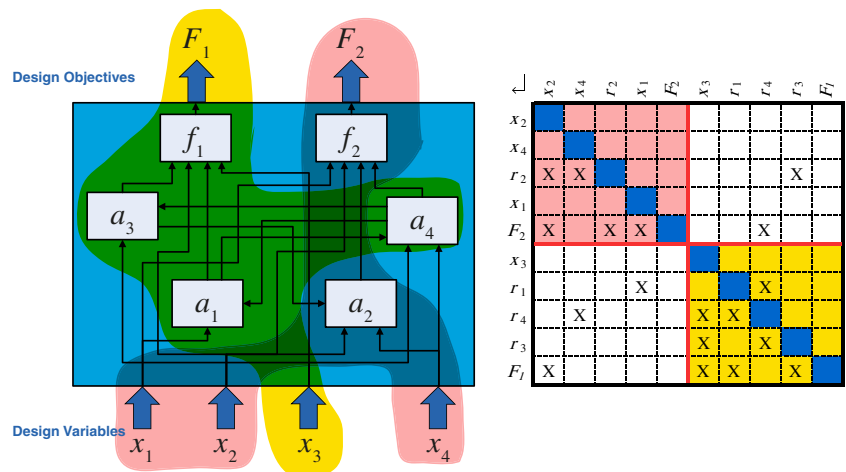
5 Illustrative example

This illustrative example considers a simplified LED system-in-package, consisting of one LED, one driver chip and a leadframe. The aim of this example is to demonstrate the method presented in the previous section, as a prelude to the full LED SiP case presented in Section 6.

5.1 Specification in Ψ

This subsection presents the Ψ specification of the simplified LED SiP. The Ψ specification is developed using a mixed object and aspect decomposition as basis for the description. Four subsystems (objects) are specified:

Fig. 9 Illustration of decomposition based on the system's coupling structure



LED, Driver, leadFrame and Boundary. The thermal, optical, electrical, and mechanical (structural) disciplinary behavior (aspects) is subsequently specified for each of these subsystems. The four systems with their disciplinary aspects are coupled in system LED-SiP representing the complete LED SiP.

The elementary building blocks in Ψ are referred to as *components* (comp). A Ψ component definition contains functions and variables. The functions may be response functions (`resfunc`), objective functions (`objfunc`) and constraint functions (`confunc`). The variables may be internal variables (`intvar`) or external variables (`extvar`). Internal variables occur only in the component, while external variables are variables that are linked to other variables outside the component. The linkage of variables is specified in the *system* environment (`syst`). A system definition contains two or more subsystems (instantiated after keyword `sub`) and describes the coupling between them (after `link`). The subsystems may be components or other systems.

The specification of system LED is shown in Listing 1. Figure 10 shows a schematic overview of the system LED. The system behavior of the LED comprises thermal, optical, and electrical behavior, instantiated as subsystems LT, LO, and LE, respectively. This disciplinary functional behavior is specified in Ψ -components LED_Thermal, LED_Optical, and LED_Electrical, respectively

Component LED_Thermal specifies the temperature behavior within the LED. Response variable T_l , the LED junction temperature (subscript l refers to LED), follows from response function a_1 with input arguments the heat loss of the LED P_l , the LED thermal resistance R_l , the temperature of the driver chip adjacent to the LED T_d (subscript d refers to Driver), lead frame temperature T_f (subscript f refers to Frame), and ambient temperature T_b (subscript b refers to Boundary). Variable R_l occurs only locally in this component, and is therefore specified as an internal variable (`intvar`). The linkage of the variables within system LED is defined after the `link` statement. For instance, `LT.T1 -- LO.T1` means that variable T_l (T1 in the specification) of subsystem LT is linked to variable T1 of subsystem LO.¹ The `alias` statement enables shorthand notation when variables are passed up to higher levels in the system hierarchy. For instance, variable T_l in subsys-

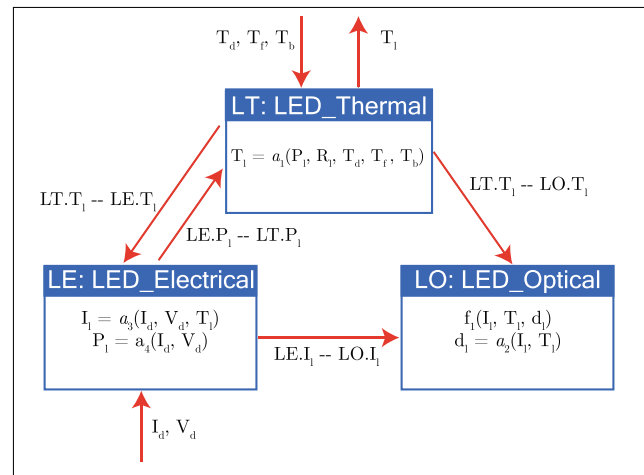


Fig. 10 Schematic overview of Ψ system LED

tem LT of system LED is coupled to systems Driver and leadFrame. This coupling is specified in higher level system LED-SiP (see Listing 5); therefore an alias `T1 = LT.T1` is introduced.

The optical disciplinary Ψ component LED_Optical specifies the light output of the LED as an objective function. The objective function f_l has three input arguments: electrical current I_l , junction temperature T_l and the degradation of the LED d_l . All three inputs are response variables, where d_l is a local response variable that follows from response function a_2 , while I_l and T_l originate from response functions in Ψ components LED_Electrical and LED_Thermal, respectively.

LED_Electrical specifies the electrical behavior of the LED. Two response functions are included: the electrical current in the LED I_l and the heat loss P_l due to electrical deficiencies. They are determined by the supplied current and voltage from the driver electronics I_d and V_d , and the LED junction temperature T_l .

