Psi reference manual

Systems Engineering Group
Department of Mechanical Engineering
Eindhoven University of Technology
PO Box 513
5600 MB Eindhoven
The Netherlands
http://se.wtb.tue.nl/

SE Report: Nr. 2009-04

Ψ reference manual

S. Tosserams, A.T. Hofkamp, L.F.P. Etman, J.E. Rooda

ISSN: 1872-1567

SE Report: Nr. 2009-04
Eindhoven, June 2009
SE Reports are available via http://se.wtb.tue.nl/sereports

Contents

1 Introduction **5**

2 Decomposition-based optimization for engineering system design **7**
1 Optimal design problem in integrated form 7
2 The partitioned problem 8

3 Partition specification language Ψ **11**
1 Components 11
2 Systems 13

4 Processing of specifications **17**
1 Normalized partition 17
2 FDT generator 19

5 Examples **21**
1 Geometric programming 21
2 Speed reducer 23
3 Portal frame 27
4 Vehicle chassis 30
5 Supersonic business jet 34
6 Micro-accelerometer 39

Bibliography **45**

A Grammar **47**

B Requirements **51**
1 Specification 51
2 Components 51
3 Systems 52
4 Topsyst 54

C Specification and generated outputs for examples **55**
1 Example (3.1) 55
2 Geometric programming problem 57
3 Speed reducer 64
4 Portal frame 70
5 Chassis design 76
6 Business jet 86
7 Micro-accelerometer 99

Chapter 1
Introduction

The Ψ partition specification language [22] is a flexible and intuitive language for specification of problem partitions in decomposition-based design optimization. This reference manual introduces the constructs of the language, and illustrates how problem partitions can be specified using Ψ. A description of the language in terms of its grammar and semantic requirements is included to provide a formal definition of the language. The style of writing and the many presented examples make this manual a suitable introduction to the language and its possibilities.

Intended users of Ψ are researchers working in the field of decomposition-based design optimization. The language provides a tool for intuitive partition specification that can be also used for exploring alternative problem partitions. The language is not a stand-alone tool, but should be considered to be a pre-processor for computational frameworks that coordinate the solution of the decomposition problem. Ψ provides a *specification* of the partition of the problem; the *solution* of the specified problem is obtained by a computational framework. Examples of computational frameworks are presented in [4, 5, 8, 12, 13, 14]. The motivation behind Ψ is very similar to that of the AMPL language [6], which provides a problem specification approach for nonlinear programming frameworks. The difference is that Ψ applies to partitioned problems that consist of several optimization subproblems, and AMPL is targeted at integrated, monolithic optimization problems.

A number of steps are involved in specifying and solving a decomposed problem using Ψ:

1. Specification of the partitioned problem in Ψ (how variables and functions are distributed over the problem)
2. Conversion of the Ψ specification to the format suitable for a computational framework.
3. Definition of variables and functions (in terms of their bounds, initial guesses, associated numerical routines, etc.) such that the computational framework can use them.
4. Solution of the partitioned problem (using a computational framework)

In the first step, the partitioned problem is specified using the Ψ language. This specification is converted to a format suitable for interpretation by a computational framework in the second

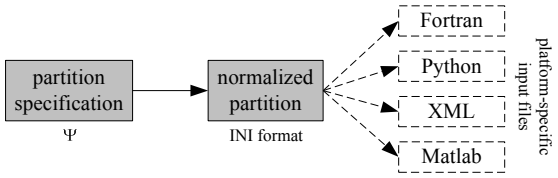


Figure 1.1: Relations between the different partition specification formats. Boxes represent partition formats, and arrows are associated with translators/generators. The shaded boxes and solid arrows represented specification formats and translators described in this manual.

step. In the third step, relevant attributes are assigned to variables and functions used in the Ψ specification. Since Ψ only defines how variables and functions are distributed over various components and systems, these numerical attributes have to be supplied separately. Once the partition specification and variable and function definitions are available, the problem is solved in Step 4 with an appropriate coordination method implemented in a computational framework.

Several tools are available for coupling partition specifications in Ψ to computational frameworks:

- An error-checker that checks whether a Ψ specification meets the semantic requirements of the language.
- A convertor/compiler to create normalized machine-readable (partition) specifications in INI format from Ψ specifications. From these normalized partitions, platform-specific input files for the computational coordination frameworks can be generated.

Normalized specifications in the INI format are generic, and independent of the language used to program the computational frameworks (for example Python, Matlab, C++, Fortran, or XML). Normalized partitions are machine-readable, and straightforward input file generation is possible for the various computational frameworks. Note that these input file generators are framework-specific, and beyond the scope of this manual or the Ψ language. The relations between the different partition formats is illustrated in Figure 1.1.

Due to the generic nature of normalized partition, conversion to other output formats is also possible. For example, a generator that constructs functional dependence tables (FDT) from normalized partitions is available as well. A functional dependence table is a compact, matrix representation of the structure of the problem [11]. These FDTs, or related representations, are often used by automated partitioning methods that derive partitions that can be coordinated efficiently.

This reference manual is organized as follows. First, the general system design problem and its decomposition are introduced in Section 2. Second, we present the language elements of Ψ in Section 3, followed by the description of the associated tools in Section 4. Third, a number of examples are presented in Section 5 to demonstrate the use of the Ψ language and its tool set. A definition of Ψ's grammar and the semantic requirements are given in Appendices A and B, respectively. Appendix C includes the literal Ψ specifications and the generated normalized partitions and functional dependence tables for the examples described in this manual.

Chapter 2

Decomposition-based optimization for engineering system design

Decomposition-based optimization approaches are used for the distributed design of large-scale and/or multidisciplinary systems. Decomposition methods consist of two main steps: partitioning and coordination. In partitioning, the optimal design problem is divided into a number of smaller subproblems, each typically associated with a discipline or component of the system. The task of coordination is to drive these individual subproblems towards a design that is consistent, feasible, and optimal for the system as a whole. The main advantage of decomposition methods is that a degree of disciplinary autonomy is given to each subproblem, such that designers are free to select their own analysis and design tools.

In this section, we introduce the general system design problem followed by a description of the two main steps in decomposition: partitioning and coordination.

1 Optimal design problem in integrated form

The starting point of decomposition methods is the system design problem in integrated form:

$$\begin{aligned} \min_{\mathbf{z}} \quad & \mathbf{f}(\mathbf{z}, \mathbf{s}) \\ \text{subject to} \quad & \mathbf{g}(\mathbf{z}, \mathbf{s}) \leq \mathbf{0} \\ & \mathbf{h}(\mathbf{z}, \mathbf{s}) = \mathbf{0} \\ \text{where} \quad & \mathbf{s} = \mathbf{a}(\mathbf{z}, \mathbf{s}) \end{aligned} \tag{2.1}$$

where $\mathbf{z} = [z_1, \ldots, z_n]$ is the vector that contains the design variables of the entire system. Responses $\mathbf{s} = [s_1, \ldots, s_{m_\text{a}}]$ are intermediate quantities computed by analysis functions $\mathbf{a} = [a_1, \ldots, a_{m_\text{a}}]$. With $\mathbf{s} = \mathbf{a}(\mathbf{z}, \mathbf{s})$, we mean to express that each analysis function a_i for response s_i may depend on the *other* responses $s_i | i \neq j$: $s_i = a_i(\mathbf{z}, s_j | j \neq i)$. The response s_i may not depend on itself. $\mathbf{f} = [f_1, \ldots, f_{\text{m}_\text{f}}]$ is the vector of system objective functions, and constraints $\mathbf{g} = [g_1, \ldots, g_{\text{m}_\text{g}}]$ and $\mathbf{h} = [h_1, \ldots, h_{\text{m}_\text{h}}]$ are the collections of all disciplinary inequality and equality constraints, respectively. We refer to the above formulation as integrated since it includes the variables and functions of all disciplines in a single optimization problem.

2 The partitioned problem

The purpose of partitioning is to distribute the variables and functions of integrated problem (2.1) over a number of subproblems. These subproblems are typically mathematical entities that perform (possibly coupled) analyses, evaluate objective and constraint values, or solve optimization problems. The subproblems may therefore (partially) differ from the original disciplines from which the integrated problem was synthesized.

Three main partition types are often identified: aspect-based, object-based, and model-based partitions. Aspect-based partitions follow the human organization of disciplinary experts and analysis tools. Object-based partitions are aligned with the subsystems and components that comprise the system. The third, model-based approach relies on mathematical techniques to obtain an appropriately balanced partition computationally. Model-based partitioning methods often rely on graph theory or matrix representations of problem structure.

Partitioning problem (2.1) requires a distribution of the variables and functions over a number of subproblems. To this end, the variables $\mathbf{z}$ are partitioned into M sets of local variables $\mathbf{x}_j$ allocated to subproblems $j = 1, \ldots, M$, and a set of coupling variables $\mathbf{y}$ that influence two or more subproblems. Similarly, responses $\mathbf{s}$ are partitioned into M sets of subproblem responses $\mathbf{s}_j$, $j = 1, \ldots, M$, and a set of system-wide responses $\mathbf{s}_0$. Each of these sets is further partitioned into a set of local responses $\mathbf{u}_j$ and a set of coupling responses $\mathbf{r}_j$, $j = 0, 1, \ldots, M$. The local responses $\mathbf{u}_j$ for $j = 1, \ldots, M$ are similar to local variables $\mathbf{x}_j$ and are exclusively associated with subproblem j. Local system responses $\mathbf{u}_0$ are exclusively used as an argument to the coupling functions $\mathbf{f}_0$, $\mathbf{g}_0$, $\mathbf{h}_0$, and $\mathbf{a}_0$ that are defined below. Coupling responses $\mathbf{r} = [\mathbf{r}_0, \mathbf{r}_1, \ldots, \mathbf{r}_M]$ are relevant to two or more subproblems, similar to the coupling variables $\mathbf{y}$.

The analysis functions $\mathbf{a}$ are partitioned into M local analysis functions $\mathbf{a}_j$, $j = 1, \ldots, M$, that compute $\mathbf{s}_j = [\mathbf{r}_j, \mathbf{u}_j]$ and a set of coupling analysis functions $\mathbf{a}_0$ that determine $\mathbf{s}_0 = [\mathbf{r}_0, \mathbf{u}_0]$. The local analysis functions $\mathbf{a}_j(\mathbf{y}, \mathbf{r}, \mathbf{x}_j, \mathbf{u}_j)$ may depend on the coupling variables, the coupling responses, and the local variables and local responses for subproblem j. With the dependency of $\mathbf{a}_j$ on responses $\mathbf{s}_j = [\mathbf{r}_j, \mathbf{u}_j]$, we express that an analysis function a_i of $\mathbf{a}_j$ for response s_i may depend on the other responses $s_k | k \neq i$ of $\mathbf{s}_j$. Analysis functions $\mathbf{a}_0(\mathbf{y}, \mathbf{r}, \mathbf{u}_0, \mathbf{x}_1, \mathbf{u}_1, \ldots, \mathbf{x}_M, \mathbf{u}_M)$ can depend on all variables and all responses. The objective functions $\mathbf{f}$ are assumed to have coupling functions $\mathbf{f}_0(\mathbf{y}, \mathbf{r}, \mathbf{u}_0, \mathbf{x}_1, \mathbf{u}_1, \ldots, \mathbf{x}_M, \mathbf{u}_M)$ that may depend on all variables and responses, and local functions $\mathbf{f}_j(\mathbf{y}, \mathbf{r}, \mathbf{x}_j, \mathbf{u}_j)$, $j = 1, \ldots, M$ that depend only on one set of local variables $\mathbf{x}_j$, local responses $\mathbf{u}_j$, the coupling variables $\mathbf{y}$, and coupling responses $\mathbf{r}$. Similarly, the constraints $\mathbf{g}$ and $\mathbf{h}$ can be partitioned into coupling constraints $\mathbf{g}_0(\mathbf{y}, \mathbf{r}, \mathbf{u}_0, \mathbf{x}_1, \mathbf{u}_1, \ldots, \mathbf{x}_M, \mathbf{u}_M)$ and $\mathbf{h}_0(\mathbf{y}, \mathbf{r}, \mathbf{u}_0, \mathbf{x}_1, \mathbf{u}_1, \ldots, \mathbf{x}_M, \mathbf{u}_M)$, and a set of local constraints for each subproblem $\mathbf{g}_j(\mathbf{y}, \mathbf{r}, \mathbf{x}_j, \mathbf{u}_j)$ and $\mathbf{h}_j(\mathbf{y}, \mathbf{r}, \mathbf{x}_j, \mathbf{u}_j)$.

Under these conventions, the partitioned problem is defined as:

$$
\begin{array}{rll}
\underset{\mathbf{y}, \mathbf{x}_1, \ldots, \mathbf{x}_M}{\min} & [\mathbf{f}_0(\mathbf{y}, \mathbf{r}, \mathbf{u}_0, \mathbf{x}_1, \mathbf{u}_1, \ldots, \mathbf{x}_M, \mathbf{u}_M), & \\
& \quad \mathbf{f}_1(\mathbf{y}, \mathbf{r}, \mathbf{x}_1, \mathbf{u}_1), \ldots, \mathbf{f}_M(\mathbf{y}, \mathbf{r}, \mathbf{x}_M, \mathbf{u}_M)] & \\
\text{subject to} & \mathbf{g}_0(\mathbf{y}, \mathbf{r}, \mathbf{u}_0, \mathbf{x}_1, \mathbf{u}_1, \ldots, \mathbf{x}_M, \mathbf{u}_M) \leq \mathbf{0} & \\
& \mathbf{h}_0(\mathbf{y}, \mathbf{r}, \mathbf{u}_0, \mathbf{x}_1, \mathbf{u}_1, \ldots, \mathbf{x}_M, \mathbf{u}_M) = \mathbf{0} & \\
& \mathbf{g}_j(\mathbf{y}, \mathbf{r}, \mathbf{x}_j, \mathbf{u}_j) \leq \mathbf{0} & j = 1, \ldots, M \\
& \mathbf{h}_j(\mathbf{y}, \mathbf{r}, \mathbf{x}_j, \mathbf{u}_j) = \mathbf{0} & j = 1, \ldots, M \\
\text{where} & \mathbf{r} = [\mathbf{r}_0, \mathbf{r}_1, \ldots, \mathbf{r}_M] & \\
& (\mathbf{r}_j, \mathbf{u}_j) = \mathbf{a}_j(\mathbf{y}, \mathbf{r}, \mathbf{x}_j, \mathbf{u}_j) & j = 1, \ldots, M \\
& (\mathbf{r}_0, \mathbf{u}_0) = \mathbf{a}_0(\mathbf{y}, \mathbf{r}, \mathbf{u}_0, \mathbf{x}_1, \mathbf{u}_1, \ldots, \mathbf{x}_M, \mathbf{u}_M) &
\end{array}
\tag{2.2}
$$

Although the above formulation assumes that integrated problem (2.1) possesses a certain sparsity structure (the presence of local variables and functions), it is general enough to encompass many practical engineering problems.

A common technique of decomposition methods is to introduce a number of auxiliary variables to separate the local functions with respect to the variables of each subproblem. For the coupling variables $\mathbf{y}$, we can introduce separate copies $\mathbf{y}_1, \ldots, \mathbf{y}_M$ at each subproblem, where each $\mathbf{y}_j$ only contains those coupling variables of $\mathbf{y}$ that are relevant to subproblem j. To enforce consistency between the coupling variables, consistency constraints are introduced:

$$\mathbf{c}^{\mathrm{y}}_{jn} = S_{jn}\mathbf{y}_j - S_{nj}\mathbf{y}_n = \mathbf{0} \qquad j = 1, \ldots, M, \qquad n \in \mathcal{N}_j \tag{2.3}$$

where the binary selection matrix S_{jn} selects those copies of subproblem j that are linked to the selected copies of its neighbor n, and $\mathcal{N}_j$ are the set of neighbors to which subproblem j is coupled through the consistency constraints $\mathbf{c}^{\mathrm{y}}_{jn}$.

Similarly, separate copies $\mathbf{r}_{jm}$, $m \in \mathcal{R}_j$, for the coupling responses $\mathbf{r}_j$, $j = 0, 1, \ldots, M$ are introduced at the subproblems $m \in \mathcal{R}_j$ that require response $\mathbf{r}_j$ as an input. These response variables are then added to the set of optimization variables. To enforce consistency between response $\mathbf{r}_j$ and each of its copies $\mathbf{r}_{jm}$, consistency constraints are introduced

$$\mathbf{c}^{\mathrm{r}}_{jm} = T_{jm}\mathbf{r}_j - \mathbf{r}_{jm} = \mathbf{0} \qquad j = 0, 1, \ldots, M, \qquad m \in \mathcal{R}_j \tag{2.4}$$

where the binary selection matrix T_{jm} selects those responses of subproblem j that are relevant to its neighbor m.

After introduction of the variable copies and the consistency constraints, the partitioned problem can be written as:

$$\begin{aligned}
\min_{\overline{\mathbf{x}}_0, \overline{\mathbf{x}}_1, \ldots, \overline{\mathbf{x}}_M} \quad & [\mathbf{f}_0(\overline{\mathbf{x}}_0, \overline{\mathbf{x}}_1, \ldots, \overline{\mathbf{x}}_M), \mathbf{f}_1(\overline{\mathbf{x}}_1), \ldots, \mathbf{f}_M(\overline{\mathbf{x}}_M)] \\
\text{subject to} \quad & \mathbf{g}_0(\overline{\mathbf{x}}_0, \overline{\mathbf{x}}_1, \ldots, \overline{\mathbf{x}}_M) \leq \mathbf{0} \\
& \mathbf{h}_0(\overline{\mathbf{x}}_0, \overline{\mathbf{x}}_1, \ldots, \overline{\mathbf{x}}_M) = \mathbf{0} \\
& (\mathbf{r}_0, \mathbf{u}_0) = \mathbf{a}_0(\overline{\mathbf{x}}_0, \overline{\mathbf{x}}_1, \ldots, \overline{\mathbf{x}}_M) \\
& \mathbf{g}_j(\overline{\mathbf{x}}_j) \leq \mathbf{0} && j = 1, \ldots, M \\
& \mathbf{h}_j(\overline{\mathbf{x}}_j) = \mathbf{0} && j = 1, \ldots, M \\
& (\mathbf{r}_j, \mathbf{u}_j) = \mathbf{a}_j(\overline{\mathbf{x}}_j) && j = 1, \ldots, M \\
& \mathbf{c}(\overline{\mathbf{x}}_0, \overline{\mathbf{x}}_1, \ldots, \overline{\mathbf{x}}_M) = \mathbf{0} \\
\text{where} \quad & \overline{\mathbf{x}}_0 = [\mathbf{r}_0, \mathbf{u}_0] \\
& \overline{\mathbf{x}}_j = [\mathbf{y}_j, \mathbf{x}_j, \mathbf{r}_j, \mathbf{u}_j, \mathbf{r}_{mj} | j \in \mathcal{R}_m] && j = 1, \ldots, M
\end{aligned} \tag{2.5}$$

where $\mathbf{c}$ denotes the collection of the consistency constraints (2.3)–(2.4), and the local responses $\mathbf{u}_0, \ldots, \mathbf{u}_M$ have been included as optimization variables.

The artificially introduced variables in the above problem can be eliminated from the optimization variables through the consistency constraints $\mathbf{c}$ or the analysis equations. Whether or not these variables are eliminated is however a matter of *coordination*, and not a choice we want to make at the *partitioning* stage. Hence, we will refer to problem (2.5) with all optimization variables included when we speak of the partitioned problem for the remainder of the user manual.

Chapter 3
Partition specification language Ψ

Ψ (PS – Partition Specification) is a linguistic approach for the intuitive specification of problem partitions. Before describing the proposed partition specification language Ψ in more detail, we first introduce the following main definitions:

- A *variable* is an optimization variable of the system design problem, and can be an actual design variable or a response variable computed as the output of an analysis.
- A *function* represents an analysis that takes variables as arguments, and computes responses based on the values of the variables.
- A *component* represents a computational subproblem in a partition, which contains a number of variables and functions.
- A *system* contains a collection of coupled sub-components whose coupled solution is guided by a coordination method. A sub-component can be a component or another system.

The Ψ language specifies a partitioned problem by defining how variables and functions are distributed over the components, and how these components are combined into larger subsystems and systems. The building blocks of a specification in Ψ are therefore components and systems. The specification of detailed information of variables and functions is beyond its purpose. It is assumed that such additional information is supplied in conjunction with the specification of the partitioned problem.