System Driver describes the behavior of the driver electronics, which provides the LED with suitable electrical voltage and current. The Ψ specification of the driver is given in Listing 2 and illustrated in Fig. 11. Ψ component D_Thermal defines the driver temperature as a function of external variables P_d , T_l , T_f , T_b and internal variable R_d , the thermal resistance of the driver. Component D_Electrical specifies two response functions regarding the electrical performance of the driver and its degradation. Note that response function a_6 has three response variables as output: P_d , V_d and I_d . The input variables are the voltage and current from the power supply V_b and I_b , the driver temperature T_d and the degradation of the

¹These variables names need not be the same since variables are locally defined. But for ease of presentation, we choose to use equal variables names for variables that occur in multiple components and that need to be linked.

Listing 1 Ψ specification of the LED

```

syst LED =
| [ sub      LT: LED_Thermal, LO: LED_Optical,
      LE: LED_Electrical
  link      LT.Tl -- LO.Tl, LT.Tl -- LE.Tl,
            LE.I1 -- LO.I1, LE.Pl -- LT.Pl
  alias     Tl = LT.Tl,
            Id = LE.Id, Td = LT.Td, Vd = LE.Vd,
            Tf = LT.Tf, Tb = LT.Tb
  ] |

comp LED_Thermal =
| [ extvar  Pl, Tl, Td, Tf, Tb
  intvar    Rl
  resfunc   Tl = a1( Pl, Rl, Td, Tf, Tb )
  ] |

comp LED_Optical =
| [ extvar  I1, Tl
  intvar    dl
  objfunc   f1( dl, I1, Tl )
  resfunc   dl = a2( I1, Tl )
  ] |

comp LED_Electrical =
| [ extvar  Pl, Tl, I1, Id, Vd
  resfunc   I1 = a3( Tl, Id, Vd ),
            Pl = a4( Id, Vd )
  ] |

```

driver electronics d_d . The driver degradation follows from a_7 with T_d , I_b and V_b as inputs. The link statement in system Driver specifies the coupling of variables P_d and T_d between components D_Thermal and D_Electrical. System Driver provides I_d , V_d , and T_d as output to systems LED and leadFrame, and needs T_l , T_f , T_b , I_b , and V_b as input. These couplings are specified in the higher-level system LED-SiP. Aliases are specified for shorthand notation.

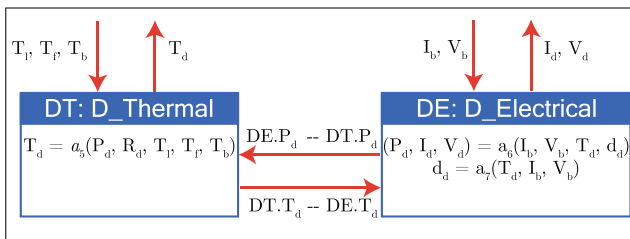


Fig. 11 Schematic overview of Ψ system Driver

The driver and the LED are placed on the lead frame which forms the mechanical platform of the physical components. The lead frame transports heat from the driver and the LED to an attached heat sink. Both mechanical and thermal aspects are present in the behavior of the lead frame. The Ψ specification is given in Listing 3 and a schematic overview of the system is shown in Fig. 12. The linking variable between the mechanical and thermal subsystems is the lead frame temperature T_f . The lead frame temperature response function a_8 has as input arguments T_l , T_d , T_b and the thermal resistance of the lead frame R_f . The mechanical stresses in the lead frame are induced by the temperature in the lead frame T_l , as specified by constraint function f_2 . The variables T_b , T_l , T_d , and T_f are inputs, respectively outputs of the leadFrame system.

The fourth system, Boundary is given in Listing 4. It defines the boundary parameters of the LED SiP system, which include the ambient temperature T_b , electrical input voltage V_b , and input current I_b . These parameters are input

Listing 2 Ψ specification of the Driver

```

syst Driver =
| [ sub      DT: D_Thermal, DE: D_Electrical
  link      DE.Pd -- DT.Pd, DT.Td -- DE.Td
  alias     Id = DE.Id, Td = DT.Td,
           Tl = DT.Tl,
           Tf = DT.Tf
           Ib = DE.Ib, Vb = DE.Vb,
           Tb = DT.Tb, Vd = DE.Vd
  ] |

comp D_Thermal =
| [ extvar   Pd, Td, Tl, Tf, Tb
  intvar    Rd
  resfunc   Td = a5( Pd, Rd, Tl, Tf, Tb )
  ] |

comp D_Electrical =
| [ extvar   Td, Pd, Id, Vd, Ib, Vb
  intvar    dd
  resfunc   (Pd, Id, Vd) = a6( dd, Td, Ib, Vb )
           ,      dd = a7( Td, Ib, Vb )
  ] |
    
```

to the LED and Driver systems and therefore defined as external variables.

The four systems LED, Driver, leadFrame, and Boundary are coupled as shown in Fig. 13. The corresponding Ψ specification of the complete system LED-SiP is given by Listing 5. The four subsystems are instantiated using sub. The couplings are specified using link, as explained before, that is, D.Id -- L.Id means that (alias) variable Id of subsystem D is linked to (alias) variable Id of subsystem L. Another example is L.Tl -- {D.Tl, F.Tl} which links variable Tl of subsystem L to variable Tl of subsystem D, respectively variable Tl

of subsystem F. Statement topsyst LED-SiP means to say that system LED-SiP is considered the top level system which has no input or output of variables external to LED-SiP.

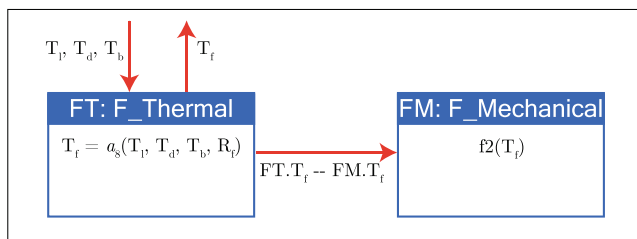


Fig. 12 Schematic overview of Ψ system leadFrame

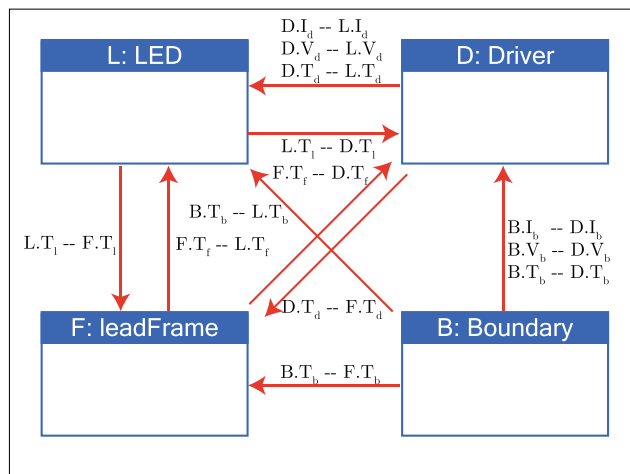


Fig. 13 Schematic overview of Ψ system LED-SiP

Listing 3 Ψ specification of the leadFrame

```

syst leadFrame =
| [ sub      FT: F_Thermal, FM: F_Mechanical
  link      FT.Tf -- FM.Tf
  alias     Tf = FT.Tf,
           Tl = FT.Tl,
           Td = FT.Td,
           Tb = FT.Tb
] |

comp F_Thermal =
| [ extvar   Tf, Tl, Td, Tb
  intvar    Rf
  resfunc   Tf = a8(Rf, Tl, Td, Tb)
] |

comp F_Mechanical =
| [ extvar   Tf
  confunc   f2(Tf)
] |

```

5.2 Multidisciplinary DSM

Figure 14 shows the DSM that is automatically generated from the Ψ specification in Section 5.1. The elements on the horizontal and the vertical axis of the DSM are objective F_1 , constraint F_2 , eight responses $r_1, \dots, r_8 = T_1, d_1, I_1, P_1, T_d, [P_d, I_d, V_d], d_d, T_f$, and six design variables $x_1, \dots, x_6 = T_b, R_1, R_d, R_f, I_b, V_b$.

5.3 Partitioning

Figure 15 shows the DSM after partitioning using the DSM partitioning algorithm of Thebeau (2001). The depicted result is the solution with the lowest cost function value

out of 200 runs of the algorithm (due to the stochastic nature of the simulated annealing search technique every run of the algorithm will produce slightly different partitioning results). The solution shows three partitions of unequal size.

The DSM of Fig. 14 is also partitioned using the Graclus partitioning software (Dhillon et al. 2007). Since Graclus needs the number of partitions a priori, solutions for 2,3, up to 16 partitions are calculated. Cost function (1) is evaluated to rank the outcomes. Figure 16 shows that the minimum cost function value is obtained for the solution of 3 partitions. This partitioning is exactly equal to the partitioning obtained by the algorithm of Thebeau (2001) as displayed in Fig. 15.

Listing 4 Ψ specification of the conditional parameters

```

syst Boundary =
| [ sub      Amb: Ambienttemp, Pow: Powersupply
  alias     Tb = Amb.Tb, Ib = Pow.Ib, Vb = Pow.Vb
] |

comp Ambienttemp =
| [ extvar   Tb
] |

comp Powersupply =
| [ extvar   Ib, Vb
] |

```

Listing 5 Ψ specification of top system LED-SiP

```

syst LED-SiP =
| [ sub      L: LED, D: Driver, F: leadFrame,
           B: Boundary
  link      L.Tl -- {D.Tl, F.Tl},
           D.Id -- L.Id,
           D.Vd -- L.Vd,
           D.Td -- {L.Td, F.Td},
           F.Tf -- {L.Tf, D.Tf},
           B.Tb -- {L.Tb, D.Tb, F.Tb},
           B.Ib -- D.Ib,
           B.Vb -- D.Vb
  ] |

top syst LED-SiP

```

5.4 Sequencing

Figure 17 shows the DSM, where the partitions are reordered such that the number of interaction marks above the diagonal outside the partitions is minimized. Furthermore the elements within the partitions are sequenced such that the number of interactions above the diagonal are minimized within the partitions. Note that the resulting sequence

of elements within the three partitions is, first the design variables, then responses, and finally objectives/constraints.

The partitioning and sequencing analysis shows that the example problem should be decomposed in three partitions, corresponding to respectively the thermal behavior ($x_1, x_2, x_3, x_4, r_1, r_5, r_8, F_2$), the electrical behavior of the driver (x_5, x_6, r_7, r_6, r_4), and the electrical and optical behavior of the LED (r_3, r_2, F_1). The sequence reveals