1 Components

The main building blocks of a partitioned problem definition in Ψ are components. These components are typically associated with analysis disciplines in aspect-based partitions or with components in object-based partitions, but may also be purely computational subproblems that have no direct relation to the physical system. At the partitioning stage, we are not concerned with

the assignment of analysis and/or optimization authorities to components. This is a choice that is made at the coordination stage, and does therefore not appear in the component definitions.

A component definition in Ψ is given by

```
comp N =
|[  extvar V_e
    intvar V_i
    objfunc F_o
    confunc F_c
    resfunc F_r
]|
```

The keyword comp is followed by the name *N* of the component definition. The name of a component must be unique with respect to the names of other components and systems. The body of the component definition is placed between the brackets |[and]|, and includes the definition of variables *V* and functions *F*.

The variables for a component define the elements of the vector $\overline{\mathbf{x}}_j$ in Eq. (2.5). The language distinguishes between two types of variables: external variables defined after the keyword extvar, and internal variables defined after the keyword intvar. External variables V_e can be accessed by the system the component is part of. These external variables are used in variable couplings or system-wide functions. Internal variables V_i are only accessible within the component. A variable's name must be unique with respect to the other variable names included in the component definition. The variable names do not have to be unique with respect to the variable names of other component definitions.

It should be noted that the external variables of a component may include more variables than just the coupling variables $\mathbf{y}_j$ and response variables $\mathbf{r}_j$ and $\mathbf{r}_{mj}$ (for which $j \in \mathcal{R}_m$) in Eq. (2.5). The local variables $\mathbf{x}_j$ and local responses $\mathbf{u}_j$ that are used as arguments in one of the system-wide functions $\mathbf{f}_0$, $\mathbf{g}_0$, $\mathbf{h}_0$, and $\mathbf{a}_0$ are external variables as well. The reason for taking a different division of variables is that from a system designer's viewpoint it is relevant to know which variables have an influence beyond the component in which they are defined. From this perspective, external variables are those variables that affect other components.

Functions *F* are divided into three groups: objective functions defined after the keyword objfunc, constraint functions defined after the keyword confunc, and response functions defined after the keyword resfunc. The objective functions of a component represent the local objectives $\mathbf{f}_j$ in Eq. (2.5), the constraint functions represent the local constraints $\mathbf{g}_j, \mathbf{h}_j$, and response functions express the relations between variables through the local analysis functions $\mathbf{a}_j$. Each objective and constraint function is defined as *f*(*X*), where *f* refers to its name, and *X* are its arguments. The arguments of a function are a list of one or more variables whose values are required to compute the value of the function. Response functions are defined as (*R*)=*f*(*X*), where *R* is a list of variables that are computed as responses of function *f*. The parenthesis around *R* are optional, and we apply the convention that they are only used when *R* is a list of more than one output variables (for example $r_1 = a_1(x)$ and $(r_1, r_2) = a_{12}(x)$). The arguments and responses of a response function may not share common elements. Coupled response functions of the form $r_1 = a_1(x, r_2)$, $r_2 = a_2(x, r_1)$ have to be included as separate response functions, instead of being composed into a single one of the form $(r_1, r_2) = a_{12}(x, r_1, r_2)$. Such a coupled form implies that the function a must include a description of how the coupling between the analysis functions a_1 and a_2 is resolved. Since this is a actually a *coordination* decision, we do not want to make this choice at the *partitioning* stage. It is possible to apply the same function multiple times with different arguments.

To illustrate the use of Ψ, consider the following example optimization problem:

$$\begin{array}{ll} \min\limits_{x_1,x_2,x_3,x_4,y,r,u} & [f_0(x_2,x_3), f_1(x_1,x_2,y,r), f_2(x_3,x_4,y)] \\ \text{subject to} & g_1(x_1,y) \leq 0 \\ & g_2(x_3,x_4,y,u) \leq 0 \\ & r = a_1(x_2,y) \\ & u = a_2(x_3,x_4,r) \end{array} \tag{3.1}$$

A possible partition of this problem consists of two subproblems. The first subproblem has local variables x_1, x_2 and local functions f_1, g_1, a_1. The second subproblem has local variables x_3, x_4, u and local functions f_2, g_2, a_2. The two components are coupled through the coupling variables y, r and the system-wide objective f_0 that depends on variable x_2 of the first subproblem and on x_3 of the second.

For this partitioned problem, the specification of the two subproblems is given below. The first subproblem is called *First*. Component *First* has four variables x_1, x_2, y, r and three functions f_1, g_1, a_1. Variables y, r are external variables since they are coupling variables of the system. Variable x_2 is also an external variable since it is an argument of the system-wide objective f_0. Variable x_1 is internal since it appears only in the functions of component *First*. Function $f_1(x_1, x_2, y, r)$ is a local objective with four arguments x_1, x_2, y, r, and function $g_1(x_1, y)$ is a local constraint with two arguments x_1, y. Function $a_1(x_2, y)$ is a response that determines the values of r. The definition for the second subproblem follows along similar lines.

comp *First* =
|[extvar x_2, y, r
intvar x_1
objfunc $f_1(x_1, x_2, y, r)$
confunc $g_1(x_1, y)$
resfunc $r = a_1(x_2, y)$
]|

comp *Second* =
|[extvar x_3, y, r
intvar x_4, u
objfunc $f_2(x_3, x_4, y)$
confunc $g_2(x_3, x_4, y, u)$
resfunc $u = a_2(x_3, x_4, r)$
]|

2 Systems

Once the components of a partition are defined, they can be assembled into systems. A system definition includes two or more subcomponents and describes the couplings between them. The subcomponents of a system can be components or other systems. Components and systems together form the partitioned problem. A system definition in Ψ is given by

syst N =
|[extvar V_e
intvar V_i
sub C
link L
objfunc F_o

confunc F_c
resfunc F_r
alias *W*
]|

The keyword syst is followed by the name *N* of the system definition. The name of a system must be unique with respect to the names of other components and systems. The body of the system definition contains a definition of variables, sub-components, variable links, coupling functions, and so-called alias variables.

Since a system is a collection of components and their couplings, it does not have *design* variables of its own. The *response* variables $\mathbf{s}_0 = [\mathbf{r}_0, \mathbf{u}_0]$ associated with the coupling analysis functions $\mathbf{a}_0$ (see Eq. (2.5)) are however included in a system definition. The keywords extvar and intvar are used to define which response variables are external and which are internal.

The sub-components *C* are defined after the sub keyword by a comma-separated list of instantiations of the sub-components that comprise the system. Both components and systems can be instantiated as sub-components. These instantiations are of the form $S : D$, where D refers to the name of the component or system definition that is instantiated, and S is a comma-separated list of names, one for each instantiation of D that is included in the system.

Coupling functions *F* are divided into three groups: objective functions defined after objfunc, constraint functions defined after confunc, and response functions defined after resfunc. The objective functions of a system represent the coupling objectives $\mathbf{f}_0$ in Eq. (2.5), the constraint functions represent the coupling constraints $\mathbf{g}_0, \mathbf{h}_0$, and response functions express the relations between the response variables through the coupling analysis functions $\mathbf{a}_0$. Each objective and constraint function is defined as *f*(*X*), where *f* refers to its name, and *X* are its arguments. Response functions are defined as (*R*)=*f*(*X*), in which *R* is a list of variables that are computed as responses of function *f*. For systems, the arguments are a list of response variables and/or sub-component variables *S*.*v*, where the identifier *S*.*v* is used to access variable *v* of sub-component *S*. Each coupling function should have arguments from at least two different sub-components. This dependency may be directly visible in the arguments, but can also be implicit through the use of response variables and response functions. Again it is possible to use the same function multiple times with different arguments.

The variable couplings *L* that follow the link keyword define which variables are coupled. Essentially, each coupling represents a consistency constraint between two variables of different sub-components, or between a system response variable and a sub-component variable. A coupling is of the form I_1 -- I_2 where one of the I_1 and I_2 is always a variable of a sub-component, and the other is either a variable of another sub-component, or a system response variable. This latter case defines a consistency constraint $\mathbf{c}^{\mathrm{r}}_{0m}$ in Eq. (2.4). The collection of couplings gives the set of consistency constraints $\mathbf{c}$ of the partitioned problem (2.5). Observe that specifying consistency constraints using the proposed symbolic expression is far more intuitive than using the collections of binary selection matrices S_{jn} and T_{jm} and index sets $\mathcal{R}_j$ in Eqs. (2.3)–(2.4).

Additional syntax is available for specifying links that form star or chain structures. A star structure is used if one variable is coupled to many other variables. An example of a star link is $S_1.v_1$ -- $\{S_2.v_2, S_3.v_3, S_4.v_4\}$, which is equivalent to $S_1.v_1$ -- $S_2.v_2$, $S_1.v_1$ -- $S_3.v_3$, $S_1.v_1$ -- $S_4.v_4$, and couples variable v_1 of sub-component S_1 to variables v_2, v_3, and v_4, of components S_2, S_3, and S_4, respectively. An example of a chain link is $S_1.v_1$ -- $S_2.v_2$ -- $S_3.v_3$, which is equivalent to $S_1.v_1$ -- $S_2.v_2$, $S_2.v_2$ -- $S_3.v_3$.

Aliases *W* are used for systems that are themselves part of another system. An alias *w* is used to make a variable *v* of subcomponent *S* accessible outside the current system. Aliases refer

strictly to variables of subcomponents, and are externally accessible similar to a system's external variables. The difference between an alias and an external system variable is that an alias is computed at the subcomponent level, whereas the external variable is computed at the system level.

An alias *w* in system *N* is simply defined as *w = S.v* and introduces alias *w* for variable *v* of subcomponent *S*. This alias can then be used in a higher-level system using the identifier *N.w*. An advantage of using aliases instead of an identifier such as *N.S.v* is that the higher-level systems do not need to have detailed knowledge of the structure of its subsystems. Additionally, the definition of the higher level system does not need to be changed if the structure of the subsystem is modified. Observe that an alias definition does *not* define a consistency constraint; aliases are simply used to forward variable values from lower to higher levels in the problem hierarchy.

The final ingredient of a partitioned problem specification is the statement

topsyst *S*

which instantiates the partitioned problem by defining that the highest system in the coordination process is *S*.

The system definition for Example (3.1) is given below. The system has name *Problem*, and includes two sub-components A of type *First*, and B of type *Second*. One link $A.y$ -- $B.y$ couples variable y of A with variable y of B, and the second link $A.r$ -- $B.r$ couples r of A and B. Function $f_0(A.x_2, B.x_3)$ is included as a coupling objective that takes variable x_2 of A and x_3 of B as arguments. Finally, system *Problem* is defined as the highest-level system by using the topsyst statement.

```
syst Problem =
|[  sub A: First, B: Second
    link A.y -- B.y, A.r -- B.r
    objfunc f0(A.x2, B.x3)
]|

topsyst Problem
```

The definitions for components *First* and *Second*, system *Problem*, and the topsyst statement comprise the partition specification for our example problem. The resulting partition structure is depicted in Figure 3.1. For a complete description of the allowed syntax of Ψ, the reader is referred to Appendix A.

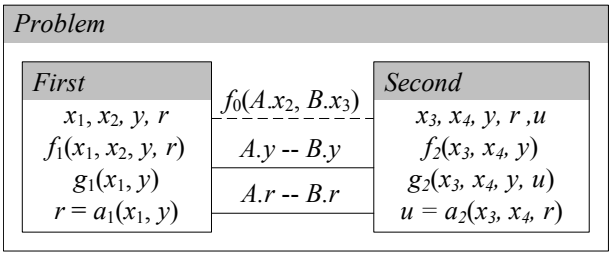


Figure 3.1: Illustration of specified partition for Problem (3.1).

Chapter 4
Processing of specifications

We have developed a compiler that checks partition specifications in Ψ for errors, and translates them into a normalized partition in INI format. These normalized definitions are less compact and harder to read than specifications in Ψ, but have the advantage that they can be easily interpreted by other programs [3, 23]. Additional, a generator has been developed that constructs a functional dependence table (FDT) from a normalized partition. Other representations such as graph sets or adjacency matrices can easily be derived in a similar fashion.

The following sections describe the normalized partition and the generator for the FDT in more detail. A description of the requirements that are asserted by the error checker is given in Appendix B.

1 Normalized partition

The normalized partition definition is defined in the INI format, and contains sections for each *instantiated* variable, function, component, coupling, and system. This in contrast to the partition specification in Ψ that includes *definitions* that can be instantiated multiple times. For example, a specification in Ψ can include a single component definition that is instantiated multiple times (for example the *Beam* components for the portal frame). In the normalized definition however, separate component sections for each instance are included (a separate section for each of the three beams). As a consequence, the normalized partition is defined such that it relates directly to the mathematical definitions associated with the partitioned problem of Eq. (2.5). Input files for computational frameworks are often based on these mathematical definitions, and we expect that automatic generation of these input files from the normalized definition is relatively easy.

Files in the INI format are composed of a number of sections, where each section contains a number of keys and associated values. The heading of such a section is placed between square brackets [heading], and after each heading, the key-value pairs of the section are listed in the form key = value. Here, a value can also be a comma-separated list of multiple values that is possibly empty.

Variable section var_i is introduced for each instantiated variable, where i is a counter. The key-value pairs in this section define the variable's name *N*, the definition in which it is specified *D*, the instantiation path *P* of this definition. The generic variable section is given below at the left, together with the section for variable x_2 of component A : *First* of Example (3.1) at the right. The specification for this example is given in Section 1.

[var_i]		[var_1]	
name	= *N*	name	= x_2
defined_in	= *D*	defined_in	= *First*
path	= *P*	path	= *Problem.A*

Function sections func_k are introduced for each instantiated function $k = 1, \ldots, m_\mathrm{f} + m_\mathrm{g} + m_\mathrm{h} + m_\mathrm{a}$. Function sections are similar to variable sections, but include additional keys argvars and resvars that define the arguments *X* and responses *R* of a function, respectively (where only analysis functions have responses). These arguments and responses are defined as comma-separated lists of appropriate variable section headings. Below, the generic function section is given, together with the section associated with function a_1 of Example (3.1).

[func_k]		[func_3]	
name	= *N*	name	= a_1
defined_in	= *D*	defined_in	= *First*
path	= *P*	path	= *Problem.A*
resvars	= *R*	resvars	= var_3
argvars	= *A*	argvars	= var_1,var_2

Component sections comp_j are introduced for each instantiated component $j = 1, \ldots, M$. A component section includes definitions of its instantiation name (or label) *N*, its definition type *T*, its instantiation path *P*, its coupling and local variables V_c and V_l, its coupling and local responses R_c and R_l, and its local objective, constraint, and response functions F_o, F_c ,and F_r. Note that the local and coupling variables correspond to the definitions of $\mathbf{x}_j$ and $\mathbf{y}_j$ in the partitioned problem (2.5), and that the local and coupling responses correspond to the definitions of $\mathbf{u}_j$, and $\mathbf{r}_j$ and $\mathbf{r}_{mj} | j \in \mathcal{R}_m$. Below, the general component section is given, together with the section associated with sub-component A of Example (3.1).

[comp_j]		[comp_1]	
name	= *N*	name	= *A*
type	= *T*	type	= *First*
path	= *P*	path	= *Problem.A*
coupling_vars	= V_c	coupling_vars	= var_2
local_vars	= V_l	local_vars	= var_1,var_4
coupling_resvars	= R_c	coupling_resvars	= var_3
local_resvars	= R_l	local_resvars	=
objfuncs	= F_o	objfuncs	= func_1
confuncs	= F_c	confuncs	= func_2
resfuncs	= F_r	resfuncs	= func_3

For each consistency coupling, a section link_p is introduced that includes the variables V_1,V_2 that it couples, the name *D* of the system in which it is defined, and the instantiation path of this system *P*. The generic coupling section is given below, together with the section associated with link $A.y$ -- $B.y$ of Example (3.1).

[link_p]		[link_1]	
defined_in	= D	defined_in	= *Problem*
path	= P	path	= *Problem*
coupling	= V_1, V_2	coupling	= var_2,var_6

The system sections syst_s for each system $s = 1, \ldots, S$ includes its name *N*, its type *T*, its instantiation path *P*, and its sub-components *S*. Furthermore, a system section includes a number of links *L*, its objective, constraint, and analysis functions F_{o}, F_{c}, and F_{r}. The function names are again section headings. The generic system section is given below, together with the section for system *Problem* of Example (3.1).

[syst_s]		[syst_1]	
name	= N	name	= *Problem*
type	= T	type	= *Problem*
path	= P	path	= *Problem*
coupling_resvars	= R_{c}	coupling_resvars	=
local_resvars	= R_{l}	local_resvars	=
sub_comps	= S	sub_comps	= comp_1,comp_2
links	= L	links	= link_1,link_2
objfuncs	= F_{o}	objfuncs	= func_7
confuncs	= F_{c}	confuncs	=
resfuncs	= F_{r}	resfuncs	=

Finally, a single topsyst section is included to define which system *S* is the highest in the hierarchy. The generic topsyst section is given below, together with the topsyst section for Example (3.1).

[topsyst]		[topsyst]	
system	= S	system	= syst_1

2 FDT generator

At this point, we have implemented a second generator that computes the functional dependence table (FDT) from a normalized partition. The FDT is a common representation of problem structure that is used in the context of automated partitioning methods. The FDT is a binary matrix in which rows correspond to the objective and constraint functions of a problem, and the columns correspond to optimization variables. Element (i, j) of the FDT is 1 if function i depends on variable j, and 0 otherwise. Essentially, the FDT is a binary representation of the Jacobian matrix of the problem, and similar representations appear in the large-scale optimization literature.

In its default setting, the FDT of the partitioned problem (2.5) is generated, where each consistency constraint is included as a separate row of the FDT. The generated FDT for Example (3.1) is depicted in Figure 4.1(a) and clearly reflects the structure of components and systems. Alternatively, the generator can generate an FDT where each consistency constraints $\mathbf{c}$ is used to eliminate an artificially introduced variable. The reduced FDT for Example (3.1) is depicted in Figure 4.1(b). Where the former "full" FDT reflects the structure of Problem (2.5) and the specification in Ψ, the latter "reduced" FDT reflects the structure of Problem (2.2) and is a representation often used by automated partitioning approaches.

		Problem.A				*Problem.B*				
		x_2	y	r	x_1	x_3	y	r	x_4	u
syst	f_0	1	0	0	0	1	0	0	0	0
Problem	c^{y}_{12}	0	1	0	0	0	1	0	0	0
	c^{r}_{12}	0	0	1	0	0	0	1	0	0
comp	f_1	1	1	1	1	0	0	0	0	0
Problem.A	g_1	0	1	0	1	0	0	0	0	0
	a_1	1	1	1	0	0	0	0	0	0
comp	f_2	0	0	0	0	1	1	0	1	0
Problem.B	g_2	0	0	0	0	1	1	0	1	1
	a_2	0	0	0	0	1	0	1	1	1

(a) Full, associated with formulation (2.5)

		$\mathbf{y}$	$\mathbf{r}_1$	$\mathbf{x}_1$		$\mathbf{x}_2$		$\mathbf{u}_2$
		y	r	x_2	x_1	x_3	x_4	u
$\mathbf{f}_0$	f_0	0	0	1	0	1	0	0
$\mathbf{f}_1$	f_1	1	1	1	1	0	0	0
$\mathbf{g}_1$	g_1	1	0	0	1	0	0	0
$\mathbf{a}_1$	a_1	1	1	1	0	0	0	0
$\mathbf{f}_2$	f_2	1	0	0	0	1	1	0
$\mathbf{g}_2$	g_2	1	0	0	0	1	1	1
$\mathbf{a}_2$	a_2	0	1	0	0	1	1	1

(b) Reduced, associated with formulation (2.2)

Figure 4.1: Functional dependence tables generated from normalized partition of Example (3.1).

Chapter 5
Examples

In this section, we demonstrate the use of the partition specification language on a number of example problems from the literature. The first example is a geometric programming problems, the second example is Golinski's speed reducer design problem [7], the third example is the portal frame design problem introduced by [18], the fourth is a vehicle chassis design problem of [10], and the fifth example is the supersonic business jet example introduced by [1]. The partition specification files, generated normalized partitions, and generated functional dependence tables for these examples are included in Appendix C.