Fig. 14 Multidisciplinary DSM of the illustrative example

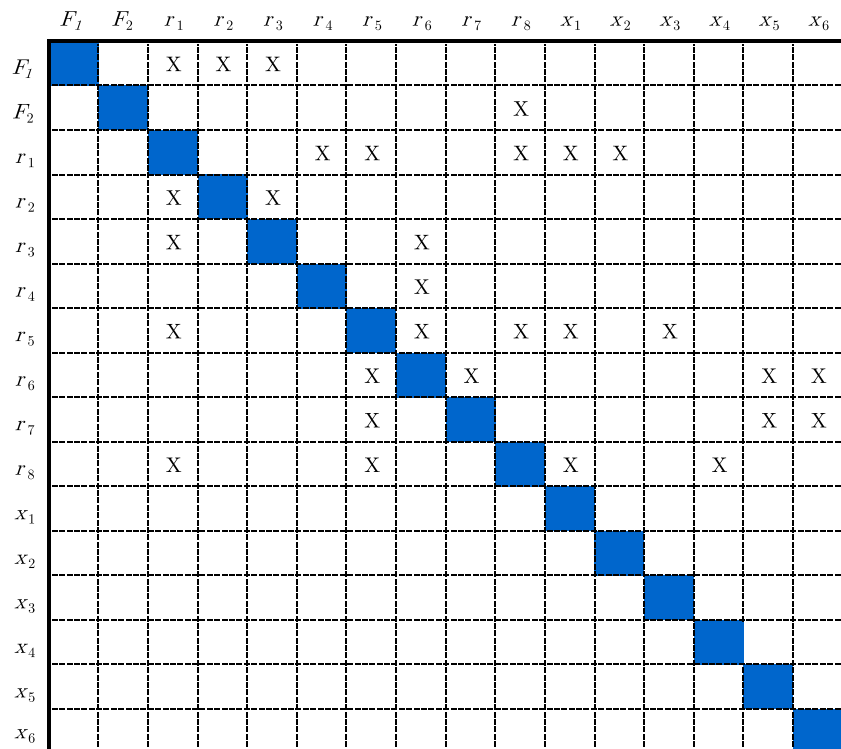
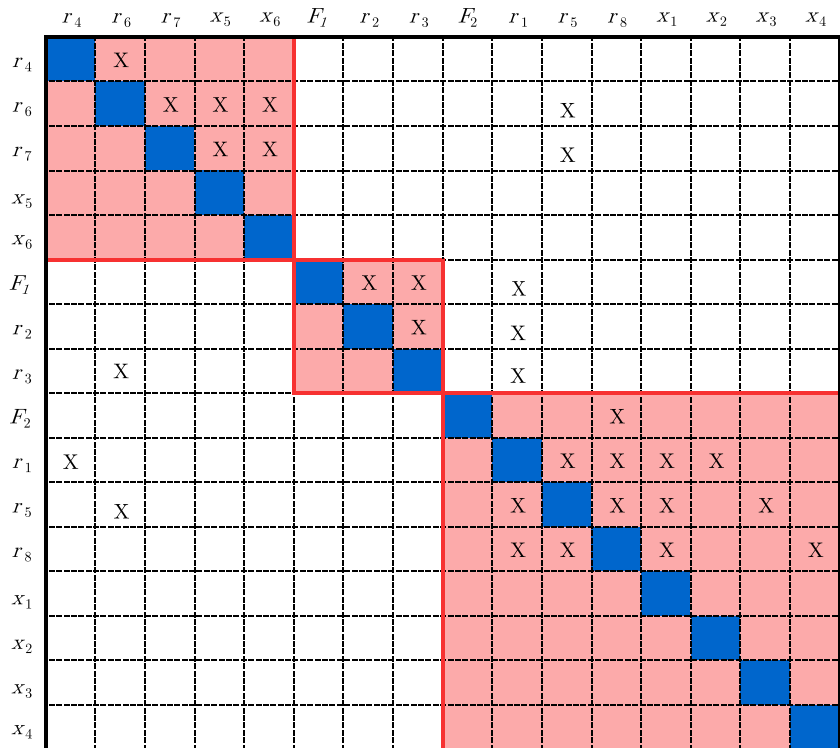


Fig. 15 Partitioned multidisciplinary DSM of the illustrative example



that for the illustrative example problem the design should start with the thermal design, followed by electrical design, and finally optical design. Finally we observe that the resulting partitioning is entirely disciplinary (aspect) oriented, whereas in the building process of our Ψ specification we departed from a mixed object and aspect decomposition. We

will return to these observations in the next section when we study the industry SiP design case.

6 LED SiP design case

A LED SiP prototype developed by TNO and industry partners is used as a case to validate the proposed four step procedure. Figure 18 shows the recently manufactured prototype of the LED SiP. In this study we assume an early phase of the design, where design decisions have not yet been made, so without prejudice of the current LED SiP design prototype. The system elements and their interactions were obtained by means of a literature review and interviews with designers. The literature provided background knowledge regarding the physical working principles of the various components of the SiP. The interviews provided information on the functionality, components, and disciplinary aspects that should be considered to design a LED SiP. Furthermore, the interviews revealed that the design engineers prefer to first decompose the LED SiP into physical components, and subsequently decompose each component regarding the relevant disciplinary aspects. Also, there was general agreement that thermal and mechanical disciplinary analysis is necessary not only at the component

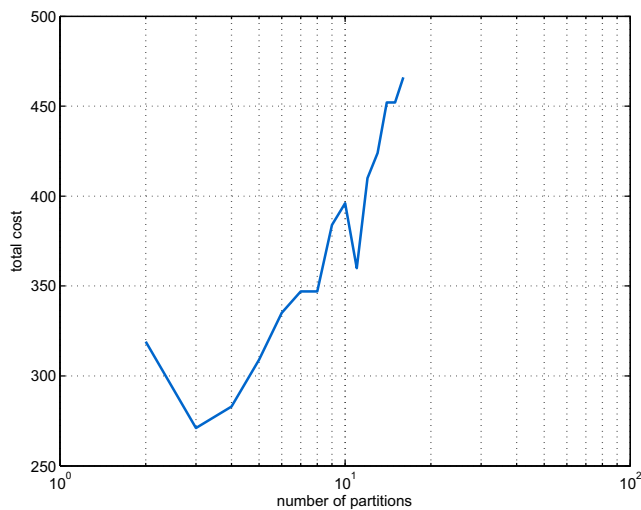
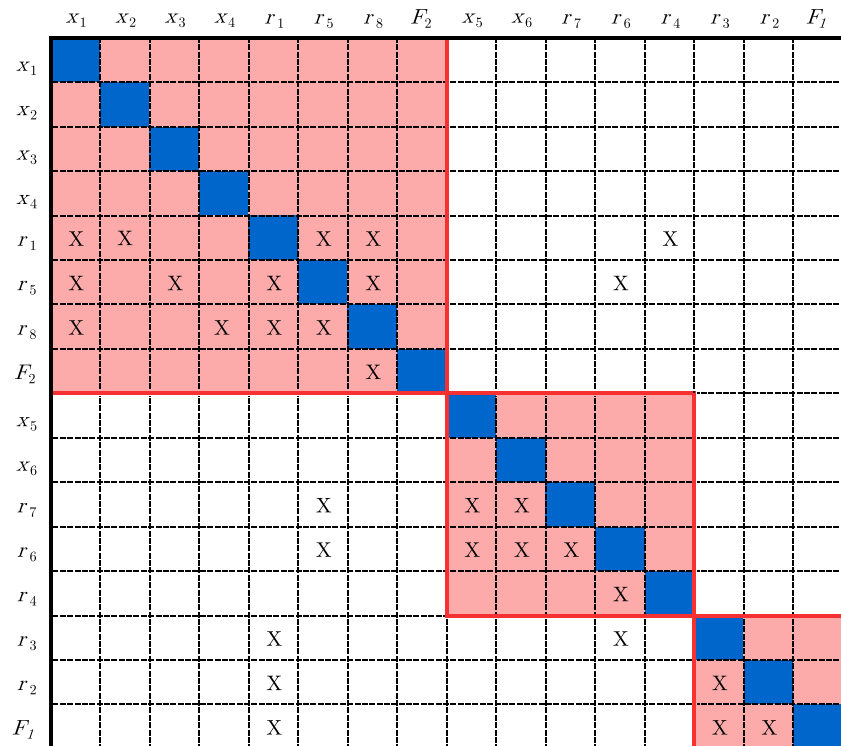


Fig. 16 Calculated costs for the illustrative example

Fig. 17 Partitioned and sequenced multidisciplinary DSM of the illustrative example



level, but also at the level of the complete LED SiP design, to guarantee proper functioning.

6.1 Specification in Ψ

The preference by the engineers to first consider components and subsequently consider disciplines is used as

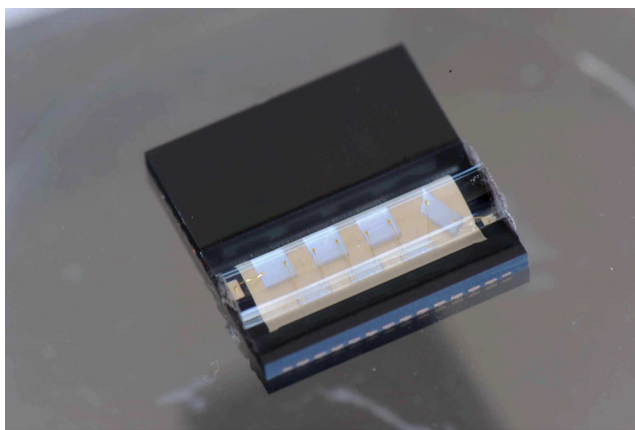


Fig. 18 First prototype of the LED SiP (courtesy of TNO)

framework to build the Ψ specification of the LED SiP (similar to Section 5.1). In the Ψ specification, LED dies, driver chips, lead frame, transparent encapsulant, phosphor film, black compound, interconnects, I/O pads, die attach, heat sink, and PCB components are distinguished. For each of these subsystems, corresponding thermal, optical, electrical and mechanical (structural) disciplinary Ψ components are specified. The behavior is specified by means of variables, response functions, constraint functions and objective functions. Additionally, a thermal and a mechanical analysis system are included representing system wide analysis. Grand total, the LED SiP specification consists of 45 Ψ (sub)systems, where each system may contain one or more disciplinary Ψ components. The systems are assembled by means of linkage of variables.

6.2 Multidisciplinary DSM

From the Ψ specification the DSM containing the interactions between design variables, behavioral responses, and objective/constraint responses was automatically generated. The generated matrix has a size of 711 by 711 elements. Among these 711 elements there are 450 design/parameter variables, 253 behavioral responses, 6 constraint responses and 2 design objective responses.

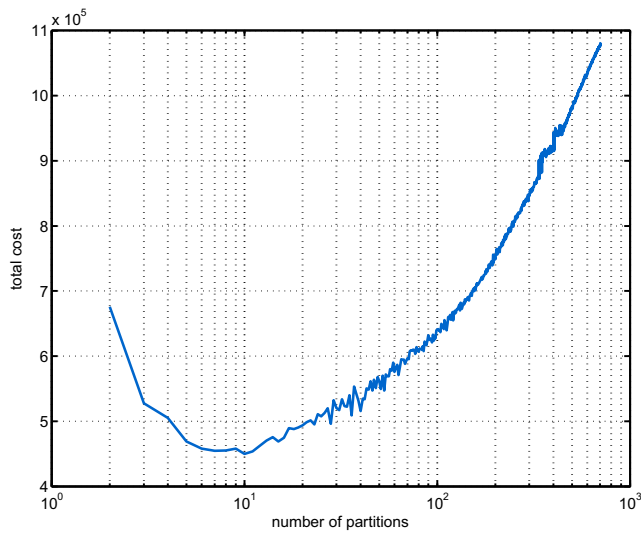


Fig. 19 Calculated costs for the LED SiP design case

6.3 Partitioning

Graph partitioning using Graclus is applied to the DSM. Since the desired number of partitions is unknown, the DSM matrix is partitioned for a range of numbers of partitions, from 2 to approximately 700 partitions. The cost function given by (1), due to Thebeau (2001), is used to evaluate the trade-off between the number of interactions between partitions and the size of the partitions. Figure 19 shows the resulting calculated costs for the range of partitioning solutions. The cost value first decreases as the number of partitions increases, which is caused by the decrease in partition size. Then the cost value increases again, as a result of an increasing number of interactions outside the partitions. The minimum value for this cost function is found around ten partitions. Note that by changing the weights between the two parts of the cost function the minimum may shift to the left or to the right.