1 Geometric programming

The first example is the geometric programming problem taken from [9, 19, 20]. The problem is given by:

$$
\begin{aligned}
\min_{z_1,\ldots,z_{14}} \quad & f(z_1) + f(z_2) = z_1^2 + z_2^2 \\
\text{subject to} \quad & g_1(z_3, z_4, z_5) = (z_3^{-2} + z_4^2) z_5^{-2} - 1 \leq 0 \\
& g_2(z_5, z_6, z_7) = (z_5^2 + z_6^{-2}) z_7^{-2} - 1 \leq 0 \\
& g_3(z_8, z_9, z_{11}) = (z_8^2 + z_9^2) z_{11}^{-2} - 1 \leq 0 \\
& g_4(z_8, z_{10}, z_{11}) = (z_8^{-2} + z_{10}^2) z_{11}^{-2} - 1 \leq 0 \\
& g_5(z_{11}, z_{12}, z_{13}) = (z_{11}^2 + z_{12}^{-2}) z_{13}^{-2} - 1 \leq 0 \\
& g_6(z_{11}, z_{12}, z_{14}) = (z_{11}^2 + z_{12}^2) z_{14}^{-2} - 1 \leq 0 \\
& h_1(z_1, z_3, z_4, z_5) = (z_3^2 + z_4^{-2} + z_5^2) z_1^{-2} - 1 = 0 \\
& h_2(z_2, z_5, z_6, z_7) = (z_5^2 + z_6^2 + z_7^2) z_2^{-2} - 1 = 0 \\
& h_3(z_3, z_8, z_9, z_{10}, z_{11}) = (z_8^2 + z_9^{-2} + z_{10}^{-2} + z_{11}^2) z_3^{-2} - 1 = 0 \\
& h_4(z_6, z_{11}, z_{12}, z_{13}, z_{14}) = (z_{11}^2 + z_{12}^2 + z_{13}^2 + z_{14}^2) z_6^{-2} - 1 = 0
\end{aligned}
\tag{5.1}
$$

The first partition of this example defines two subproblems. The first subproblem has local variables $\mathbf{x}_1 = [z_1, z_2, z_4, z_5, z_7]$, local objective $f(z_1) + f(z_2)$, and local constraints g_1, g_2, h_1, h_2. The second subproblem has local variables $\mathbf{x}_2 = [z_8, z_9, z_{10}, z_{11}, z_{12}, z_{13}, z_{14}]$, no local objective, and local constraints $g_3, g_4, g_5, g_6, h_3, h_4$. The subproblems are coupled through the linking variables $\mathbf{y} = [z_3, z_6]$. The specification for this partition is given below:

comp *First* =
|[extvar z_3, z_6
intvar z_1, z_2, z_4, z_5, z_7
objfunc $f(z_1)$, $f(z_2)$
confunc $g_1(z_3, z_4, z_5)$
, $g_2(z_5, z_6, z_7)$
, $h_1(z_1, z_3, z_4, z_5)$
, $h_2(z_2, z_5, z_6, z_7)$
]|

comp *Second* =
|[extvar z_3, z_6
intvar $z_8, z_9, z_{10}, z_{11}, z_{12}, z_{13}, z_{14}$
confunc $g_3(z_8, z_9, z_{11})$
, $g_4(z_8, z_{10}, z_{11})$
, $g_5(z_{11}, z_{12}, z_{13})$
, $g_6(z_{11}, z_{12}, z_{14})$
, $h_3(z_3, z_8, z_9, z_{10}, z_{11})$
, $h_4(z_6, z_{11}, z_{12}, z_{13}, z_{14})$
]|

syst *Geo2a* =
|[sub A: *First*, B: *Second*
link $A.z_3$ -- $B.z_3$, $A.z_6$ -- $B.z_6$
]|

topsyst *Geo2a*

The coupling between the two subproblems through the variables z_3, z_6 can easily observed from the system definition *Geo2a*, as well as the local nature of the remaining variables and functions in the definitions of *First* and *Second*.

A second partition of the above splits the second subproblem into two components *Second*$_1$ and *Second*$_2$ that are coupled through the variable z_{11}. The definition of component *First* of the previous partition remains the same, and the specifications of the two new components and the system *Geo2b* for this partition are given by

comp *Second*$_1$ =
|[extvar z_3, z_{11}
intvar z_8, z_9, z_{10}
confunc $g_3(z_8, z_9, z_{11})$
, $g_4(z_8, z_{10}, z_{11})$
, $h_3(z_3, z_8, z_9, z_{10}, z_{11})$
]|

comp *Second*$_2$ =
|[extvar z_6, z_{11}
intvar z_{12}, z_{13}, z_{14}
confunc $g_5(z_{11}, z_{12}, z_{13})$
, $g_6(z_{11}, z_{12}, z_{14})$
, $h_4(z_6, z_{11}, z_{12}, z_{13}, z_{14})$
]|

syst *Geo2b* =
|[sub A: *First*, B_1: *Second*$_1$, B_2: *Second*$_2$
link $A.z_3$ -- $B_1.z_3$, $A.z_6$ -- $B_2.z_6$, $B_1.z_{11}$ -- $B_2.z_{11}$
]|

topsyst *Geo2b*

The third partition for this example splits component *First* of the second partition into two components *First*$_1$ and *First*$_2$ that are coupled through variable z_5. The specifications of components *Second*$_1$ and *Second*$_2$ remain the same as in the previous partition, and the specifications for components *First*$_1$ and *First*$_2$ and system *Geo2c* are given by:

comp *First*$_1$ =
|[extvar z_3, z_5
intvar z_1, z_4
objfunc $f(z_1)$
confunc $g_1(z_3, z_4, z_5)$
, $h_1(z_1, z_3, z_4, z_5)$
]|

comp *First*$_2$ =
|[extvar z_5, z_6
intvar z_2, z_7
objfunc $f(z_2)$
confunc $g_2(z_5, z_6, z_7)$
, $h_2(z_2, z_5, z_6, z_7)$
]|

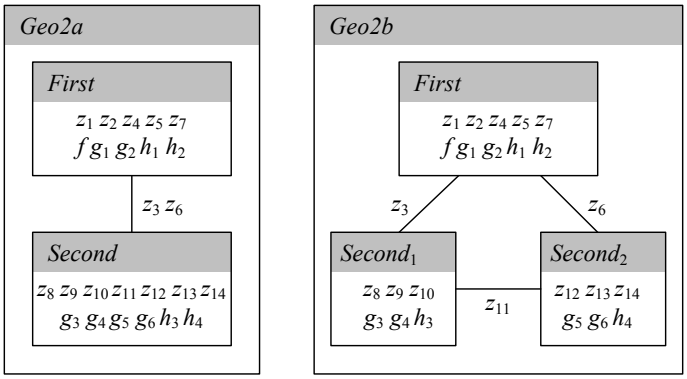
syst *Geo2c* =
|[sub A_1: *First*$_1$, A_2: *First*$_2$, B_1: *Second*$_1$, B_2: *Second*$_2$
link $A_1.z_3$ -- $B_1.z_3$, $A_1.z_5$ -- $A_2.z_5$
, $A_2.z_6$ -- $B_2.z_6$, $B_1.z_{11}$ -- $B_2.z_{11}$
]|

topsyst *Geo2c*

The three partitions of this example are depicted in Figure 5.1.

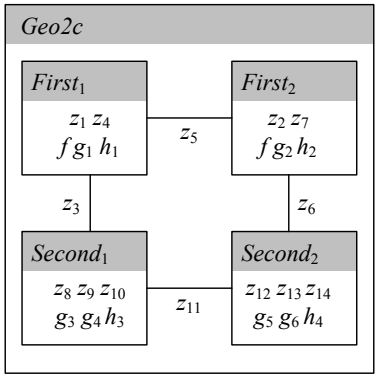
2 Speed reducer

The second example is the speed reducer design problem taken from [7, 15, 20]. The objective of this problem is to minimize the volume of a speed reducer (Fig. 5.2), subjected to stress, deflection, and geometric constraints. The design variables are the dimensions of the gear itself (x_1, x_2, x_3), and both the shafts (x_4, x_6 and x_5, x_7). The design problem for the speed reducer is



(a) First partition

(b) Second partition



(c) Third partition

Figure 5.1: Three partitions of geometric programming problem.

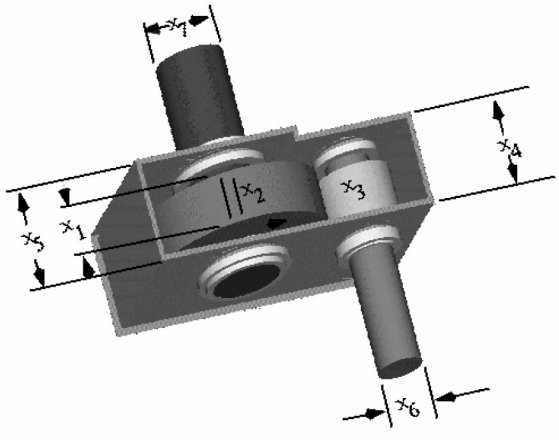


Figure 5.2: Schematic of the speed reducer. Shaft A is the input shaft on the right (with diameter x_6), and shaft B is the output shaft on the left.

defined by:

$$
\begin{aligned}
\min_{x_1,\ldots,x_7} \quad & [F_1(x_1,x_2,x_3), F_2(x_1,x_6), F_3(x_1,x_7), F_4(x_6), F_5(x_7), F_6(x_4,x_6), F_7(x_5,x_7)] \\
\text{subject to} \quad & g_1(x_2,x_3,x_4,x_6) = \tfrac{1}{110x_6^3}\sqrt{\left(\tfrac{745x_4}{x_2x_3}\right)^2 + 1.69\cdot 10^7} - 1 \le 0 \\
& g_2(x_2,x_3,x_5,x_7) = \tfrac{1}{85x_7^3}\sqrt{\left(\tfrac{745x_5}{x_2x_3}\right)^2 + 1.575\cdot 10^8} - 1 \le 0 \\
& g_3(x_4,x_6) = \tfrac{1.5x_6+1.9}{x_4} - 1 \le 0 \\
& g_4(x_5,x_7) = \tfrac{1.1x_7+1.9}{x_5} - 1 \le 0 \\
& g_5(x_1,x_2,x_3) = \tfrac{27}{x_1x_2^2x_3} - 1 \le 0 \\
& g_6(x_1,x_2,x_3) = \tfrac{397.5}{x_1x_2^2x_3^2} - 1 \le 0 \\
& g_7(x_2,x_3,x_4,x_6) = \tfrac{1.93x_4^3}{x_2x_3x_6^4} - 1 \le 0 \\
& g_8(x_2,x_3,x_5,x_7) = \tfrac{1.93x_5^3}{x_2x_3x_7^4} - 1 \le 0 \\
& g_9(x_2,x_3) = \tfrac{x_2x_3}{40} - 1 \le 0 \\
& g_{10}(x_1,x_2) = \tfrac{5x_2}{x_1} - 1 \le 0 \\
& g_{11}(x_1,x_2) = \tfrac{x_1}{12x_2} - 1 \le 0 \\
\text{where} \quad & F_1 = 0.7854x_1x_2^2(3.3333x_3^2 + 14.9335x_3 - 43.0934) \\
& F_2 = -1.5079x_1x_6^2,\ F_3 = -1.5079x_1x_7^2,\ F_4 = 7.477x_6^3 \\
& F_5 = 7.477x_7^3,\ F_6 = 0.7854x_4x_6^2,\ F_7 = 0.7854x_5x_7^2
\end{aligned}
\tag{5.2}
$$

The first partition of this problem has three subproblems. Subproblem 1 (*Gear*) is concerned with designing the gear, and subproblems 2 and 3 are associated with the design of shafts 1 and 2, respectively (*ShaftA* and *ShaftB*). The local objective of the gear subproblem *Gear* is $f_1 = [F_1]$, and its local constraints are $\mathbf{g}_1 = [g_5, g_6, g_9, g_{10}, g_{11}]$. The local objective of *ShaftA* is $f_2 = [F_2, F_4, F_6]$, and the local constraints are $\mathbf{g}_2 = [g_1, g_3, g_7]$. For *ShaftB*, the local objective is given by $f_3 = [F_3, F_5, F_7]$, and the local constraints are $\mathbf{g}_3 = [g_2, g_4, g_8]$. For this partition, the vector of shared variables is $\mathbf{y} = [x_1, x_2, x_3]$, which are the design variables associated with the gear. Subproblem *Gear* has no local design variables $\mathbf{x}_1 = []$. The local variables for *ShaftA* are the length and diameter of shaft 1 $\mathbf{x}_2 = [x_4, x_6]$. Similarly, the local variables for *ShaftB* are the length and diameter of shaft 2 $\mathbf{x}_2 = [x_5, x_7]$. The specification for this partition is given by

comp *Gear* =
|[extvar x_1, x_2, x_3
  objfunc $F_1(x_1, x_2, x_3)$
  confunc $g_5(x_1, x_2, x_3)$
    , $g_6(x_1, x_2, x_3)$
    , $g_9(x_2, x_3)$
    , $g_{10}(x_1, x_2)$
    , $g_{11}(x_1, x_2)$
]|

comp *ShaftA* =
|[extvar x_1, x_2, x_3
  intvar x_4, x_6
  objfunc $F_2(x_1, x_6)$
    , $F_4(x_6)$
    , $F_6(x_4, x_6)$
  confunc $g_1(x_2, x_3, x_4, x_6)$
    , $g_3(x_4, x_6)$
    , $g_7(x_2, x_3, x_4, x_6)$
]|

comp *ShaftB* =
|[extvar x_1, x_2, x_3
intvar x_5, x_7
objfunc $F_3(x_1, x_7)$
, $F_5(x_7)$
, $F_7(x_5, x_7)$
confunc $g_2(x_2, x_3, x_5, x_7)$
, $g_4(x_5, x_7)$
, $g_8(x_2, x_3, x_5, x_7)$
]|

syst *SpeedReducer* =
|[sub G: *Gear*, S_1: *ShaftA*, S_2: *ShaftB*
link $G.x_1$ -- $\{S_1.x_1, S_2.x_1\}$
, $G.x_2$ -- $\{S_1.x_2, S_2.x_2\}$
, $G.x_3$ -- $\{S_1.x_3, S_2.x_3\}$
]|

topsyst *SpeedReducer*

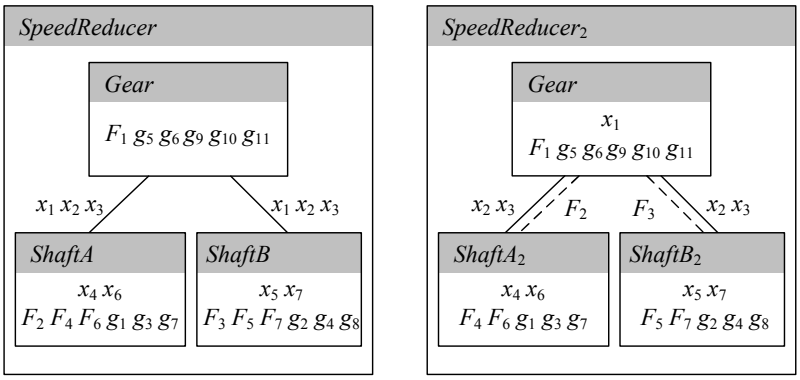
An alternative partition is to select F_2 and F_3 as coupling objectives such that variable x_1 is no longer a coupling variable, but becomes a local to the first subproblem $\mathbf{x}_1 = [x_1]$. Note that x_1 is still an external variable of *Gear*, since it is an argument of the coupling objectives F_2 and F_3. The *Gear* subproblem of the previous partitions can be used, and the specification of the alternative shaft subproblems *ShaftA*$_2$ and *ShaftB*$_2$ and system *SpeedReducer*$_2$ are given below.

comp *ShaftA*$_2$ =
|[extvar x_2, x_3, x_6
intvar x_4
objfunc $F_4(x_6)$
, $F_6(x_4, x_6)$
confunc $g_1(x_2, x_3, x_4, x_6)$
, $g_3(x_4, x_6)$
, $g_7(x_2, x_3, x_4, x_6)$
]|

comp *ShaftB*$_2$ =
|[extvar x_2, x_3, x_7
intvar x_5
objfunc $F_5(x_7)$
, $F_7(x_5, x_7)$
confunc $g_2(x_2, x_3, x_5, x_7)$
, $g_4(x_5, x_7)$
, $g_8(x_2, x_3, x_5, x_7)$
]|

syst *SpeedReducer*$_2$ =
|[sub G: *Gear*, S_1: *ShaftA*$_2$, S_2: *ShaftB*$_2$
objfunc $F_2(x_1, x_6)$, $F_3(x_1, x_7)$
link $G.x_2$ -- $\{S_1.x_2, S_2.x_2\}$
, $G.x_3$ -- $\{S_1.x_3, S_2.x_3\}$
]|

topsyst *SpeedReducer*$_2$



(a) First partition

(b) Second partition

Figure 5.3: Two partitions for the speed reducer problem.

The two partitions for the speed reducer problem are illustrated in Figure 5.3.

3 Portal frame

The third example concerns the structural optimization of a portal frame subjected to an external load. The portal frame consists of three I-beams $i = 1, 2, 3$, each with six cross-sectional dimensions $\mathbf{z}^i = [h, w_1, w_2, b, t_1, t_2]^i$ as design variables (see Figure 5.4). The objective of the structural optimization problem is to minimize the total mass of the frame m. Constraints are posed on maximal deflection of the loaded node (g_d), a number of stresses in the beams ($\mathbf{g}_\text{b}^i$, $i = 1, 2, 3$), and a number of cross-sectional constraints that limit the aspect ratios of the flanges ($\mathbf{g}_\text{c}^i$, $i = 1, 2, 3$).

The evaluation of the design constraints requires a number of responses computed with a set of analysis functions. First, the area A^i and moment of inertia I^i of each beam is computed from the detailed cross-sectional dimensions $\mathbf{z}^i$ through the analysis function $(A^i, I^i) = \mathbf{a}_\text{c}(\mathbf{z}_i)$. These responses are required to perform the structural analysis of the portal frame that determines the reaction forces $\mathbf{F}^i = [X_1, Y_1, M_1, X_2, Y_2, M_2]^i$ in each beam, and the deflection u of the loaded node: $(\mathbf{F}^1, \mathbf{F}^2, \mathbf{F}^3, u) = \mathbf{a}_\text{f}(A^1, A^2, A^3, I^1, I^2, I^3)$. At each beam, the stresses $\boldsymbol{\sigma}$ are computed from the reaction forces $\mathbf{F}^i$ and the detailed dimensions of the beam $\mathbf{z}^i$ through the analysis relation $\boldsymbol{\sigma}^i = \mathbf{a}_\text{b}(\mathbf{F}^i, \mathbf{z}^i)$.

The portal frame optimization problem is defined by

$$
\begin{array}{lll}
\text{find} & \mathbf{z}^1, \mathbf{z}^2, \mathbf{z}^3, A^1, A^2, A^3, I^1, I^2, I^3, & \\
& u, \mathbf{F}^1, \mathbf{F}^2, \mathbf{F}^3, \boldsymbol{\sigma}^1, \boldsymbol{\sigma}^2, \boldsymbol{\sigma}^3 & \\
\min & m(A^1, A^2, A^3) & \\
\text{s.t.} & g_\text{d}(u) \leq 0 & \\
& \mathbf{g}_\text{b}^i(\boldsymbol{\sigma}^i) \leq \mathbf{0} & i = 1, 2, 3 \\
& \mathbf{g}_\text{c}^i(\mathbf{z}^i) \leq \mathbf{0} & i = 1, 2, 3 \\
& \boldsymbol{\sigma}^i = \mathbf{a}_\text{b}(\mathbf{F}^i, \mathbf{z}^i) & i = 1, 2, 3 \\
& (A^i, I^i) = \mathbf{a}_\text{c}(\mathbf{z}^i) & i = 1, 2, 3 \\
& (\mathbf{F}^1, \mathbf{F}^2, \mathbf{F}^3, u) = \mathbf{a}_\text{f}(A^1, A^2, A^3, I^1, I^2, I^3) &
\end{array}
\tag{5.3}
$$

where both design and response variables are included as optimization variables. A possible

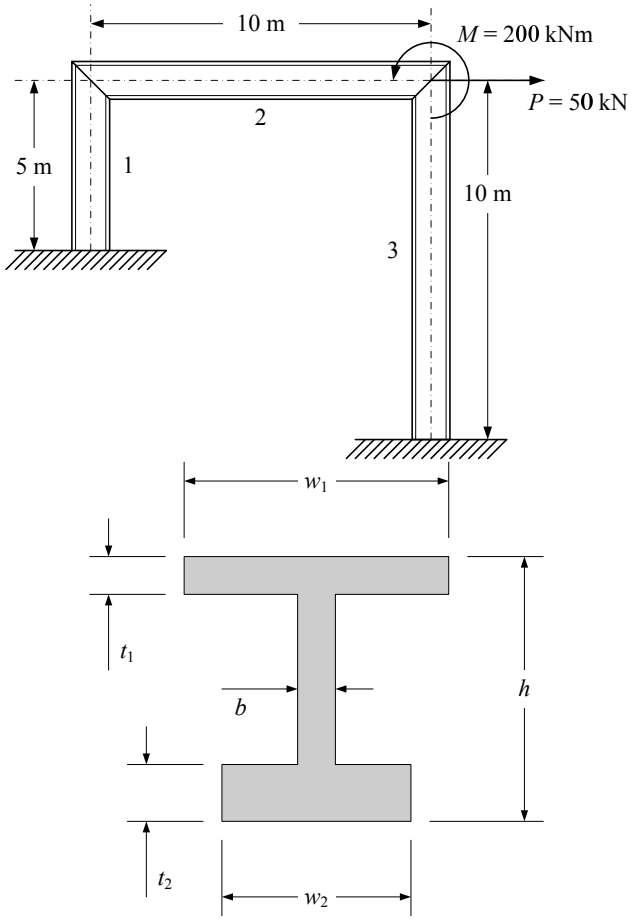


Figure 5.4: Illustration of the portal frame.

partition is to define one component for the portal frame and one component for each beam (see Figure 5.5(a)). The specification for this partition is given by

comp *Frame* =
|[extvar $A^1, A^2, A^3, I^1, I^2, I^3, \mathbf{F}^1, \mathbf{F}^2, \mathbf{F}^3$
intvar u
objfunc $m(A^1, A^2, A^3)$
confunc $g_{\mathrm{d}}(u)$
resfunc $(\mathbf{F}^1, \mathbf{F}^2, \mathbf{F}^3, u) = \mathbf{a}_{\mathrm{f}}(A^1, A^2, A^3, I^1, I^2, I^3)$
]|

comp *Beam* =
|[extvar $A, I, \mathbf{F}$
intvar $\mathbf{z}, \boldsymbol{\sigma}$
confunc $\mathbf{g}_{\mathrm{b}}(\boldsymbol{\sigma})$, $\mathbf{g}_{\mathrm{c}}(\mathbf{z})$
resfunc $\boldsymbol{\sigma} = \mathbf{a}_{\mathrm{b}}(\mathbf{F}, \mathbf{z})$
, $(A, I) = \mathbf{a}_{\mathrm{c}}(\mathbf{z})$
]|

syst *Portal* =
|[sub F: *Frame*, B_1, B_2, B_3: *Beam*
link $F.A^1$ -- $B_1.A$, $F.I^1$ -- $B_1.I$, $F.\mathbf{F}^1$ -- $B_1.\mathbf{F}$
, $F.A^2$ -- $B_2.A$, $F.I^2$ -- $B_2.I$, $F.\mathbf{F}^2$ -- $B_2.\mathbf{F}$
, $F.A^3$ -- $B_3.A$, $F.I^3$ -- $B_3.I$, $F.\mathbf{F}^3$ -- $B_3.\mathbf{F}$
]|

topsyst *Portal*

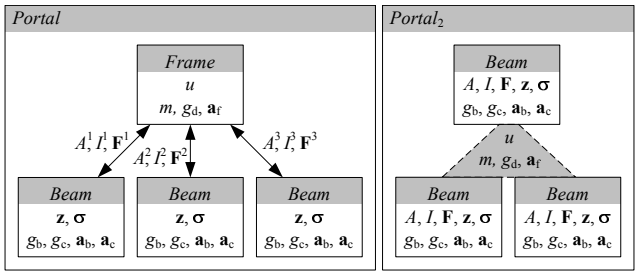
Component *Frame* includes the response variables A^i, I^i, $\mathbf{F}^i$, and u. Responses $\mathbf{F}^i$ and u are determined from the values of A^i and I^i with the analysis functions $\mathbf{a}_{\mathrm{f}}$. Since the response u is only required locally, this response is an internal variable. The component has the total mass m as a local objective function, and the deflection constraint g_{d} as a local constraint.

Since the beam components are very alike, one can use a single generic definition that is instantiated three times within the system. The generic beam definition of component *Beam* has external variables A, I, and $\mathbf{F}$, and internal variables $\mathbf{z}$ and $\boldsymbol{\sigma}$. The responses A, I, and $\boldsymbol{\sigma}$ are determined from the reactions forces $\mathbf{F}$ and the cross-section details $\mathbf{z}$ through the analysis functions $\mathbf{a}_{\mathrm{b}}$ and $\mathbf{a}_{\mathrm{c}}$. The local constraints are the stress constraints $\mathbf{g}_{\mathrm{b}}$ and the cross-sectional constraints $\mathbf{g}_{\mathrm{c}}$.

The system *Portal* includes one sub-component F of type *Frame*, and three sub-components B_1, B_2, and B_3 of type *Beam*. The external variables of each *Beam* sub-component are coupled to their counterparts in the *Frame* sub-component. The system does not have coupling functions. Finally, the *Portal* system is defined as the highest-level system through the topsyst statement.

An alternative partition can be set up that does not include component *Frame*, but defines coupling functions for m, g_{d}, and $\mathbf{a}_{\mathrm{f}}$ (see Figure 5.5(b)). The specification of this partition uses the same definition for component *Beam*, and defines an alternative system *Portal*$_2$ as

syst *Portal*$_2$ =
|[intvar $\mathbf{F}^1, \mathbf{F}^2, \mathbf{F}^3, u$
sub B_1, B_2, B_3: *Beam*
objfunc $m(B_1.A, B_2.A, B_3.A)$
confunc $g_{\mathrm{d}}(u)$



(a) Four components with coupling variables

(b) Three components with coupling functions

Figure 5.5: Two partitions for the portal frame example.

resfunc $(\mathbf{F}^1, \mathbf{F}^2, \mathbf{F}^3, u) =$
$\mathbf{a}_f(B_1.A, B_2.A, B_3.A, B_1.I, B_2.I, B_3.I)$
link $\mathbf{F}^1$ -- $B_1.\mathbf{F}$, $\mathbf{F}^2$ -- $B_2.\mathbf{F}$, $\mathbf{F}^3$ -- $B_3.\mathbf{F}$
]|

topsyst *Portal*$_2$

This alternative system *Portal*$_2$ has three *Beam* sub-components and includes a number of internal response variables for those responses computed by the coupling analysis function $\mathbf{a}_f$. These responses are then used as arguments of the coupling constraint g_d, or are linked to the external response variables $\mathbf{F}$ of the individual beams. Both partitions are easily and compactly specified using Ψ, which illustrates the expressivity of the language.

4 Vehicle chassis

The fourth example is a vehicle chassis design problem that aims at optimizing five handling and ride quality metrics while considering the design of front and rear suspensions, and vertical and cornering stiffness models. A detailed description of the problem can be found in [10]. The reader is referred to Table 5.1 for a brief description of the optimization variables. The chassis

Table 5.1: Description of the optimization variables for the vehicle chassis problem. Indices "f" refer to front and "r" to rear.

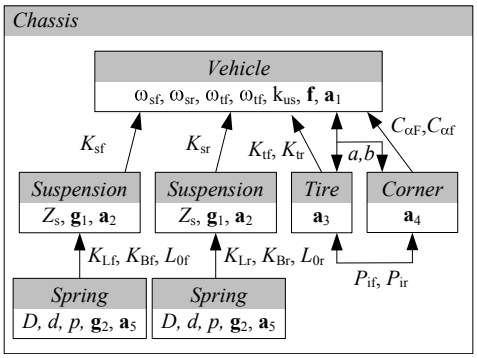
Design variables		Response variables	
a	tire position	ω_{sf}	spring nat. freq.
b	tire position	ω_{sr}	spring nat. freq.
P_{if}	tire pressure	ω_{tf}	tire nat. freq.
P_{ir}	tire pressure	ω_{tr}	tire nat. freq.
D_{f}	coil diameter	k_{us}	understeer gradient
D_{r}	coil diameter	K_{sf}	spring stiffness
d_{f}	wire diameter	K_{sr}	spring stiffness
d_{r}	wire diameter	K_{tf}	tire stiffness
p_{f}	pitch	K_{tr}	tire stiffness
p_{r}	pitch	$C_{\alpha\text{f}}$	cornering stiffness
Z_{sf}	suspension deflection	$C_{\alpha\text{r}}$	cornering stiffness
Z_{sr}	suspension deflection	K_{Lf}	linear stiffness
		K_{Lr}	linear stiffness
		K_{Bf}	bending stiffness
		K_{Br}	bending stiffness
		$L_{0\text{f}}$	free length
		$L_{0\text{r}}$	free length

design optimization problem is given by

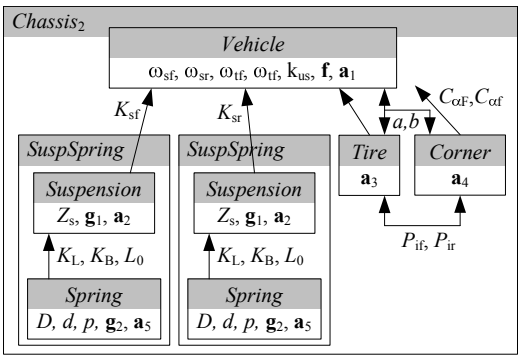
$$
\begin{array}{rl}
\text{find} & a, b, \omega_{\text{sf}}, \omega_{\text{sr}}, \omega_{\text{tf}}, \omega_{\text{tr}}, k_{\text{us}}, K_{\text{sf}}, K_{\text{sr}}, K_{\text{tf}}, K_{\text{tr}}, C_{\alpha\text{f}}, C_{\alpha\text{r}}, \\
& Z_{\text{sf}}, Z_{\text{sr}}, K_{\text{Lf}}, K_{\text{Lr}}, K_{\text{Bf}}, K_{\text{Br}}, L_{0\text{f}}, L_{0\text{r}}, P_{\text{if}}, P_{\text{ir}}, \\
& D_{\text{f}}, D_{\text{r}}, d_{\text{f}}, d_{\text{r}}, p_{\text{f}}, p_{\text{r}} \\
\min & \mathbf{f}(\omega_{\text{sf}}, \omega_{\text{sr}}, \omega_{\text{tf}}, \omega_{\text{tr}}, k_{\text{us}}) \\
\text{subject to} & \mathbf{g}_1(Z_{\text{sf}}, K_{\text{Lf}}, K_{\text{Bf}}, L_{0\text{f}}) \leq \mathbf{0} \\
& \mathbf{g}_1(Z_{\text{sr}}, K_{\text{Lr}}, K_{\text{Br}}, L_{0\text{r}}) \leq \mathbf{0} \\
& \mathbf{g}_2(D_{\text{f}}, d_{\text{f}}, p_{\text{f}}) \leq \mathbf{0} \\
& \mathbf{g}_2(D_{\text{r}}, d_{\text{r}}, p_{\text{r}}) \leq \mathbf{0} \\
& (\omega_{\text{sf}}, \omega_{\text{sr}}, \omega_{\text{tf}}, \omega_{\text{tr}}, k_{\text{us}}) = \mathbf{a}_1(a, b, K_{\text{sf}}, K_{\text{sr}}, K_{\text{tf}}, K_{\text{tr}}, C_{\alpha\text{f}}, C_{\alpha\text{r}}) \\
& K_{\text{sf}} = \mathbf{a}_2(Z_{\text{sf}}, K_{\text{Lf}}, K_{\text{Bf}}, L_{0\text{f}}) \\
& K_{\text{sr}} = \mathbf{a}_2(Z_{\text{sr}}, K_{\text{Lr}}, K_{\text{Br}}, L_{0\text{r}}) \\
& (K_{\text{tf}}, K_{\text{tr}}) = \mathbf{a}_3(P_{\text{if}}, P_{\text{ir}}, a, b) \\
& (C_{\alpha\text{f}}, C_{\alpha\text{r}}) = \mathbf{a}_4(P_{\text{if}}, P_{\text{ir}}, a, b) \\
& (K_{\text{Lf}}, K_{\text{Bf}}) = \mathbf{a}_5(D_{\text{f}}, d_{\text{f}}, p_{\text{f}}, L_{0\text{f}}) \\
& (K_{\text{Lr}}, K_{\text{Br}}) = \mathbf{a}_5(D_{\text{r}}, d_{\text{r}}, p_{\text{r}}, L_{0\text{r}})
\end{array}
\tag{5.4}
$$

The partition of this problem given in [10] is specified in Ψ below, and illustrated in Figure 5.6(a). The system *Chassis* has seven sub-components: *Vehicle*, *Tire*, *Corner*, two of type *Suspension*, and two of type *Spring*. Each sub-component includes its relevant set of optimization variables and functions. Similar to the portal frame example, the similarity of the front and rear suspensions and springs is exploited by defining a single suspension and a single spring component, which are then instantiated multiple times in system *Chassis*.

comp *Vehicle* =
|[extvar $a, b, K_{\text{sf}}, K_{\text{sr}}, K_{\text{tf}}, K_{\text{tr}}, C_{\alpha\text{f}}, C_{\alpha\text{r}}$
intvar $\omega_{\text{sf}}, \omega_{\text{sr}}, \omega_{\text{tf}}, \omega_{\text{tr}}, k_{\text{us}}$
objfunc $\mathbf{f}(\omega_{\text{sf}}, \omega_{\text{sr}}, \omega_{\text{tf}}, \omega_{\text{tr}}, k_{\text{us}})$
resfunc $(\omega_{\text{sf}}, \omega_{\text{sr}}, \omega_{\text{tf}}, \omega_{\text{tr}}, k_{\text{us}}) =$
$\mathbf{a}_1(a, b, K_{\text{sf}}, K_{\text{sr}}, K_{\text{tf}}, K_{\text{tr}}, C_{\alpha\text{f}}, C_{\alpha\text{r}})$
]|



(a) Original partition [10]



(b) Alternative partition with suspension-spring subsystems

Figure 5.6: Two partitions for the chassis design example.

comp *Tire* =
|[extvar $a, b, K_{\mathrm{tf}}, K_{\mathrm{tr}}, P_{\mathrm{if}}, P_{\mathrm{ir}}$
resfunc $(K_{\mathrm{tf}}, K_{\mathrm{tr}}) = \mathbf{a}_3(P_{\mathrm{if}}, P_{\mathrm{ir}}, a, b)$
]|

comp *Corner* =
|[extvar $a, b, C_{\alpha\mathrm{f}}, C_{\alpha\mathrm{r}}, P_{\mathrm{if}}, P_{\mathrm{ir}}$
resfunc $(C_{\alpha\mathrm{f}}, C_{\alpha\mathrm{r}}) = \mathbf{a}_4(P_{\mathrm{if}}, P_{\mathrm{ir}}, a, b)$
]|

comp *Suspension* =
|[extvar $K_{\mathrm{s}}, K_{\mathrm{L}}, K_{\mathrm{B}}, L_0$
intvar Z_{s}
confunc $\mathbf{g}_1(Z_{\mathrm{s}}, K_{\mathrm{L}}, K_{\mathrm{B}}, L_0)$
resfunc $K_{\mathrm{s}} = \mathbf{a}_2(Z_{\mathrm{s}}, K_{\mathrm{L}}, K_{\mathrm{B}}, L_0)$
]|

comp *Spring* =
|[extvar $K_{\mathrm{L}}, K_{\mathrm{B}}, L_0$
intvar D, d, p
confunc $\mathbf{g}_2(D, d, p)$
resfunc $(K_{\mathrm{L}}, K_{\mathrm{B}}) = \mathbf{a}_5(D, d, p, L_0)$
]|

syst *Chassis* =
|[sub V: *Vehicle*, T: *Tire*, C: *Corner*
, $S_{\mathrm{f}}, S_{\mathrm{r}}$: *Suspension*, Sp_{f},Sp_{r}: *Spring*
link $V.a$ -- $\{T.a, C.a\}$, $T.P_{\mathrm{if}}$ -- $C.P_{\mathrm{if}}$
, $V.b$ -- $\{T.b, C.b\}$, $T.P_{\mathrm{ir}}$ -- $C.P_{\mathrm{ir}}$
, $V.K_{\mathrm{tf}}$ -- $T.K_{\mathrm{tf}}$, $V.C_{\alpha\mathrm{f}}$ -- $C.C_{\alpha\mathrm{f}}$
, $V.K_{\mathrm{tr}}$ -- $T.K_{\mathrm{tr}}$, $V.C_{\alpha\mathrm{r}}$ -- $C.C_{\alpha\mathrm{r}}$
, $V.K_{\mathrm{sf}}$ -- $S_{\mathrm{f}}.K_{\mathrm{s}}$, $V.K_{\mathrm{sr}}$ -- $S_{\mathrm{r}}.K_{\mathrm{s}}$
, $S_{\mathrm{f}}.K_{\mathrm{L}}$ -- $Sp_{\mathrm{f}}.K_{\mathrm{L}}$, $S_{\mathrm{f}}.L_0$ -- $Sp_{\mathrm{f}}.L_0$
, $S_{\mathrm{r}}.K_{\mathrm{L}}$ -- $Sp_{\mathrm{r}}.K_{\mathrm{L}}$, $S_{\mathrm{r}}.L_0$ -- $Sp_{\mathrm{r}}.L_0$
, $S_{\mathrm{f}}.K_{\mathrm{B}}$ -- $Sp_{\mathrm{f}}.K_{\mathrm{B}}$
, $S_{\mathrm{r}}.K_{\mathrm{B}}$ -- $Sp_{\mathrm{r}}.K_{\mathrm{B}}$
]|

topsyst *Chassis*

Instead of including the *Suspension* and *Spring* components directly in the system, an alternative is to define a subsystem *SuspSpring* that includes a *Suspension* and a *Spring* component. To complete the partition, two instantiations of this subsystem are included in a system *Chassis*$_2$ that also includes the *Vehicle*, *Tire*, and *Corner* components of the above definition. The specification of this alternative partition is given below.

syst *SuspSpring* =
|[sub S: *Suspension*, *Sp*: *Spring*
link $S.K_{\mathrm{L}}$ -- *Sp*.K_{L}, $S.L_0$ -- *Sp*.L_0, $S.K_{\mathrm{B}}$ -- *Sp*.K_{B}
alias $K_{\mathrm{s}} = S.K_{\mathrm{s}}$
]|

syst *Chassis*$_2$ =

|[sub V: *Vehicle*, T: *Tire*, C: *Corner*
, $S_\mathrm{f}, S_\mathrm{r}$: *SuspSpring*
link $V.a$ -- $\{T.a, C.a\}$, $T.P_\mathrm{if}$ -- $C.P_\mathrm{if}$
, $V.b$ -- $\{T.b, C.b\}$, $T.P_\mathrm{ir}$ -- $C.P_\mathrm{ir}$
, $V.K_\mathrm{tf}$ -- $T.K_\mathrm{tf}$, $V.C_{\alpha\mathrm{f}}$ -- $C.C_{\alpha\mathrm{f}}$
, $V.K_\mathrm{tr}$ -- $T.K_\mathrm{tr}$, $V.C_{\alpha\mathrm{r}}$ -- $C.C_{\alpha\mathrm{r}}$
, $V.K_\mathrm{sf}$ -- $S_\mathrm{f}.K_\mathrm{s}$, $V.K_\mathrm{sr}$ -- $S_\mathrm{r}.K_\mathrm{s}$
]|

topsyst *Chassis*$_2$

The couplings between the variables of *Suspension* and *Spring* are included in the system *SuspSpring*, in which an alias K_s is introduced to make this variable of component *Suspension* accessible by system *Chassis*$_2$. Two systems *SuspSpring* are instantiated in system *Chassis*$_2$, and links between the different sub-components are defined appropriately. With this alternative partition, the coordination of the *SuspSpring* subsystems can be performed nested within the coordination of the top-level system *Chassis*$_2$.

5 Supersonic business jet

The fifth example considers the range maximization for a supersonic business jet while considering the disciplines Propulsion, Aerodynamics, Structures, and Range. Figure 5.7 depicts the data exchange between these disciplines (see, e.g. [1, 17] for a more detailed description of the example). The reader is referred to Table 5.2 for a brief description of the optimization variables.