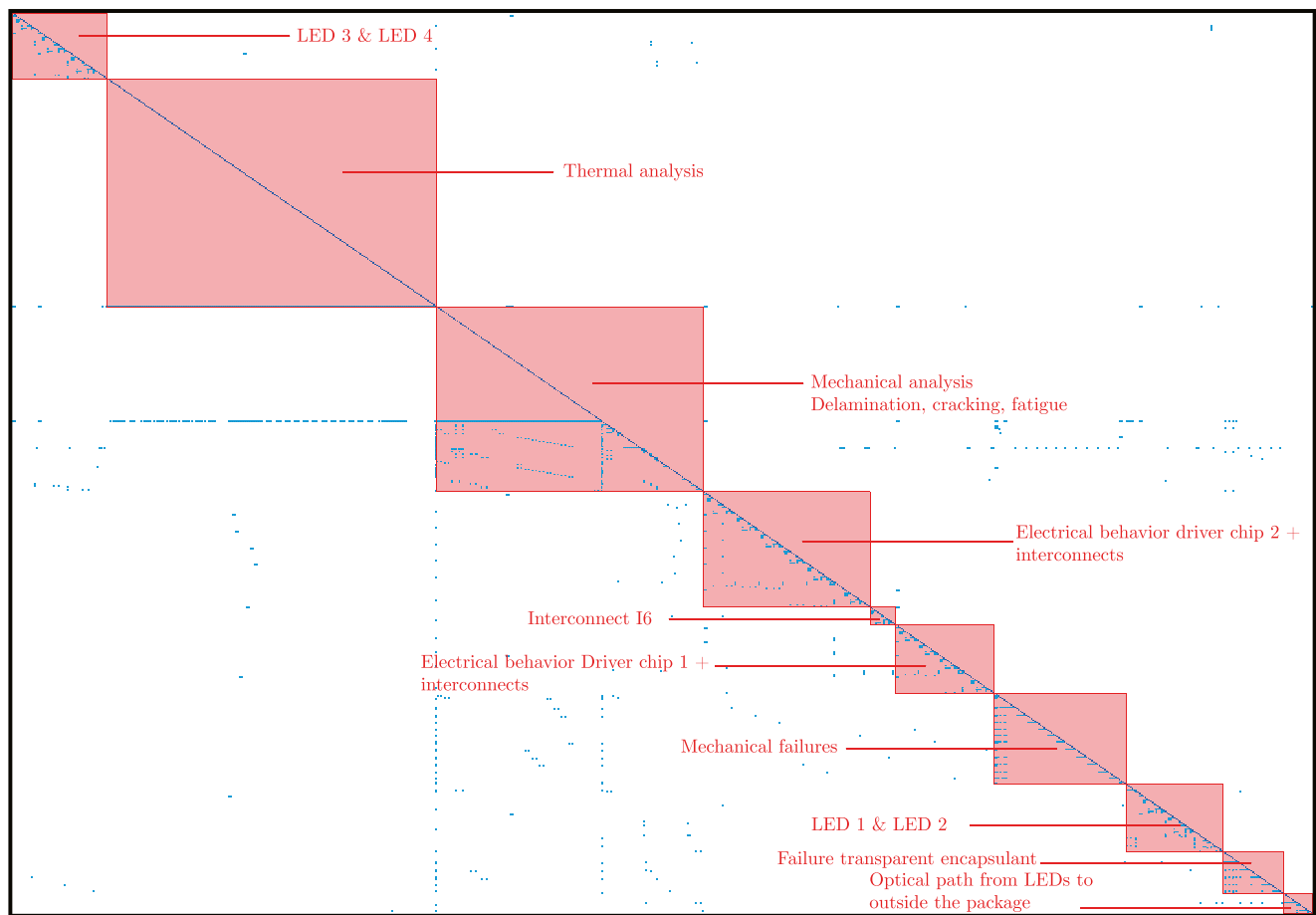


Fig. 20 Partitioned and sequenced multidisciplinary DSM of the LED SiP design for a decomposition consisting of ten partitions

6.4 Sequencing

Figure 20 shows the multidisciplinary DSM for the solution with ten partitions after additional sequencing. Each partition represents a strongly coupled part of the design. The thermal analysis partition and the mechanical analysis partition are both large partitions and are placed early in the sequence. This contradicts with current design practice to do the mechanical and thermal analysis during the final stage of design once the SiP system has been almost completed. We also observe that some partitions in Fig. 20 are aspect (disciplinary) oriented while other partitions are object (component) oriented. A few partitions have a mixed character which follows from the tight coupling between the design of the interconnects and the electrical design. The disciplinary partitions are mostly placed early in the sequence, while the component partitions are placed later in the sequence. This suggests to depart the design process from a disciplinary decomposition of the SiP and to gradually transform towards a component based decomposition.

7 Concluding remarks

The multidisciplinary coupling structure is modeled by means of a design structure matrix (DSM) using design variables, behavioral response variables, and objective/constraint responses as rows and columns of the matrix. The DSM represents the input-output functional relations between these variables in a binary form. By partitioning and sequencing of such a DSM with behavioral response variables included, the multidisciplinary coupling structure can be revealed. For the LED SiP prototype the partitioned and sequenced DSM consists of a mixture of discipline and component related partitions. The matrix shows that thermal and mechanical system analysis should play a more prominent role in guiding the design process of the SiP.