The integrated problem formulation is given by

$$
\begin{aligned}
\text{find} \quad & \mathbf{y}_\mathrm{sa}, \mathbf{x}_\mathrm{a}, \mathbf{x}_\mathrm{s}, \mathbf{x}_\mathrm{p}, \\
& \quad h, M, R, \mathit{ESF}, W_\mathrm{e}, \mathit{SFC}, L, \tfrac{L}{D}, D, W_\mathrm{t}, W_\mathrm{f}, \theta \\
\min \quad & f_\mathrm{r}(R) \\
\text{subject to} \quad & \mathbf{g}_\mathrm{p}(\mathbf{x}_\mathrm{p}, h, M, D) \leq \mathbf{0} \\
& \mathbf{g}_\mathrm{a}(\mathbf{y}_\mathrm{sa}, \mathbf{x}_\mathrm{a}, h, M, \mathit{ESF}, W_\mathrm{t}, \theta) \leq \mathbf{0} \\
& \mathbf{g}_\mathrm{s}(\mathbf{y}_\mathrm{sa}, \mathbf{x}_\mathrm{s}, W_\mathrm{e}, L) \leq \mathbf{0} \\
& R = \mathbf{a}_\mathrm{r}(h, M, \mathit{SFC}, \tfrac{L}{D}, W_\mathrm{t}, W_\mathrm{f}) \\
& (\mathit{ESF}, W_\mathrm{e}, \mathit{SFC}) = \mathbf{a}_\mathrm{p}(\mathbf{x}_\mathrm{p}, h, M, D) \\
& (L, \tfrac{L}{D}, D) = \mathbf{a}_\mathrm{a}(\mathbf{y}_\mathrm{sa}, \mathrm{x}_\mathrm{a}, h, M, \mathit{ESF}, W_\mathrm{t}, \theta) \\
& (W_\mathrm{t}, W_\mathrm{f}, \theta) = \mathbf{a}_\mathrm{s}(\mathbf{y}_\mathrm{sa}, \mathbf{x}_\mathrm{s}, W_\mathrm{e}, L) \\
\text{where} \quad & \mathbf{y}_\mathrm{sa} = [\tfrac{t}{c}, \mathit{AR}_\mathrm{w}, \Lambda_\mathrm{w}, S_\mathrm{ref}, S_\mathrm{ht}, \mathit{AR}_\mathrm{ht}] \\
& \mathbf{x}_\mathrm{a} = [\Lambda_\mathrm{ht}, L_\mathrm{w}, L_\mathrm{ht}] \\
& \mathbf{x}_\mathrm{s} = [[t], [t_\mathrm{s}], \lambda] \\
& \mathbf{x}_\mathrm{p} = [T]
\end{aligned}
\tag{5.5}
$$

Here, the response variables $R, \mathit{ESF}, W_\mathrm{e}, \mathit{SFC}, L, \frac{L}{D}, D, W_\mathrm{t}, W_\mathrm{f}, \theta$ are included as optimization variables.

A possible partition for this problem simply defines one component for each discipline, and defines the links according to the functional dependencies (see Figure 5.8(a)):

comp *Range* =
|[extvar $h, M, \frac{L}{D}, SFC, W_\mathrm{t}, W_\mathrm{f}$
intvar R
objfunc $f_\mathrm{r}(R)$

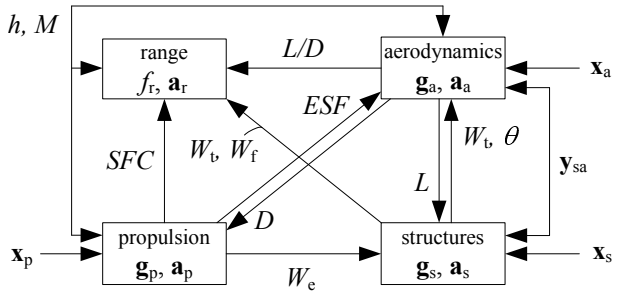
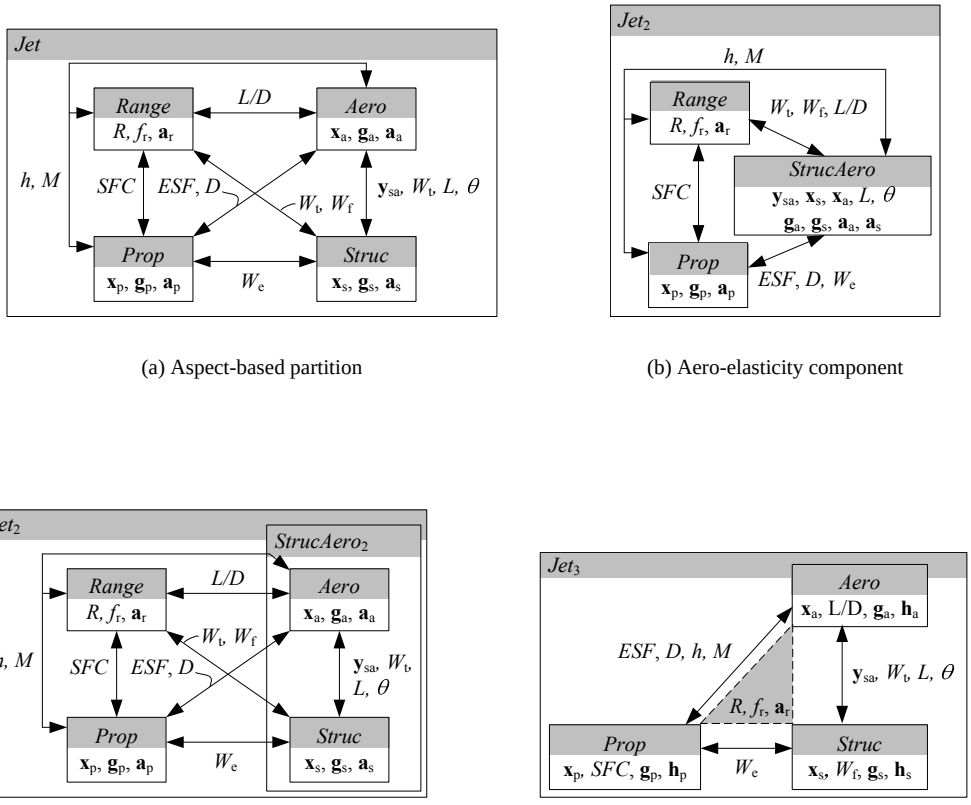


Figure 5.7: Functional dependencies for the business jet example. Boxes represent functions, and arrow directions are associated with variable dependencies. Coupling variables have two-headed arrows.

Table 5.2: Description of the optimization variables for the supersonic business jet problem.

Design variables		Response variables	
h	cruise altitude	ESF	engine scaling factor
M	cruise speed	W_e	engine weight
$\frac{t}{c}$	thickness/chord	SFC	specific fuel consumption
AR_w	wing aspect ratio	L	total lift
Λ_w	wing sweep angle	$\frac{L}{D}$	lift/drag ratio
S_{ref}	wing surface area	D	total drag
S_{ht}	tail surface area	W_t	total weight
AR_{ht}	tail aspect ratio	W_f	fuel weight
T	throttle	θ	wing twist
Λ_{ht}	tail sweep	R	range
L_w	wing lift		
L_{ht}	tail lift		
$\mathbf{t}$	wing box thicknesses		
$\mathbf{t}_s$	wing box thicknesses		
λ	taper ratio		



(a) Aspect-based partition

(b) Aero-elasticity component

(c) Aero-elasticity subsystem

(d) Range coupling function

Figure 5.8: Four alternative partitions for the business jet example.

resfunc $R = \mathbf{a}_{\mathrm{r}}(h, M, \frac{L}{D}, SFC, W_{\mathrm{t}}, W_{\mathrm{f}})$
]|

comp *Struc* =
|[extvar $\mathbf{y}_{\mathrm{sa}}, W_{\mathrm{f}}, W_{\mathrm{e}}, L, W_{\mathrm{t}}, \theta$
intvar $\mathbf{x}_{\mathrm{s}}$
confunc $\mathbf{g}_{\mathrm{s}}(\mathbf{y}_{\mathrm{sa}}, \mathbf{x}_{\mathrm{s}}, W_{\mathrm{e}}, L)$
resfunc $(W_{\mathrm{t}}, W_{\mathrm{f}}, \theta) = \mathbf{a}_{\mathrm{s}}(\mathbf{y}_{\mathrm{sa}}, \mathbf{x}_{\mathrm{s}}, W_{\mathrm{e}}, L)$
]|

comp *Aero* =
|[extvar $\mathbf{y}_{\mathrm{sa}}, h, M, \frac{L}{D}, ESF, D, L, W_{\mathrm{t}}, \theta$
intvar $\mathbf{x}_{\mathrm{a}}$
confunc $\mathbf{g}_{\mathrm{a}}(\mathbf{y}_{\mathrm{sa}}, \mathbf{x}_{\mathrm{a}}, h, M, ESF, W_{\mathrm{t}}, \theta)$
resfunc $(L, \frac{L}{D}, D) = \mathbf{a}_{\mathrm{a}}(\mathbf{y}_{\mathrm{sa}}, \mathrm{x}_{\mathrm{a}}, h, M, ESF, W_{\mathrm{t}}, \theta)$
]|

comp *Prop* =
|[extvar $h, M, SFC, ESF, D, W_{\mathrm{e}}$
intvar $\mathbf{x}_{\mathrm{p}}$
confunc $\mathbf{g}_{\mathrm{p}}(\mathbf{x}_{\mathrm{p}}, h, M, D)$
resfunc $(ESF, W_{\mathrm{e}}, SFC) = \mathbf{a}_{\mathrm{p}}(\mathbf{x}_{\mathrm{p}}, h, M, D)$
]|

syst *Jet* =
|[sub R: *Range*, S: *Struc*, A: *Aero*, P: *Prop*
link $R.\frac{L}{D}$ -- $A.\frac{L}{D}$, $R.h$ -- $\{A.h, P.h\}$
, $R.W_{\mathrm{f}}$ -- $S.W_{\mathrm{f}}$, $R.M$ -- $\{A.M, P.M\}$
, $R.W_{\mathrm{t}}$ -- $S.W_{\mathrm{t}}$, $R.SFC$ -- $P.SFC$
, $A.D$ -- $P.D$, $A.ESF$ -- $P.ESF$
, $A.\mathbf{y}_{\mathrm{sa}}$ -- $S.\mathbf{y}_{\mathrm{sa}}$, $A.L$ -- $S.L$
, $A.W_{\mathrm{t}}$ -- $S.W_{\mathrm{t}}$, $A.\theta$ -- $S.\theta$
, $P.W_{\mathrm{e}}$ -- $S.W_{\mathrm{e}}$
]|

topsyst *Jet*

Each component includes the variables relevant to its objective or constraint functions. Variables that couple two or more components are defined as external, while variables that are local to a single component are defined as internal. The system definition first instantiates the four components, after which the external variables are coupled appropriately.

In aircraft design, the Structures and Aerodynamics components are often strongly coupled. An alternative partition could therefore merge these disciplines into an aero-elasticity component *StrucAero* (see Figure 5.8(b)) that is defined as

comp *StrucAero* =
|[extvar $h, M, ESF, W_{\mathrm{e}}, \frac{L}{D}, D, W_{\mathrm{t}}, W_{\mathrm{f}}$
intvar $\mathbf{y}_{\mathrm{sa}}, \mathbf{x}_{\mathrm{s}}, \mathbf{x}_{\mathrm{a}}, L, \theta$
confunc $\mathbf{g}_{\mathrm{s}}(\mathbf{y}_{\mathrm{sa}}, \mathbf{x}_{\mathrm{s}}, W_{\mathrm{e}}, L)$
, $\mathbf{g}_{\mathrm{a}}(\mathbf{y}_{\mathrm{sa}}, \mathbf{x}_{\mathrm{a}}, h, M, ESF, W_{\mathrm{t}}, \theta)$
resfunc $(W_{\mathrm{t}}, W_{\mathrm{f}}, \theta) = \mathbf{a}_{\mathrm{s}}(\mathbf{y}_{\mathrm{sa}}, \mathbf{x}_{\mathrm{s}}, W_{\mathrm{e}}, L)$
, $(L, \frac{L}{D}, D) = \mathbf{a}_{\mathrm{a}}(\mathbf{y}_{\mathrm{sa}}, \mathrm{x}_{\mathrm{a}}, h, M, ESF, W_{\mathrm{t}}, \theta)$
]|

Observe that the response functions of this component are coupled. The arguments of function $\mathbf{a}_\text{s}$ include the lift L, which is computed by the function $\mathbf{a}_\text{a}$. In turn, the arguments of $\mathbf{a}_\text{a}$ include the total weight W_t and wing twist θ that are computed by $\mathbf{a}_\text{s}$.

The system definition *Jet*$_2$ completes the specification for this second partition.

```
syst Jet2 =
|[  sub R: Range, SA: StrucAero, P: Prop
    link R.L/D -- SA.L/D,   R.h -- {SA.h, P.h}
       , R.Wf -- SA.Wf, R.M -- {SA.M, P.M}
       , R.Wt -- SA.Wt, R.SFC -- P.SFC
       , P.D -- SA.D,   P.ESF -- SA.ESF
       , P.We -- SA.We
]|

topsyst Jet2
```

Another alternative is to define a subsystem that includes Structures and Aerodynamics as components (see Figure 5.8(c)). Components S and A are then equal to the first definition, and the above system definition for *Jet*$_2$ can be used if the component *StrucAero* is replaced with subsystem *StrucAero*$_2$ defined as

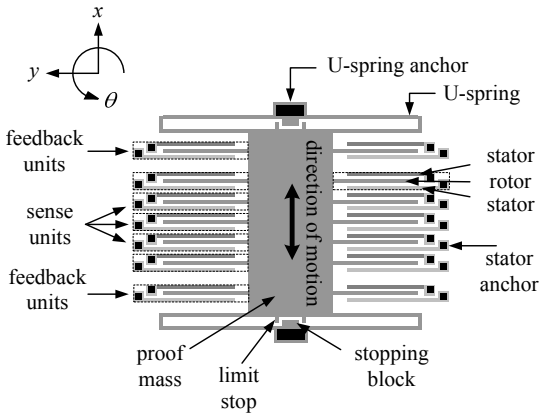
```
syst StrucAero2 =
|[  sub S: Struc, A: Aero
    link A.ysa -- S.ysa, A.L -- S.L
       , A.Wt -- S.Wt, A.θ -- S.θ
    alias Wf = S.Wf, ESF = A.ESF
        , We = S.We, L/D = A.L/D
        , Wt = S.Wt, D = A.D
        , h = A.h,,  M = A.M
]|
```

The system *StrucAero*$_2$ defines the links between components S and A, and defines aliases for those variables that need to be externally accessible. Observe that the same system *Jet*$_2$ can be used. The system definition remains correct, even though the definition of one of its subcomponent has changed. This feature allows the designer to separate the different levels in a system's hierarchy.

The final partition of the business jet example defines the range objective f_r and range analysis $\mathbf{a}_\text{r}$ as coupling functions instead of a separate component (see Figure 5.8(d)). Components S, A, and P of the first partition can be used, when system *Jet*$_3$ is defined to include f_r and $\mathbf{a}_\text{r}$ as coupling functions and R as an internal response variable.

```
syst Jet3 =
|[  intvar R
    sub S: Struc, A: Aero, P: Prop
    objfunc fr(R)
    resfunc R = ar(A.h, A.M, A.L/D, P.SFC, S.Wf, S.Wt)
    link A.ESF -- P.ESF, A.D -- P.D
       , A.h -- P.h,          A.M -- P.M
       , A.ysa -- S.ysa,      A.L -- S.L
       , A.Wt -- S.Wt,        A.θ -- S.θ
       , P.We -- S.We
```

(a) Scanning electron microscope image of suspended proof mass.



(b) Schematic illustration of the sensing mechanism (top view).

Figure 5.9: Layout of micro-accelerometer by Analog Devices [16, 2]. The suspended proof with comb fingers takes up an area of about 600 μm $\times$ 700 μm = 0.42 mm^2.

]|

topsyst Jet_3

where the dependence of the range function on variables from the different subsystems can be observed.

6 Micro-accelerometer

Our final example is the micro-accelerometer problem taken from [21]. The design objective for this example is to minimize the footprint area A while meeting a number of performance constraints on sensitivity, noise floor, and measuring range. Additional functional constraints are imposed to assure proper functioning of the accelerometer. A description of the design variables is given in Table 5.3, and the reader is referred to [21] for a detailed description of the problem.

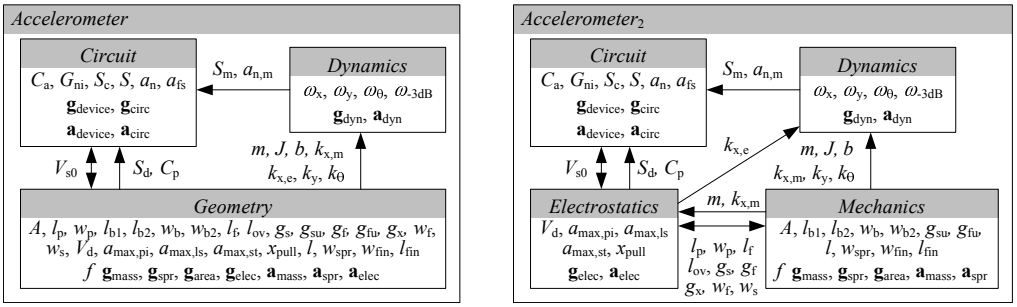
Table 5.3: Description of the optimization variables for the micro-accelerometer design problem.

Design variables		Response variables	
l_{p}	length of proof mass	m	effective mass
w_{p}	width of proof mass	J	moment of inertia
l_{b1}	length of U-spring beam 1	b	damping coefficient
l_{b2}	length of U-spring beam 2	$k_{\text{x,m}}$	spring stiffness in x
w_{b}	width of U-spring beams 1,3	k_{y}	spring stiffness in y
w_{b2}	width of U-spring beam 2	k_{θ}	spring stiffness in ϑ
l_{f}	length of rotor fingers	A	required wafer area
l_{ov}	length of finger overlap	S_{d}	diff. capacitance sensitivity
g_{s}	gap sense rotor/stator fingers	$k_{\text{x,e}}$	spring softening contribution
g_{su}	gap sense units	$a_{\max,\text{pi}}$	max. detectable acc. pull-in
g_{f}	gap feedback rotor/stator fingers	$a_{\max,\text{ls}}$	max. detectable acc. limit stop
g_{fu}	gap feedback units	$a_{\max,\text{st}}$	max. self test acc.
g_{x}	x-gap of limit stop	x_{pull}	pull-in displacement
w_{f}	width of rotor fingers	C_{p}	parasitic capacitance proof mass/substrate
w_{s}	width of stator finger	ω_{x}	resonance frequency in x
V_{s0}	source voltage amplitude	ω_{y}	resonance frequency in y
V_{d}	differential drive voltage	ω_{θ}	resonance frequency in x
C_{a}	charge amplifier capacitance	$\omega_{\text{-3dB}}$	mechanical bandwidth
G_{ni}	non-inverting amplifier gain	S_{m}	mechanical sensitivity
		$a_{\text{n,m}}$	noise equiv. acceleration
		S_{c}	circuit sensitivity
		S	total sensitivity
		a_{n}	total noise equiv. acc.
		a_{fs}	full-scale range
		l	total length of device
		w_{spr}	width of device at springs
		w_{fin}	width of device at fingers
		l_{fin}	length span of fingers

The problem is given by:

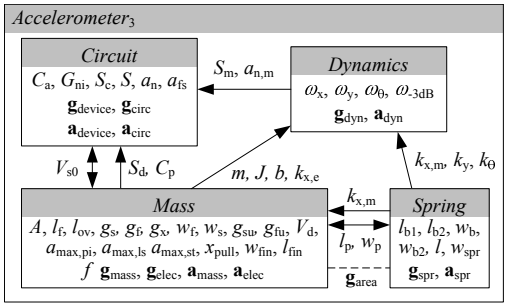
$$
\begin{aligned}
\text{find} \quad & l_{\text{p}}, w_{\text{p}}, l_{\text{b1}}, l_{\text{b2}}, w_{\text{b}}, w_{\text{b2}}, l_{\text{f}}, l_{\text{ov}}, g_{\text{s}}, g_{\text{su}}, g_{\text{f}}, g_{\text{fu}}, g_{\text{x}}, w_{\text{f}}, w_{\text{s}}, V_{\text{s0}}, V_{\text{d}}, C_{\text{a}}, G_{\text{ni}}, \\
& A, m, J, b, k_{\text{x,m}}, k_{\text{x,e}}, k_{\text{y}}, k_{\theta}, S_{\text{d}}, k_{\text{x,e}}, a_{\text{max,pi}}, a_{\text{max,ls}}, a_{\text{max,st}}, x_{\text{pull}}, C_{\text{p}}, \\
& S_{\text{m}}, a_{\text{n,m}}, \omega_{\text{x}}, \omega_{\text{y}}, \omega_{\theta}, \omega_{\text{-3dB}}, S_{\text{c}}, S, a_{\text{n}}, a_{\text{fs}}, l, w_{\text{spr}}, w_{\text{fin}}, l_{\text{fin}} \\
\min \quad & f(A) \\
\text{subject to} \quad & \mathbf{g}_{\text{device}}(S, a_{\text{n}}, a_{\text{fs}}) \leq \mathbf{0} \\
& \mathbf{g}_{\text{mass}}(l_{\text{p}}, w_{\text{p}}, l_{\text{f}}, l_{\text{ov}}, g_{\text{s}}, g_{\text{su}}, g_{\text{f}}, g_{\text{fu}}, g_{\text{x}}, w_{\text{f}}, w_{\text{s}}) \leq \mathbf{0} \\
& \mathbf{g}_{\text{spr}}(l_{\text{p}}, w_{\text{p}}, l_{\text{b1}}, l_{\text{b2}}, w_{\text{b}}, w_{\text{b2}}) \leq \mathbf{0} \\
& \mathbf{g}_{\text{area}}(g_{\text{x}}, l_{\text{b2}}, l, w_{\text{spr}}, l_{\text{fin}}, w_{\text{fin}}, A) \leq \mathbf{0} \\
& \mathbf{g}_{\text{elec}}(g_{\text{s}}, g_{\text{x}}, x_{\text{pull}}, k_{\text{x,e}}, k_{\text{x,m}}, a_{\text{max,pi}}, a_{\text{max,ls}}, a_{\text{max,st}}) \leq \mathbf{0} \\
& \mathbf{g}_{\text{dyn}}(\omega_{\text{x}}, \omega_{\text{y}}, \omega_{\theta}, \omega_{\text{-3dB}}) \leq \mathbf{0} \\
& \mathbf{g}_{\text{circ}}(C_{\text{a}}, C_{\text{p}}) \leq \mathbf{0} \\
& (S, a_{\text{n}}, a_{\text{fs}}) = \mathbf{a}_{\text{device}}(S_{\text{m}}, S_{\text{d}}, S_{\text{c}}, a_{\text{n,m}}, V_{\text{s0}}) \\
& (A, m, J, b, w_{\text{fin}}, l_{\text{fin}}) = \mathbf{a}_{\text{mass}}(l_{\text{p}}, w_{\text{p}}, l_{\text{f}}, l_{\text{ov}}, g_{\text{s}}, g_{\text{su}}, g_{\text{f}}, g_{\text{fu}}, g_{\text{x}}, w_{\text{f}}, w_{\text{s}}) \\
& (k_{\text{x,m}}, k_{\text{y}}, k_{\theta}, w_{\text{spr}}, l) = \mathbf{a}_{\text{spr}}(l_{\text{p}}, w_{\text{p}}, l_{\text{b1}}, l_{\text{b2}}, w_{\text{b}}, w_{\text{b2}}) \\
& (S_{\text{m}}, a_{\text{n,m}}, \omega_{\text{x}}, \omega_{\text{y}}, \omega_{\theta}, \omega_{\text{-3dB}}) = \mathbf{a}_{\text{dyn}}(m, J, b, k_{\text{x,m}}, k_{\text{x,e}}, k_{\text{y}}, k_{\theta}) \\
& S_{\text{c}} = \mathbf{a}_{\text{circ}}(V_{\text{s0}}, C_{\text{a}}, G_{\text{ni}}) \\
& (S_{\text{d}}, k_{\text{x,e}}, a_{\text{max,pi}}, a_{\text{max,ls}}, a_{\text{max,st}}, x_{\text{pull}}, C_{\text{p}}) \\
& \qquad = \mathbf{a}_{\text{elec}}(m, k_{\text{x,m}}, l_{\text{p}}, w_{\text{p}}, l_{\text{f}}, l_{\text{ov}}, g_{\text{s}}, g_{\text{f}}, g_{\text{x}}, w_{\text{f}}, w_{\text{s}}, V_{\text{s0}}, V_{\text{d}})
\end{aligned}
\tag{5.6}
$$

In [21], four partitions of the above problem are presented. All four partitions are illustrated in Figure 5.10. The first partition is has three subproblems *Circuit*, *Sensor*, and *Geometry*. Sub-

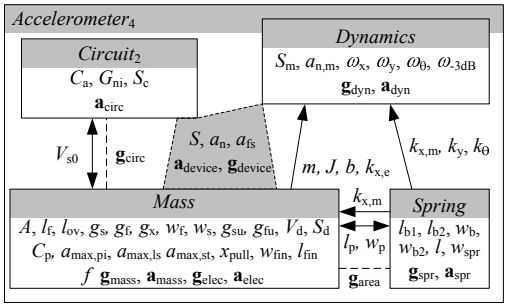


(a) First partition

(b) Second partition



(c) Third partition



(d) Fourth partition

Figure 5.10: Four alternative partitions for the micro-accelerometer example.

problem *Geometry* includes variables and functions associated with the dimensioning of the sensor area, *Sensor* includes dynamical variables and functions, and *Circuit* entails the electronic design. The specification for this first partition is given below.

comp *Circuit* =
|[extvar $S_{\mathrm{m}}, a_{\mathrm{n,m}}, S_{\mathrm{d}}, C_{\mathrm{p}}, V_{\mathrm{s0}}$
intvar $C_{\mathrm{a}}, G_{\mathrm{ni}}, S_{\mathrm{c}}, S, a_{\mathrm{n}}, a_{\mathrm{fs}}$
confunc $\mathbf{g}_{\mathrm{device}}(S, a_{\mathrm{n}}, a_{\mathrm{fs}})$
, $\mathbf{g}_{\mathrm{circ}}(C_{\mathrm{a}}, C_{\mathrm{p}})$
resfunc $(S, a_{\mathrm{n}}, a_{\mathrm{fs}}) = \mathbf{a}_{\mathrm{device}}(S_{\mathrm{m}}, S_{\mathrm{d}}, S_{\mathrm{c}}, a_{\mathrm{n,m}}, V_{\mathrm{s0}})$
, $S_{\mathrm{c}} = \mathbf{a}_{\mathrm{circ}}(V_{\mathrm{s0}}, C_{\mathrm{a}}, G_{\mathrm{ni}})$
]|

comp *Dynamics* =
|[extvar $S_{\mathrm{m}}, a_{\mathrm{n,m}}, m, J, b, k_{\mathrm{x,m}}, k_{\mathrm{x,e}}, k_{\mathrm{y}}, k_{\theta}$
intvar $\omega_{\mathrm{x}}, \omega_{\mathrm{y}}, \omega_{\theta}, \omega_{\text{-3dB}}$
confunc $\mathbf{g}_{\mathrm{dyn}}(\omega_{\mathrm{x}}, \omega_{\mathrm{y}}, \omega_{\theta}, \omega_{\text{-3dB}})$
resfunc $(S_{\mathrm{m}}, a_{\mathrm{n,m}}, \omega_{\mathrm{x}}, \omega_{\mathrm{y}}, \omega_{\theta}, \omega_{\text{-3dB}}) = \mathbf{a}_{\mathrm{dyn}}(m, J, b, k_{\mathrm{x,m}}, k_{\mathrm{x,e}}, k_{\mathrm{y}}, k_{\theta})$
]|

comp *Geometry* =
|[extvar $m, J, b, k_{\mathrm{x,m}}, k_{\mathrm{x,e}}, k_{\mathrm{y}}, k_{\theta}, S_{\mathrm{d}}, C_{\mathrm{p}}, V_{\mathrm{s0}}$
intvar $A, l_{\mathrm{p}}, w_{\mathrm{p}}, l_{\mathrm{b1}}, l_{\mathrm{b2}}, w_{\mathrm{b}}, w_{\mathrm{b2}}, l_{\mathrm{f}}, l_{\mathrm{ov}}, g_{\mathrm{s}}, g_{\mathrm{su}}, g_{\mathrm{f}}, g_{\mathrm{fu}}, g_{\mathrm{x}}, w_{\mathrm{f}}, w_{\mathrm{s}}, V_{\mathrm{d}}$
, $a_{\mathrm{max,pi}}, a_{\mathrm{max,ls}}, a_{\mathrm{max,st}}, x_{\mathrm{pull}}, l, w_{\mathrm{spr}}, w_{\mathrm{fin}}, l_{\mathrm{fin}}$
objfunc $f(A)$
confunc $\mathbf{g}_{\mathrm{mass}}(l_{\mathrm{p}}, w_{\mathrm{p}}, l_{\mathrm{f}}, l_{\mathrm{ov}}, g_{\mathrm{s}}, g_{\mathrm{su}}, g_{\mathrm{f}}, g_{\mathrm{fu}}, g_{\mathrm{x}}, w_{\mathrm{f}}, w_{\mathrm{s}})$
, $\mathbf{g}_{\mathrm{spr}}(l_{\mathrm{p}}, w_{\mathrm{p}}, l_{\mathrm{b1}}, l_{\mathrm{b2}}, w_{\mathrm{b}}, w_{\mathrm{b2}})$
, $\mathbf{g}_{\mathrm{area}}(g_{\mathrm{x}}, l_{\mathrm{b2}}, l, w_{\mathrm{spr}}, l_{\mathrm{fin}}, w_{\mathrm{fin}}, A)$
, $\mathbf{g}_{\mathrm{elec}}(g_{\mathrm{s}}, g_{\mathrm{x}}, x_{\mathrm{pull}}, k_{\mathrm{x,e}}, k_{\mathrm{x,m}}, a_{\mathrm{max,pi}}, a_{\mathrm{max,ls}}, a_{\mathrm{max,st}})$
resfunc $(A, m, J, b, w_{\mathrm{fin}}, l_{\mathrm{fin}}) = \mathbf{a}_{\mathrm{mass}}(l_{\mathrm{p}}, w_{\mathrm{p}}, l_{\mathrm{f}}, l_{\mathrm{ov}}, g_{\mathrm{s}}, g_{\mathrm{su}}, g_{\mathrm{f}}, g_{\mathrm{fu}}, g_{\mathrm{x}}, w_{\mathrm{f}}, w_{\mathrm{s}})$
, $(k_{\mathrm{x,m}}, k_{\mathrm{y}}, k_{\theta}, w_{\mathrm{spr}}, l) = \mathbf{a}_{\mathrm{spr}}(l_{\mathrm{p}}, w_{\mathrm{p}}, l_{\mathrm{b1}}, l_{\mathrm{b2}}, w_{\mathrm{b}}, w_{\mathrm{b2}})$
, $(S_{\mathrm{d}}, k_{\mathrm{x,e}}, a_{\mathrm{max,pi}}, a_{\mathrm{max,ls}}, a_{\mathrm{max,st}}, x_{\mathrm{pull}}, C_{\mathrm{p}})$
$= \mathbf{a}_{\mathrm{elec}}(m, k_{\mathrm{x,m}}, l_{\mathrm{p}}, w_{\mathrm{p}}, l_{\mathrm{f}}, l_{\mathrm{ov}}, g_{\mathrm{s}}, g_{\mathrm{f}}, g_{\mathrm{x}}, w_{\mathrm{f}}, w_{\mathrm{s}}, V_{\mathrm{s0}}, V_{\mathrm{d}})$
]|

syst *Accelerometer* =
|[sub C: *Circuit*, D: *Dynamics*, G: *Geometry*
link $C.S_{\mathrm{m}}$ -- $D.S_{\mathrm{m}}$, $C.a_{\mathrm{n,m}}$ -- $D.a_{\mathrm{n,m}}$
, $C.S_{\mathrm{d}}$ -- $G.S_{\mathrm{d}}$, $C.V_{\mathrm{s0}}$ -- $G.V_{\mathrm{s0}}$
, $C.C_{\mathrm{p}}$ -- $G.C_{\mathrm{p}}$, $D.k_{\mathrm{x,m}}$ -- $G.k_{\mathrm{x,m}}$
, $D.m$ -- $G.m$, $D.k_{\mathrm{x,e}}$ -- $G.k_{\mathrm{x,e}}$
, $D.J$ -- $G.J$, $D.k_{\mathrm{y}}$ -- $G.k_{\mathrm{y}}$
, $D.b$ -- $G.b$, $D.k_{\theta}$ -- $G.k_{\theta}$
]|

topsyst *Accelerometer*

The second partition splits the *Geometry* subproblem in two subproblems: *Mechanics* associated with structural analysis of the proof mass and U-springs, and *Electrostatics* is concerned with the electrostatic analysis of the comb drives. Subproblems *Circuit* and *Dynamics* of the previous partition can be used. The specification of *Mechanics* and *Electrostatics* is given below, together with the specification for system *Accelerometer*$_2$ that integrates the four subproblems.

comp *Mechanics* =

|[extvar $m, J, b, k_{\text{x,m}}, k_{\text{y}}, k_\theta, l_{\text{p}}, w_{\text{p}}, l_{\text{f}}, l_{\text{ov}}, g_{\text{x}}, w_{\text{f}}, w_{\text{s}}, g_{\text{s}}, g_{\text{f}}$
intvar $A, l_{\text{b1}}, l_{\text{b2}}, w_{\text{b}}, w_{\text{b2}}, g_{\text{su}}, g_{\text{fu}}, l, w_{\text{spr}}, w_{\text{fin}}, l_{\text{fin}}$
objfunc $f(A)$
confunc $\mathbf{g}_{\text{mass}}(l_{\text{p}}, w_{\text{p}}, l_{\text{f}}, l_{\text{ov}}, g_{\text{s}}, g_{\text{su}}, g_{\text{f}}, g_{\text{fu}}, g_{\text{x}}, w_{\text{f}}, w_{\text{s}})$
, $\mathbf{g}_{\text{spr}}(l_{\text{p}}, w_{\text{p}}, l_{\text{b1}}, l_{\text{b2}}, w_{\text{b}}, w_{\text{b2}})$
, $\mathbf{g}_{\text{area}}(g_{\text{x}}, l_{\text{b2}}, l, w_{\text{spr}}, l_{\text{fin}}, w_{\text{fin}}, A)$
resfunc $(A, m, J, b, w_{\text{fin}}, l_{\text{fin}}) = \mathbf{a}_{\text{mass}}(l_{\text{p}}, w_{\text{p}}, l_{\text{f}}, l_{\text{ov}}, g_{\text{s}}, g_{\text{su}}, g_{\text{f}}, g_{\text{fu}}, g_{\text{x}}, w_{\text{f}}, w_{\text{s}})$
, $(k_{\text{x,m}}, k_{\text{y}}, k_\theta, w_{\text{spr}}, l) = \mathbf{a}_{\text{spr}}(l_{\text{p}}, w_{\text{p}}, l_{\text{b1}}, l_{\text{b2}}, w_{\text{b}}, w_{\text{b2}})$
]|

comp *Electrostatics* =
|[extvar $m, k_{\text{x,m}}, k_{\text{x,e}}, S_{\text{d}}, C_{\text{p}}, V_{\text{s0}}, l_{\text{p}}, w_{\text{p}}, l_{\text{f}}, l_{\text{ov}}, g_{\text{s}}, g_{\text{f}}, g_{\text{x}}, w_{\text{f}}, w_{\text{s}}$
intvar $V_{\text{d}}, a_{\text{max,pi}}, a_{\text{max,ls}}, a_{\text{max,st}}, x_{\text{pull}}$
confunc $\mathbf{g}_{\text{elec}}(g_{\text{s}}, g_{\text{x}}, x_{\text{pull}}, k_{\text{x,e}}, k_{\text{x,m}}, a_{\text{max,pi}}, a_{\text{max,ls}}, a_{\text{max,st}})$
resfunc $(S_{\text{d}}, k_{\text{x,e}}, a_{\text{max,pi}}, a_{\text{max,ls}}, a_{\text{max,st}}, x_{\text{pull}}, C_{\text{p}})$
$= \mathbf{a}_{\text{elec}}(m, k_{\text{x,m}}, l_{\text{p}}, w_{\text{p}}, l_{\text{f}}, l_{\text{ov}}, g_{\text{s}}, g_{\text{f}}, g_{\text{x}}, w_{\text{f}}, w_{\text{s}}, V_{\text{s0}}, V_{\text{d}})$
]|

syst *Accelerometer*$_2$ =
|[sub C: *Circuit*, D: *Dynamics*, M: *Mechanics*, E: *Electrostatics*
link $M.m$ -- $\{D.m,\ E.m\}$, $M.k_{\text{x,m}}$ -- $\{D.k_{\text{x,m}},\ E.k_{\text{x,m}}\}$
, $C.S_{\text{m}}$ -- $D.S_{\text{m}}$, $C.a_{\text{n,m}}$ -- $D.a_{\text{n,m}}$
, $C.S_{\text{d}}$ -- $E.S_{\text{d}}$, $D.k_{\text{x,e}}$ -- $E.k_{\text{x,e}}$
, $C.C_{\text{p}}$ -- $E.C_{\text{p}}$, $C.V_{\text{s0}}$ -- $E.V_{\text{s0}}$
, $M.l_{\text{p}}$ -- $E.l_{\text{p}}$, $D.k_{\text{y}}$ -- $M.k_{\text{y}}$
, $M.l_{\text{f}}$ -- $E.l_{\text{f}}$, $D.k_\theta$ -- $M.k_\theta$
, $M.g_{\text{s}}$ -- $E.g_{\text{s}}$, $M.l_{\text{ov}}$ -- $E.l_{\text{ov}}$
, $M.g_{\text{f}}$ -- $E.g_{\text{f}}$, $M.w_{\text{p}}$ -- $E.w_{\text{p}}$
, $M.g_{\text{x}}$ -- $E.g_{\text{x}}$, $M.w_{\text{f}}$ -- $E.w_{\text{f}}$
, $D.b$ -- $M.b$, $M.w_{\text{s}}$ -- $E.w_{\text{s}}$
, $D.J$ -- $M.J$
]|

topsyst *Accelerometer*$_2$

The third partition splits *Geometry* in two alternative parts: *Spring* associated with the design of the U-spring, and *Mass* that entails the design of the proof mass and comb fingers. Again, the specifications of *Circuit* and *Dynamics* of the first partition are used. The specification of the *Spring* and *Mass* components, and system *Accelerometer*$_3$ are given below.

comp *Spring* =
|[extvar $k_{\text{x,m}}, k_{\text{y}}, k_\theta, l_{\text{b2}}, l_{\text{p}}, w_{\text{p}}, l, w_{\text{spr}}$
intvar $l_{\text{b1}}, w_{\text{b}}, w_{\text{b2}}$
confunc $\mathbf{g}_{\text{spr}}(l_{\text{p}}, w_{\text{p}}, l_{\text{b1}}, l_{\text{b2}}, w_{\text{b}}, w_{\text{b2}})$
resfunc $(k_{\text{x,m}}, k_{\text{y}}, k_\theta, w_{\text{spr}}, l) = \mathbf{a}_{\text{spr}}(l_{\text{p}}, w_{\text{p}}, l_{\text{b1}}, l_{\text{b2}}, w_{\text{b}}, w_{\text{b2}})$
]|

comp *Mass* =
|[extvar $A, m, J, b, k_{\text{x,m}}, k_{\text{x,e}}, S_{\text{d}}, C_{\text{p}}, V_{\text{s0}}, l_{\text{p}}, w_{\text{p}}, g_{\text{x}}, w_{\text{fin}}, l_{\text{fin}}$
intvar $l_{\text{f}}, l_{\text{ov}}, g_{\text{s}}, g_{\text{su}}, g_{\text{f}}, g_{\text{fu}}, w_{\text{f}}, w_{\text{s}}, V_{\text{d}}, a_{\text{max,pi}}, a_{\text{max,ls}}, a_{\text{max,st}}, x_{\text{pull}}$
objfunc $f(A)$
confunc $\mathbf{g}_{\text{mass}}(l_{\text{p}}, w_{\text{p}}, l_{\text{f}}, l_{\text{ov}}, g_{\text{s}}, g_{\text{su}}, g_{\text{f}}, g_{\text{fu}}, g_{\text{x}}, w_{\text{f}}, w_{\text{s}})$
, $\mathbf{g}_{\text{elec}}(g_{\text{s}}, g_{\text{x}}, x_{\text{pull}}, k_{\text{x,e}}, k_{\text{x,m}}, a_{\text{max,pi}}, a_{\text{max,ls}}, a_{\text{max,st}})$

resfunc $(A, m, J, b, w_{\text{fin}}, l_{\text{fin}}) = \mathbf{a}_{\text{mass}}(l_{\text{p}}, w_{\text{p}}, l_{\text{f}}, l_{\text{ov}}, g_{\text{s}}, g_{\text{su}}, g_{\text{f}}, g_{\text{fu}}, g_{\text{x}}, w_{\text{f}}, w_{\text{s}})$
, $(S_{\text{d}}, k_{\text{x,e}}, a_{\text{max,pi}}, a_{\text{max,ls}}, a_{\text{max,st}}, x_{\text{pull}}, C_{\text{p}})$
$= \mathbf{a}_{\text{elec}}(m, k_{\text{x,m}}, l_{\text{p}}, w_{\text{p}}, l_{\text{f}}, l_{\text{ov}}, g_{\text{s}}, g_{\text{f}}, g_{\text{x}}, w_{\text{f}}, w_{\text{s}}, V_{\text{s0}}, V_{\text{d}})$
]|

syst *Accelerometer*$_3$ =
|[sub C: *Circuit*, D: *Dynamics*, S: *Spring*, M: *Mass*
confunc $\mathbf{g}_{\text{area}}(M.g_{\text{x}}, S.l_{\text{b2}}, S.l, S.w_{\text{spr}}, M.l_{\text{fin}}, M.w_{\text{fin}}, M.A)$
link $S.k_{\text{x,m}}$ -- $\{D.k_{\text{x,m}},\ M.k_{\text{x,m}}\}$, $C.V_{\text{s0}}$ -- $M.V_{\text{s0}}$
, $C.S_{\text{m}}$ -- $D.S_{\text{m}}$, $C.a_{\text{n,m}}$ -- $D.a_{\text{n,m}}$
, $C.S_{\text{d}}$ -- $M.S_{\text{d}}$, $D.k_{\text{x,e}}$ -- $M.k_{\text{x,e}}$
, $C.C_{\text{p}}$ -- $M.C_{\text{p}}$, $D.k_{\theta}$ -- $S.k_{\theta}$
, $S.w_{\text{p}}$ -- $M.w_{\text{p}}$, $D.k_{\text{y}}$ -- $S.k_{\text{y}}$
, $S.l_{\text{p}}$ -- $M.l_{\text{p}}$, $D.J$ -- $M.J$
, $D.m$ -- $M.m$, $D.b$ -- $M.b$
]|

topsyst *Accelerometer*$_3$

The fourth and final partition is similar to the third, but treats the functions $\mathbf{g}_{\text{circ}}$, $\mathbf{g}_{\text{device}}$, and $\mathbf{a}_{\text{device}}$ as coupling functions, rather than as local functions. The specifications for *Dynamics*, *Spring*, and *Mass* of the previous partition can be used, and the alternative *Circuit*$_2$ component and *Accelerometer*$_4$ system are given below.