To generate the DSM, the use of a specification language to specify the linkages is advantageous. The Ψ language proved to be effective, since Ψ allows to specify relationships in subproblems using locally defined variables, and linkage of variables through a bottom-up assembly using systems. The multidisciplinary DSM can be automatically generated from the Ψ specification. We have noticed that there are several opportunities to further enhance Ψ . For instance the use of typed data and arrays can make the Ψ specifications more compact in large-scale applications. We will elaborate on this in our future work.

Due to computational reasons we had to carry out the partitioning and sequencing sequentially. Also the number of

partitions was not considered as part of the partitioning decision, since Graclus does not allow this. We simply varied the number of partitions as input to the algorithm. A DSM partitioning algorithm based on simulated annealing which optimizes both for partition size and interaction proved too computationally demanding for our SiP application. For our application we need efficient computational algorithms that can do simultaneous partitioning and sequencing on DSM matrices of thousands of elements. Another topic for further research is to incorporate the coordination effort induced by the decomposition into the partitioning problem. Allison et al. (2009) presented some first results on this topic for small test problems.

Acknowledgments The authors would like to thank the designers and engineers at TNO, involved in LED SiP design, for their valuable contributions during the several interviews.

Open Access This article is distributed under the terms of the Creative Commons Attribution 4.0 International License (<http://creativecommons.org/licenses/by/4.0/>), which permits unrestricted use, distribution, and reproduction in any medium, provided you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license, and indicate if changes were made.

References

- Alexandrov NM (2005) Editorial – multidisciplinary design optimization. *Optim Eng* 6:5–7
- Alexandrov NM, Lewis RM (2004) Reconfigurability in MDO problem synthesis, part 1. In: Proceedings of the 10th AIAA/ISSMO multidisciplinary analysis and optimization conference, Albany, NY, pp AIAA 2004-4307. doi:[10.2514/6.2004-4307](https://doi.org/10.2514/6.2004-4307)
- Allison JT, Herber DR (2014) Multidisciplinary design optimization for dynamic engineering systems. *AIAA J* 52(4):691–710. doi:[10.2514/1.J052182](https://doi.org/10.2514/1.J052182)
- Allison JT, Kokkolaras M, Papalambros PY (2009) Optimal partitioning and coordination decisions in decomposition-based design optimization. *J Mech Des* 131(8):081.008. doi:[10.1115/1.3178729](https://doi.org/10.1115/1.3178729)
- Altus SS, Kroo IM, Gage PJ (1996) A genetic algorithm for scheduling and decomposition of multidisciplinary design problems. *J Mech Des* 118(4):486–489. doi:[10.1115/1.2826916](https://doi.org/10.1115/1.2826916)
- Balling RJ, Sobieszcanski-Sobieski J (1996) Optimization of coupled systems: a critical overview of approaches. *AIAA J* 34(1):6–17. doi:[10.2514/3.13015](https://doi.org/10.2514/3.13015)
- Browning TR (2001) Applying the design structure matrix to system decomposition and integration problems: a review and new directions. *IEEE Trans Eng Manag* 48(3):292–306. doi:[10.1109/17.946528](https://doi.org/10.1109/17.946528)
- Chang M, Das D, Varde PV, Pecht M (2012) Light emitting diodes reliability review. *Microelectron Reliab* 52(5):762–782. doi:[10.1016/j.microrel.2011.07.063](https://doi.org/10.1016/j.microrel.2011.07.063)
- Chen L, Macwan A, Li S (2007) Model-based rapid redesign using decomposition patterns. *J Mech Des* 129(3):283–294. doi:[10.1115/1.2406099](https://doi.org/10.1115/1.2406099)
- Clarkson PJ, Simons C, Eckert C (2004) Predicting change propagation in complex design. *J Mech Des* 126(5):788–797. doi:[10.1115/1.1765117](https://doi.org/10.1115/1.1765117)