comp *Circuit*$_2$ =
|[extvar $S_{\text{c}}, C_{\text{a}}, V_{\text{s0}}$
intvar G_{ni}
resfunc $S_{\text{c}} = \mathbf{a}_{\text{circ}}(V_{\text{s0}}, C_{\text{a}}, G_{\text{ni}})$
]|

syst *Accelerometer*$_4$ =
|[intvar $S, a_{\text{n}}, a_{\text{fs}}$
sub C: *Circuit*$_2$, D: *Dynamics*, S: *Spring*, M: *Mass*
confunc $\mathbf{g}_{\text{area}}(M.g_{\text{x}}, S.l_{\text{b2}}, S.l, S.w_{\text{spr}}, M.l_{\text{fin}}, M.w_{\text{fin}}, M.A)$
, $\mathbf{g}_{\text{device}}(S, a_{\text{n}}, a_{\text{fs}})$
, $\mathbf{g}_{\text{circ}}(C.C_{\text{a}}, M.C_{\text{p}})$
resfunc $(S, a_{\text{n}}, a_{\text{fs}}) = \mathbf{a}_{\text{device}}(D.S_{\text{m}}, M.S_{\text{d}}, C.S_{\text{c}}, D.a_{\text{n,m}}, C.V_{\text{s0}})$
link $S.k_{\text{x,m}}$ -- $\{D.k_{\text{x,m}},\ M.k_{\text{x,m}}\}$, $D.k_{\text{x,e}}$ -- $M.k_{\text{x,e}}$
, $S.w_{\text{p}}$ -- $M.w_{\text{p}}$, $D.k_{\theta}$ -- $S.k_{\theta}$
, $S.l_{\text{p}}$ -- $M.l_{\text{p}}$, $D.k_{\text{y}}$ -- $S.k_{\text{y}}$
, $D.m$ -- $M.m$, $D.J$ -- $M.J$
, $D.b$ -- $M.b$, $C.V_{\text{s0}}$ -- $M.V_{\text{s0}}$
]|

topsyst *Accelerometer*$_4$

Bibliography

[1] J. S. Agte, J. Sobieszczanski-Sobieski, and R. R. Jr. Sandusky. Supersonic business jet design through bi-level integrated system synthesis. In *Proceedings of the World Aviation Conference*, San Fransisco, CA, 1999. MCB Press, SAE paper 1999-01-5622.

[2] Analog Devices Inc. ADXL150/ADXL250 data sheet, 1998.

[3] Cloanto. Cloanto implementation of INI file format, 2009. http://www.cloanto.com/specs/ini.html. Date accessed: 30 January 2009.

[4] A. J. de Wit and F. van Keulen. Framework for multilevel optimization. In *Proceedings of the 5th China-Japan-Korea Joint Symposium on Optimization of Structural and Mechanical Systems*, Seoul, South Korea, 16 - 19 June 2008.

[5] L. F. P. Etman, M. Kokkolaras, A. T. Hofkamp, P. Y. Papalambros, and J. E. Rooda. Coordination specification in distributed optimal design of multilevel systems using the χ language. *Structural and Multidisciplinary Optimization*, 29(3):198–212, 2005.

[6] R. Fourer, D.M. Gay, and B. W. Kernighan. *AMPL: A Modeling Language for Mathematical Programming*. Duxbury Press, 1993.

[7] J. Golinski. Optimal synthesis problems solved by means of nonlinear programming and random methods. *Journal of Mechanisms*, 5:287–309, 1970.

[8] G. Q. Huang, T. Qu, and W. L. Cheung. Extensible multi-agent system for optimal design of complex systems using analytical target cascading. *International Journal of Advanced Manufacturing Technology*, 30:917–926, 2006.

[9] H. M. Kim. *Target Cascading in Optimal System Design*. PhD thesis, University of Michigan, 2001.

[10] H. M. Kim, N. F. Michelena, P. Y. Papalambros, and T. Jiang. Target cascading in optimal system design. *ASME Journal of Mechanical Design*, 125(3):474–480, 2003.

[11] A. Kusiak and N. Larson. Decomposition and representation methods in mechanical design. *Journal of Mechnical Design*, 117(3):17–24, 1995.

[12] J. R. R. A. Martins and C. Marriage. An object-oriented framework for multidisciplinary design optimization. In *Proceedings of the 3rd AIAA Multidisciplinary Design Optimization Specialist Conference*, Waikiki, HI, 23 - 26 April 2007.

[13] N. F. Michelena, C. Scheffer, R. Fellini, and P. Y. Papalambros. COBRA-based object-oriented framework for distributed system design. *Mechanics Based Design of Structures and Machines*, 27(4):365–392, 1999.

[14] K. T. Moore, B. A. Naylor, and J. S. Gray. The development of an open source framework for multidisciplinary analysis and optimization. In *Proceedings of the 12th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference*, Victoria, BC, Canada, 10-12 September 2008. AIAA paper 2008-6069.

[15] S. L. Padula, N. M. Alexandrov, and L. L. Green. MDO test suite at nasa langley research center. In *Proceedings of the 6th AIAA/USAF/NASA/ISSMO Multidisciplinary Analysis and Optimization Conference*, Bellevue, WA, 4-6 September 1996. AIAA paper 1996-4028. Website http://www.eng.buffalo.edu/Research/MODEL/mdotestsuite.html.

[16] H. Samuels. Single- and dual-axis micromachined accelerometers ADXL150 & ADXL250: New complete low-noise 50-g accelerometers. *Analog Dialogue*, 30(4), 1996.

[17] J. Sobieszczanski-Sobieski, T. D. Altus, M. Phillips, and R. R. Sandusky Jr. Bilevel integrated system synthesis for concurrent and distributed processing. *AIAA Journal*, 41(10):1996–2003, 2003.

[18] J. Sobieszczanski-Sobieski, B. B. James, and A. R. Dovi. Structural optimization by multilevel decomposition. AIAA paper 1983-0832, May 1983.

[19] S. Tosserams, L. F. P. Etman, P. Y. Papalambros, and J. E. Rooda. An augmented Lagrangian relaxation for analytical target cascading using the alternating direction method of multipliers. *Structural and Multidisciplinary Optimization*, 31(3):176–189, 2006.

[20] S. Tosserams, L. F. P. Etman, and J. E. Rooda. An augmented Lagrangian decomposition method for quasi-separable problems in MDO. *Structural and Multidisciplinary Optimization*, 34(3):211–227, 2007.

[21] S. Tosserams, L. F. P. Etman, and J. E. Rooda. A micro-accelerometer mdo benchmark problem. In *Proceedings of the 12th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference*, Victoria, BC, Canada, 10-12 September 2008. AIAA paper 2008-6008.

[22] S. Tosserams, A. T. Hofkamp, L. F. P. Etman, and J. E. Rooda. A partition specification language for decomposition-based system optimization in engineering design. *Structural and Multidisciplinary Optimization*, 2009. submitted.

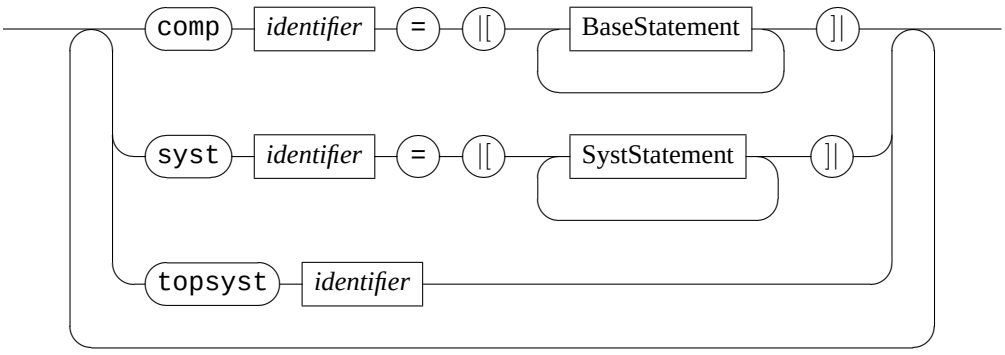
[23] Wikipedia. INI file, 2009. http://en.wikipedia.org/wiki/INI_file. Date accessed: 30 January 2009.

Appendix A
Grammar

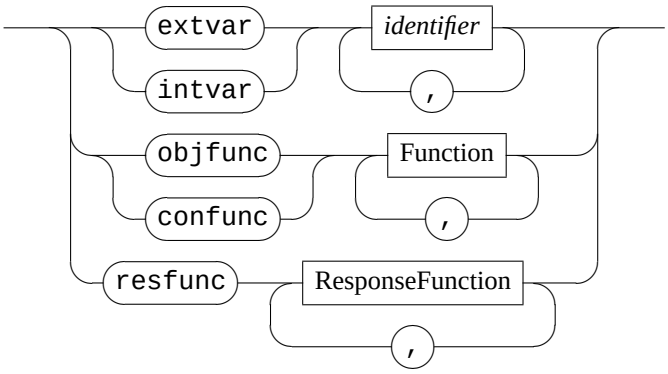
The grammar of the partition specification language Ψ is described as a number of rail road diagrams. A diagram is read starting from the left, following the lines like a train (that is, no sharp turns), until the right of the diagram is reached. Along the way, rounded boxes with text denote fixed literals that should match with the input text literally. Rectangular boxes with italic text denote non-fixed literals. In the language there are two such boxes, namely *identifier* which should be replaced by a name (a letter or underscore character, followed by zero or more letters, digits, or underscore characters), and *subsystem_identifier* which should be replaced by a name, a literal dot character, and another name. Other rectangular boxes (with text in roman font) refer to other rail road diagrams that should be inserted at that point in the original diagram. Between boxes white space may be added.

To obtain a syntactically valid specification, start with the 'Specification' rail road diagram below.

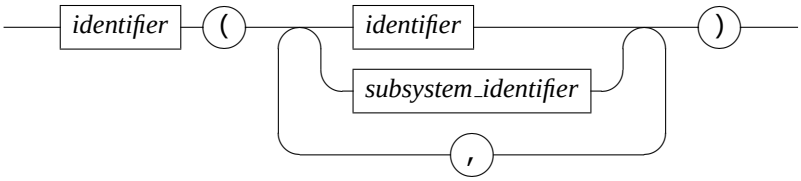
Specification



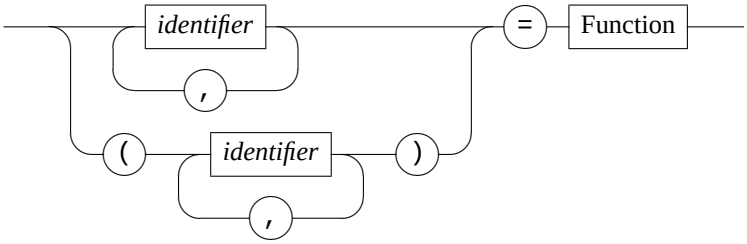
BaseStatement



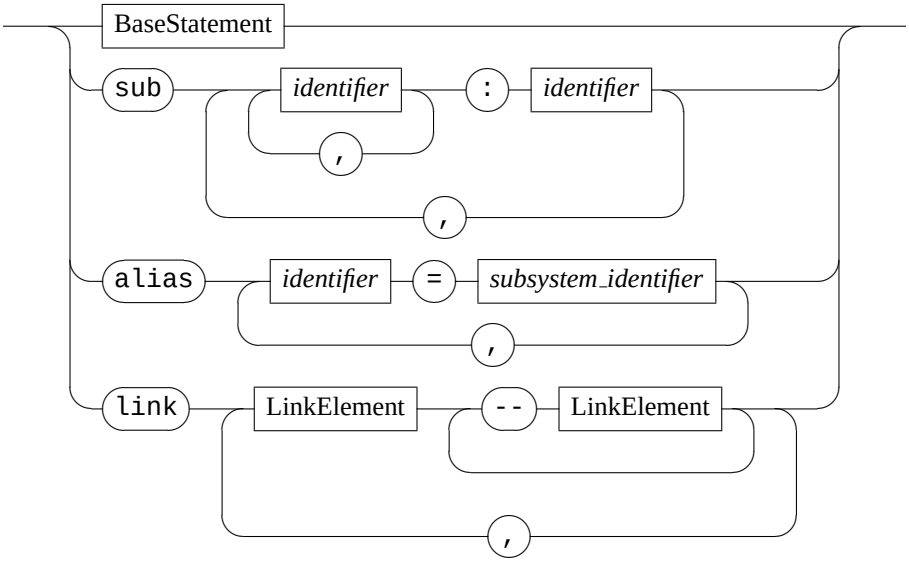
Function



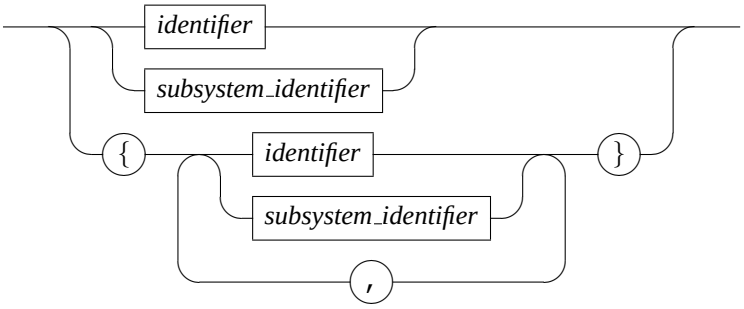
ResponseFunction



SystStatement



LinkElement



Appendix B
Requirements

This section defines the static semantic requirements for specifications, components, systems, and the topsyst statement. Each requirement has a unique identification stated in bold before the text of the requirement. These identifications are also reported by the error checker tool.

1 Specification

f1 Every component and system must be (instantiated) in the specification.

2 Components

The static semantics requirements for components are listed below. The requirements are split in several topics that cover the different areas of the component specification.

Component name

p1 Component name must be unique in the specification.

External variables

e1 External variable names must be unique within the component.

e2 At least one external variable must exist in the definition.

Internal variables

i1 Internal variable names must be unique within the component.

i2 Each internal variable must be used as an argument of a function.

Objective functions

o1 The names of objective functions may be chosen arbitrarily.

o2 Each objective function should have at least one actual parameter.

o3 Each actual parameter of an objective function is either an internal or an external variable of the component.

Constraint functions

c1 The names of constraint functions may be chosen arbitrarily.

c2 Each constraint function should have at least one actual parameter.

c3 Each actual parameter of a constraint function is either an internal or an external variable of the component.

Response functions

r1 The names of response functions may be chosen arbitrarily.

r2 Each response function should have at least one actual parameter.

r3 Each actual parameter of a response function is either an internal or an external variable of the component.

r4 Variables used as output of the response function may not be used as actual parameter of the response function.

r5 A variable may be used as (part of the) output of a response function once.

3 Systems

The static semantics requirements for systems are listed below. The requirements are split in several topics that cover the different areas of the system specification.

System name

P1 System name must be unique in the specification.

External variables

E1 External variable names must be unique within the component.

E2 Each external variable should be used as an output of a response function.

Internal variables

I1 Internal variable names must be unique within the component.

I2 Each internal variable should be used as an output of a response function.

Sub-components

S1 Only component or system definitions defined elsewhere in the specification may be used as sub-components.

S2 At least two sub-components should be instantiated.

S3 A system cannot instantiate itself directly or indirectly.

S4 Names of instantiated sub-components must be unique within the definition.

For sub-components that are components:

X1 Each external variable of the sub-component must be used in an objective, constraint, or response function, in an alias, or in a link.

For sub-components that are systems:

X2 The sub-component must have at least one alias or external variable.

X3 Each alias or external variable of the sub-component must be used in an objective, constraint, or response function, in an alias, or in a link.

Objective functions

O1 The names of objective functions may be chosen arbitrarily.

O2 The actual parameters of each objective function must be from at least two different sub-components, or must contain at least one local variable.

O3 Each actual parameter of an objective function must be an external variable from a sub-component, an alias from a sub-component, or a local variable.

Constraint functions

C1 The names of constraint functions may be chosen arbitrarily.

C2 The actual parameters of each constraint function must be from at least two different sub-components, or must contain at least one local variable.

C3 Each actual parameter of a constraint function must be an external variable from a sub-component, an alias from a sub-component, or a local variable.

Response functions

R1 The names of response functions may be chosen arbitrarily.

R2 Variables used as output of the response function may not be used as actual parameter of the response function.

R3 The actual parameters of each response function must be from at least two different sub-components, or must contain at least one local variable.

R4 Each actual parameter of a response function must be an external variable from a sub-component, an alias from a sub-component, or a local variable.

R5 A variable may be used as (part of the) output of a response function once.

Aliases

A1 Names of aliases must be unique within the system.

A2 An alias must be coupled to an external variable or alias of a sub-component.

Sub-component identifiers

D1 The module part of each sub-component identifier must exist as name of an instantiated sub-component.

D2 The name part of each sub-component identifier must exist as name of an external variable or alias of the sub-component that the module part (of the same sub-component identifier) refers to.

Links

L1 Couplings should not form a cycle.

L2 Both sides of a link must refer to variables or aliases, and exist within the definition.

L3 Coupling may only take place between variables or aliases of two different sub-components, or between a local variable and a variable or alias of a sub-component.

4 Topsyst

The static semantics requirements of the '`topsyst`' statement are

T1 There must be exactly one '`topsyst`' statement in the specification.

T2 It must identify a system as top-level element.

T3 The system it identifies as top-level element must be defined in the file.

T4 The system it identifies as top-level element may not have aliases or external variables.

Appendix C
Specification and generated outputs for examples

This appendix includes the actual input files and generated outputs for the examples presented in the user manual.