- Cramer EJ, Dennis JE, Frank PD, Lewis RM, Shubin R (1994) Problem formulation for multidisciplinary optimization. *SIAM J Optim* 4(4):754–776. doi:10.1137/0804044
- de Borst ECM, Gielen AWJ, Etman LFP (2012) Coupling structure in LED System-in-Package design: a physical responses-based critical parameter sheet like approach. In: Proceedings of the 4th electronics system integration technologies conference (ESTC). IEEE, Amsterdam, pp 1–6. doi:10.1109/ESTC.2012.6542063
- Dhillon IS, Guan Y, Kulis B (2007) Weighted graph cuts without eigenvectors a multilevel approach. *IEEE Trans Pattern Anal Mach Intell* 29(11):1944–1957
- Dong Q (2002) Predicting and managing system interactions at early phase of the product development process. Ph.D. thesis, Massachusetts Institute of Technology. <http://hdl.handle.net/1721.1/16881>
- Dulmage AL, Mendelsohn NS (1958) Coverings of bipartite graphs. *Can J Math* 10:517–534
- Eppinger SD, Browning TR (2012) Design structure matrix methods and applications. Engineering Systems. MIT Press, Cambridge, MA
- Gielen AWJ, Heslen P, Swartjes F, van Zeijl H, Boschman F, Bullema J, Werkhoven RJ, Koh S (2011) Development of an intelligent integrated LED system-in-package. In: Proceedings of the 18th european microelectronics and packaging conference (EMPC), IEEE, pp 1–7
- Hechfellner R, Landau S (2009) Understanding LED performance. *Electronics Weekly* 30 September - 6 October, www.philipslumileds.com/uploads/14/NA05-pdf
- Hong E, Narendran N (2004) A method for projecting useful life of LED lighting systems. In: Proceedings of the international conference on solid state lighting, vol SPIE 5187, pp 93–99. doi:10.1117/12.509682
- Jung S, Park G, Choi D (2013) A decomposition method for exploiting parallel computing including the determination of an optimal number of subsystems. *J Mech Des* 135(4):041,005. doi:10.1115/1.4023554
- Karypis G (2011) Metis-unstructured graph partitioning and sparse matrix ordering system, version 5.0. <http://glaros.dtc.umn.edu/gkhome/views/metis>
- Koh S, Van Driel W, Zhang GQ (2011) Degradation of epoxy lens materials in LED systems. In: Proceedings of the 12th eurosime conference on thermal, mechanical and multi-physics simulation and experiments in microelectronics and microsystems. IEEE, Linz, pp 1–5. doi:10.1109/ESIME.2011.5765850
- Lambe AB, Martins JRRA (2012) Extensions to the design structure matrix for the description of multidisciplinary design, analysis, and optimization processes. *Struct Multidiscip Optim* 46(2):273–284. doi:10.1007/s00158-012-0763-y
- Li S (2010a) Extension of the two-phase method for decomposition of matrix-based design systems. *J Mech Des* 132(6):061,003. doi:10.1115/1.4001534
- Li S (2010b) Pattern-based reasoning for rapid redesign: a proactive approach. *Res Eng Des* 21(1):25–42. doi:10.1007/s00163-009-0069-2
- Lu Z, Martins JRRA (2012) Graph partitioning-based coordination methods for large-scale multi-disciplinary design optimization problems. In: Proceedings of the 14th AIAA/ISSMO multidisciplinary analysis optimization conference. AIAA, Indianapolis, pp AIAA 2012-5522. doi:10.2514/6.2012-5522
- Martins JRRA, Lambe AB (2013) Multidisciplinary design optimization: a survey of architectures. *AIAA J* 51(9):2049–2075. doi:10.2514/1.J051895
- Meier C, Yassine AA, Browning TR (2007) Design process sequencing with competent genetic algorithms. *J Mech Des* 129:566–585. doi:10.1115/1.2717224
- Michelen NF, Papalambros PY (1997) A hypergraph framework for optimal model-based decomposition of design problems. *Comput Optim Appl* 8(2):173–196. doi:10.1023/A:1008673321406
- Narendran N, Gu Y (2005) Life of LED-based white light sources. *J Disp Technol* 1(1):167–171. doi:10.1109/JDT.2005.852510
- Rogers JL (1989) A knowledge-based tool for multilevel decomposition of a complex design problem. Technical Report. Technical paper 2903, NASA
- Rogers JL (1996) Demaid/ga – an enhanced design manager's aid for intelligent decomposition (genetic algorithms). In: Proceedings of the 6th AIAA/NASA/ISSMO symposium on multidisciplinary analysis and optimization, Bellevue, WA, pp 1497–1504, AIAA Paper 96-4157. doi:10.2514/6.1996-4157
- Sangal N, Jordan E, Sinha V, Jackson D (2005) Using dependency models to manage complex software architecture. In: Proceedings of the 20th annual ACM SIGPLAN conference on object-oriented programming, systems, languages, and applications, pp 167–176. doi:10.1145/1094811.1094824
- Schmitz S, Wynn D, Biedermann W, Lindemann U, Clarkson PJ (2011) Improving data quality in DSM modelling: a structural comparison approach. In: Proceedings of the 18th international conference on engineering design (ICED11), vol 4, pp 369–380
- Sobieszczanski-Sobieski J, Haftka RT (1997) Multidisciplinary aerospace design optimization: survey of recent developments. *Struct Multidiscip Optim* 14(1):1–23. doi:10.1007/BF01197554
- Sosa ME, Eppinger SD, Rowles CM (2004) The misalignment of product architecture and organizational structure in complex product development. *Manag Sci* 50(12):1674–1689. doi:10.1287/mnsc.1040.0289
- Steward DV (1981) The design structure system: A method for managing the design of complex systems. *IEEE Trans Eng Manag* 28(3):71–74
- Suh ES, Furst MR, Mihalyov KJ, de Weck O (2010) Technology infusion for complex systems: a framework and case study. *Syst Eng* 13(2):186–203. doi:10.1002/sys.20142
- Tarashoon S, Koh SW, van Driel WD, Zhang GQ (2011) Temperature dependency in performance of solid state lighting drivers. In: Proceedings of the 12th eurosime conference on thermal, mechanical and multi-physics simulation and experiments in microelectronics and microsystems. IEEE, Linz, pp 1–3. doi:10.1109/ESIME.2011.5765865
- Thebeau RE (2001) Knowledge management of system interfaces and interactions from product development processes. Ph.D. Thesis, Massachusetts Institute of Technology, <http://hdl.handle.net/1721.1/29168>
- Tilstra AH, Seepersad CC, Wood KL (2010) The repeatability of high definition design structure matrix (HDDSM) models for representing product architecture. In: Proceedings of the asme 2010 international design engineering technical conferences. ASME, Quebec, pp 529–542. doi:10.1115/DETC2010-28717
- Tosserams S, Etman LFP, Rooda JE (2009) A classification of methods for distributed system optimization based on formulation structure. *Struct Multidiscip Optim* 39(5):503–517. doi:10.1007/s00158-008-0347-z
- Tosserams S, Hofkamp AT, Etman LFP, Rooda JE (2010) A specification language for problem partitioning in decomposition-based design optimization. *Struct Multidiscip Optim* 42(5):707–723. doi:10.1007/s00158-010-0512-z
- Wagner TC (1993) A general decomposition methodology for optimal system design. Ph.D. thesis, University of Michigan
- Wagner TC, Papalambros PY (1993) A general framework for decomposition analysis in optimal design. In: Gilmore BJ, Hoeltzel DA, Azarm S, Eschenauer H (eds) Proceedings Asme advances in design automation, DE-Vol 65-2. Albuquerque, pp 315–325