1 Example (3.1)

Partition specification in Ψ:

```
comp First =
|[  extvar x2,y,r
    intvar x1
    objfunc f1(x1,x2,y)
    confunc g1(x1,y)
    resfunc r = a1(x2,y)
]|

comp Second =
|[  extvar x3,y,r
    intvar x4,u
    objfunc f2(x3,x4,y)
    confunc g2(x3,x4,y,u)
    resfunc u = a2(x3,x4,r)
]|

syst Problem =
|[  sub A: First, B: Second
    link A.y -- B.y, A.r -- B.r
    objfunc f0(A.x2,B.x3)
```

```
]|

topsyst Problem
```

Generated normalized partition:

```
[topsyst]
system = syst_1

[func_5]
path = Problem.B
argvars = var_7, var_5, var_8, var_6
name = g2
defined_in = Second

[func_4]
path = Problem.B
argvars = var_7, var_5, var_8
name = f2
defined_in = Second

[func_7]
path = Problem
argvars = var_2, var_7
name = f0
defined_in = Problem

[func_6]
path = Problem.B
resvars = var_6
argvars = var_7, var_5, var_9
name = a2
defined_in = Second

[func_1]
path = Problem.A
argvars = var_1, var_2, var_3
name = f1
defined_in = First

[func_3]
path = Problem.A
resvars = var_4
argvars = var_2, var_3
name = a1
defined_in = First

[func_2]
path = Problem.A
argvars = var_1, var_3
name = g1
defined_in = First

[var_9]
path = Problem.B
name = r
defined_in = Second

[var_8]
path = Problem.B
name = y
defined_in = Second

[var_7]
path = Problem.B
name = x3
defined_in = Second

[var_6]
path = Problem.B
name = u
defined_in = Second

[var_5]
path = Problem.B
name = x4
defined_in = Second

[var_4]
path = Problem.A
name = r
defined_in = First

[var_3]
path = Problem.A
name = y
defined_in = First

[var_2]
path = Problem.A
name = x2
defined_in = First

[var_1]
path = Problem.A
name = x1
defined_in = First

[syst_1]
subs = comp_1, comp_2
links = link_1, link_2
objfuncs = func_7
path = Problem
type = Problem
name = Problem

[comp_2]
resfuncs = func_6
name = B
local_resvars = var_6
objfuncs = func_4
local_vars = var_7, var_5
coupling_vars = var_9, var_8
confuncs = func_5
path = Problem.B
type = Second

[comp_1]
resfuncs = func_3
name = A
objfuncs = func_1
coupling_resvars = var_4
local_vars = var_2, var_1
coupling_vars = var_3
confuncs = func_2
path = Problem.A
type = First

[link_2]
defined_in = syst_1
coupling = var_4, var_9
path = Problem

[link_1]
defined_in = syst_1
coupling = var_3, var_8
path = Problem
```

Functional dependence table:

```
Columns:
 1: Problem.A.x2
 2: Problem.B.x3
 3: Problem.A.y
 4: Problem.B.y
 5: Problem.A.r
 6: Problem.B.r
 7: Problem.A.x1
 8: Problem.B.x4
 9: Problem.B.u

Rows:
 1: Problem.f0
 2: Problem.A.y -- Problem.B.y
 3: Problem.A.r -- Problem.B.r
 4: Problem.A.f1
 5: Problem.A.g1
 6: Problem.A.a1
 7: Problem.B.f2
 8: Problem.B.g2
 9: Problem.B.a2

FDT:
1, 1, 0, 0, 0, 0, 0, 0, 0
0, 0, 1, 1, 0, 0, 0, 0, 0
0, 0, 0, 0, 1, 1, 0, 0, 0
1, 0, 1, 0, 0, 0, 1, 0, 0
0, 0, 1, 0, 0, 0, 1, 0, 0
1, 0, 1, 0, 1, 0, 0, 0, 0
```

```
0, 1, 0, 1, 0, 0, 0, 1, 0
0, 1, 0, 1, 0, 0, 0, 1, 1
0, 1, 0, 0, 0, 1, 0, 1, 1
```

2 Geometric programming problem

First partition

Partition specification in Ψ:

```
comp First =
|[  extvar z3, z6
    intvar z1, z2, z4, z5, z7
    objfunc f(z1)
          , f(z2)
    confunc g1(z3,z4,z5)
          , g2(z5,z6,z7)
          , h1(z1,z3,z4,z5)
          , h2(z2,z5,z6,z7)
]|

comp Second =
|[  extvar  z3, z6
    intvar  z8, z9, z10, z11, z12, z13, z14
    confunc g3(z8,z9,z11)
          , g4(z8,z10,z11)
          , g5(z11,z12,z13)
          , g6(z11,z12,z14)
          , h3(z3,z8,z9,z10,z11)
          , h4(z6,z11,z12,z13,z14)
]|

syst Geo2a =
|[  sub A: First, B: Second
    link A.z3 -- B.z3, A.z6 -- B.z6
]|

topsyst Geo2a
```

Generated normalized partition:

```
[topsyst]
system = syst_1

[func_9]
path = Geo2a.B
argvars = var_11, var_12, var_13
name = g5
defined_in = Second

[func_8]
path = Geo2a.B
argvars = var_8, var_10, var_11
name = g4
defined_in = Second
```

```
[func_5]
path = Geo2a.A
argvars = var_1, var_6, var_3, var_4
name = h1
defined_in = First

[func_4]
path = Geo2a.A
argvars = var_4, var_7, var_5
name = g2
defined_in = First

[func_7]
path = Geo2a.B
argvars = var_8, var_9, var_11
name = g3
defined_in = Second

[func_6]
path = Geo2a.A
argvars = var_2, var_4, var_7, var_5
name = h2
defined_in = First

[func_1]
path = Geo2a.A
argvars = var_1
name = f
defined_in = First

[func_3]
path = Geo2a.A
argvars = var_6, var_3, var_4
name = g1
defined_in = First

[func_2]
path = Geo2a.A
argvars = var_2
name = f
defined_in = First

[func_11]
path = Geo2a.B
argvars = var_15, var_8, var_9, var_10, var_11
name = h3
defined_in = Second

[func_10]
path = Geo2a.B
argvars = var_11, var_12, var_14
name = g6
defined_in = Second

[func_12]
path = Geo2a.B
argvars = var_16, var_11, var_12, var_13, var_14
name = h4
defined_in = Second

[var_9]
path = Geo2a.B
name = z9
defined_in = Second

[var_8]
path = Geo2a.B
name = z8
defined_in = Second

[var_7]
path = Geo2a.A
name = z6
defined_in = First

[var_6]
path = Geo2a.A
name = z3
defined_in = First

[var_5]
path = Geo2a.A
name = z7
defined_in = First

[var_4]
path = Geo2a.A
name = z5
defined_in = First

[var_3]
path = Geo2a.A
name = z4
defined_in = First

[var_2]
path = Geo2a.A
name = z2
defined_in = First

[var_1]
path = Geo2a.A
name = z1
defined_in = First

[var_16]
path = Geo2a.B
name = z6
defined_in = Second

[var_15]
path = Geo2a.B
name = z3
defined_in = Second

[var_14]
path = Geo2a.B
name = z14
defined_in = Second

[var_13]
path = Geo2a.B
name = z13
defined_in = Second

[var_12]
path = Geo2a.B
name = z12
defined_in = Second

[var_11]
path = Geo2a.B
name = z11
defined_in = Second

[var_10]
path = Geo2a.B
name = z10
defined_in = Second

[comp_2]
name = B
coupling_vars = var_16, var_15
local_vars = var_14, var_13, var_12, var_11, var_10, var_9, var_8
path = Geo2a.B
confuncs = func_7, func_8, func_9, func_10, func_11, func_12
type = Second

[syst_1]
path = Geo2a
type = Geo2a
name = Geo2a
links = link_1, link_2
subs = comp_1, comp_2

[comp_1]
name = A
coupling_vars = var_7, var_6
objfuncs = func_1, func_2
local_vars = var_5, var_4, var_3, var_2, var_1
confuncs = func_3, func_4, func_5, func_6
path = Geo2a.A
type = First

[link_2]
defined_in = syst_1
coupling = var_7, var_16
path = Geo2a

[link_1]
defined_in = syst_1
coupling = var_6, var_15
path = Geo2a
```

Functional dependence table:

```
Columns:
 1: Geo2a.A.z3
 2: Geo2a.B.z3
```

```
 3: Geo2a.A.z6
 4: Geo2a.B.z6
 5: Geo2a.A.z1
 6: Geo2a.A.z2
 7: Geo2a.A.z4
 8: Geo2a.A.z5
 9: Geo2a.A.z7
10: Geo2a.B.z8
11: Geo2a.B.z9
12: Geo2a.B.z11
13: Geo2a.B.z10
14: Geo2a.B.z12
15: Geo2a.B.z13
16: Geo2a.B.z14

Rows:
 1: Geo2a.A.z3 -- Geo2a.B.z3
 2: Geo2a.A.z6 -- Geo2a.B.z6
 3: Geo2a.A.f
 4: Geo2a.A.f
 5: Geo2a.A.g1
 6: Geo2a.A.g2
 7: Geo2a.A.h1
 8: Geo2a.A.h2
 9: Geo2a.B.g3
10: Geo2a.B.g4
11: Geo2a.B.g5
12: Geo2a.B.g6
13: Geo2a.B.h3
14: Geo2a.B.h4

FDT:
1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
0, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
1, 0, 0, 0, 0, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0
0, 0, 1, 0, 0, 0, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0
1, 0, 0, 0, 1, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0
0, 0, 1, 0, 0, 1, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0
0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 1, 0, 0, 0, 0
0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 1, 1, 0, 0, 0
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 1, 1, 0
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 1, 0, 1
0, 1, 0, 0, 0, 0, 0, 0, 0, 1, 1, 1, 1, 0, 0, 0
0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 1, 0, 1, 1, 1
```

Second partition

Partition specification in Ψ:

```
comp First =
|[  extvar z3, z6
    intvar z1, z2, z4, z5, z7
    objfunc f(z1,z2)
    confunc g1(z3,z4,z5)
          , g2(z5,z6,z7)
          , h1(z1,z3,z4,z5)
          , h2(z2,z5,z6,z7)
]|

comp Second1 =
|[  extvar z3, z11
    intvar z8, z9, z10
    confunc g3(z8,z9,z11)
          , g4(z8,z10,z11)
          , h3(z3,z8,z9,z10,z11)
]|

comp Second2 =
|[  extvar z6, z11
```

```
    intvar z12, z13, z14
    confunc g5(z11,z12,z13)
          , g6(z11,z12,z14)
          , h4(z6,z11,z12,z13,z14)
]|

syst Geo2b =
|[  sub A: First, B1: Second1, B2: Second2
    link A.z3 -- B1.z3, A.z6 -- B2.z6, B1.z11 -- B2.z11
]|

topsyst Geo2b
```

Generated normalized partition:

```
[topsyst]
system = syst_1

[func_9]
path = Geo2b.B2
argvars = var_17, var_13, var_14
name = g5
defined_in = Second2

[func_8]
path = Geo2b.B1
argvars = var_11, var_8, var_9, var_10, var_12
name = h3
defined_in = Second1

[func_5]
path = Geo2b.A
argvars = var_2, var_4, var_7, var_5
name = h2
defined_in = First

[func_4]
path = Geo2b.A
argvars = var_1, var_6, var_3, var_4
name = h1
defined_in = First

[func_7]
path = Geo2b.B1
argvars = var_8, var_10, var_12
name = g4
defined_in = Second1

[func_6]
path = Geo2b.B1
argvars = var_8, var_9, var_12
name = g3
defined_in = Second1

[func_1]
path = Geo2b.A
argvars = var_1, var_2
name = f
defined_in = First

[func_3]
path = Geo2b.A
argvars = var_4, var_7, var_5
name = g2
defined_in = First

[func_2]
path = Geo2b.A
argvars = var_6, var_3, var_4
name = g1
defined_in = First

[func_11]
path = Geo2b.B2
argvars = var_16, var_17, var_13, var_14, var_15
name = h4
defined_in = Second2

[func_10]
path = Geo2b.B2
argvars = var_17, var_13, var_15
name = g6
defined_in = Second2

[var_9]
path = Geo2b.B1
name = z9
defined_in = Second1

[var_8]
path = Geo2b.B1
name = z8
defined_in = Second1

[var_7]
path = Geo2b.A
name = z6
defined_in = First

[var_6]
path = Geo2b.A
name = z3
defined_in = First

[var_5]
path = Geo2b.A
name = z7
defined_in = First

[var_4]
path = Geo2b.A
name = z5
defined_in = First

[var_3]
path = Geo2b.A
name = z4
defined_in = First

[var_2]
path = Geo2b.A
name = z2
defined_in = First

[var_1]
path = Geo2b.A
name = z1
defined_in = First

[var_17]
path = Geo2b.B2
name = z11
defined_in = Second2

[var_16]
path = Geo2b.B2
name = z6
defined_in = Second2

[var_15]
path = Geo2b.B2
name = z14
defined_in = Second2

[var_14]
path = Geo2b.B2
name = z13
defined_in = Second2

[var_13]
path = Geo2b.B2
name = z12
defined_in = Second2

[var_12]
```

```
path = Geo2b.B1
name = z11
defined_in = Second1

[var_11]
path = Geo2b.B1
name = z3
defined_in = Second1

[var_10]
path = Geo2b.B1
name = z10
defined_in = Second1

[comp_2]
name = B1
coupling_vars = var_12, var_11
local_vars = var_10, var_9, var_8
path = Geo2b.B1
confuncs = func_6, func_7, func_8
type = Second1

[comp_3]
name = B2
coupling_vars = var_17, var_16
local_vars = var_15, var_14, var_13
path = Geo2b.B2
confuncs = func_9, func_10, func_11
type = Second2

[syst_1]
```

```
path = Geo2b
type = Geo2b
name = Geo2b
links = link_1, link_2, link_3
subs = comp_1, comp_2, comp_3

[comp_1]
name = A
coupling_vars = var_7, var_6
objfuncs = func_1
local_vars = var_5, var_4, var_3, var_2, var_1
confuncs = func_2, func_3, func_4, func_5
path = Geo2b.A
type = First

[link_3]
defined_in = syst_1
coupling = var_12, var_17
path = Geo2b

[link_2]
defined_in = syst_1
coupling = var_7, var_16
path = Geo2b

[link_1]
defined_in = syst_1
coupling = var_6, var_11
path = Geo2b
```

Functional dependence table:

```
Columns:
 1: Geo2b.A.z3
 2: Geo2b.B1.z3
 3: Geo2b.A.z6
 4: Geo2b.B2.z6
 5: Geo2b.B1.z11
 6: Geo2b.B2.z11
 7: Geo2b.A.z1
 8: Geo2b.A.z2
 9: Geo2b.A.z4
10: Geo2b.A.z5
11: Geo2b.A.z7
12: Geo2b.B1.z8
13: Geo2b.B1.z9
14: Geo2b.B1.z10
15: Geo2b.B2.z12
16: Geo2b.B2.z13
17: Geo2b.B2.z14

Rows:
 1: Geo2b.A.z3 -- Geo2b.B1.z3
 2: Geo2b.A.z6 -- Geo2b.B2.z6
 3: Geo2b.B1.z11 -- Geo2b.B2.z11
 4: Geo2b.A.f
 5: Geo2b.A.g1
 6: Geo2b.A.g2
 7: Geo2b.A.h1
 8: Geo2b.A.h2
 9: Geo2b.B1.g3
10: Geo2b.B1.g4
11: Geo2b.B1.h3
12: Geo2b.B2.g5
13: Geo2b.B2.g6
14: Geo2b.B2.h4

FDT:
1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
0, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
0, 0, 0, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0
1, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0
0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 0, 0, 0
1, 0, 0, 0, 0, 0, 1, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0
0, 0, 1, 0, 0, 0, 0, 1, 0, 1, 1, 0, 0, 0, 0, 0, 0
0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 0, 0
0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 1, 0, 1, 0, 0, 0
0, 1, 0, 0, 1, 0, 0, 0, 0, 0, 0, 1, 1, 1, 0, 0, 0
0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0
0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 1
0, 0, 0, 1, 0, 1, 0, 0, 0, 0, 0, 0, 0, 1, 1, 1
```

Third partition

Partition specification in Ψ:

```
comp First1 =
|[  extvar z3, z5
    intvar z1, z4
    objfunc f(z1)
    confunc g1(z3,z4,z5)
          , h1(z1,z3,z4,z5)
]|

comp First2 =
|[  extvar z5, z6
    intvar z2, z7
    objfunc f(z2)
    confunc g2(z5,z6,z7)
          , h2(z2,z5,z6,z7)
]|

comp Second1 =
|[  extvar z3, z11
    intvar z8, z9, z10
    confunc g3(z8,z9,z11)
          , g4(z8,z10,z11)
          , h3(z3,z8,z9,z10,z11)
]|

comp Second2 =
|[  extvar z6, z11
    intvar z12, z13, z14
    confunc g5(z11,z12,z13)
          , g6(z11,z12,z14)
          , h4(z6,z11,z12,z13,z14)
]|

syst Geo2c =
|[  sub A1: First1, A2: First2, B1: Second1, B2: Second2
    link A1.z3 -- B1.z3, A2.z6 -- B2.z6, B1.z11 -- B2.z11, A1.z5 -- A2.z5
]|

topsyst Geo2c
```

Generated normalized partition:

```
[topsyst]
system = syst_1

[func_9]
path = Geo2c.B1
argvars = var_12, var_9, var_10, var_11, var_13
name = h3
defined_in = Second1

[func_8]
path = Geo2c.B1
argvars = var_9, var_11, var_13
name = g4
defined_in = Second1

[func_5]
path = Geo2c.A2
argvars = var_7, var_8, var_6
```

```
name = g2
defined_in = First2

[func_4]
path = Geo2c.A2
argvars = var_5
name = f
defined_in = First2

[func_7]
path = Geo2c.B1
argvars = var_9, var_10, var_13
name = g3
defined_in = Second1

[func_6]
path = Geo2c.A2
argvars = var_5, var_7, var_8, var_6
name = h2
defined_in = First2

[func_1]
path = Geo2c.A1
argvars = var_1
name = f
defined_in = First1

[func_3]
path = Geo2c.A1
argvars = var_1, var_3, var_2, var_4
name = h1
defined_in = First1

[func_2]
path = Geo2c.A1
argvars = var_3, var_2, var_4
name = g1
defined_in = First1

[func_11]
path = Geo2c.B2
argvars = var_18, var_14, var_16
name = g6
defined_in = Second2

[func_10]
path = Geo2c.B2
argvars = var_18, var_14, var_15
name = g5
defined_in = Second2

[func_12]
path = Geo2c.B2
argvars = var_17, var_18, var_14, var_15, var_16
name = h4
defined_in = Second2

[var_9]
path = Geo2c.B1
name = z8
defined_in = Second1

[var_8]
path = Geo2c.A2
name = z6
defined_in = First2

[var_7]
path = Geo2c.A2
name = z5
defined_in = First2

[var_6]
path = Geo2c.A2
name = z7
defined_in = First2

[var_5]
path = Geo2c.A2
name = z2
defined_in = First2

[var_4]
path = Geo2c.A1
name = z5
defined_in = First1

[var_3]
path = Geo2c.A1
name = z3
defined_in = First1

[var_2]
path = Geo2c.A1
name = z4
defined_in = First1

[var_1]
path = Geo2c.A1
name = z1
defined_in = First1

[var_17]
path = Geo2c.B2
name = z6
defined_in = Second2

[var_16]
path = Geo2c.B2
name = z14
defined_in = Second2

[var_15]
path = Geo2c.B2
name = z13
defined_in = Second2

[var_14]
path = Geo2c.B2
name = z12
defined_in = Second2

[var_13]
path = Geo2c.B1
name = z11
defined_in = Second1

[var_12]
path = Geo2c.B1
name = z3
defined_in = Second1

[var_11]
path = Geo2c.B1
name = z10
defined_in = Second1

[var_10]
path = Geo2c.B1
name = z9
defined_in = Second1

[var_18]
path = Geo2c.B2
name = z11
defined_in = Second2

[comp_4]
name = B2
coupling_vars = var_17, var_18
local_vars = var_16, var_15, var_14
path = Geo2c.B2
confuncs = func_10, func_11, func_12
type = Second2

[comp_2]
name = A2
coupling_vars = var_7, var_8
objfuncs = func_4
local_vars = var_6, var_5
confuncs = func_5, func_6
path = Geo2c.A2
type = First2

[comp_3]
name = B1
coupling_vars = var_13, var_12
local_vars = var_11, var_9, var_10
path = Geo2c.B1
confuncs = func_7, func_8, func_9
type = Second1

[syst_1]
path = Geo2c
type = Geo2c
name = Geo2c
links = link_1, link_2, link_3, link_4
subs = comp_1, comp_2, comp_3, comp_4

[comp_1]
name = A1
coupling_vars = var_4, var_3
objfuncs = func_1
local_vars = var_2, var_1
confuncs = func_2, func_3
path = Geo2c.A1
type = First1

[link_3]
defined_in = syst_1
coupling = var_13, var_18
path = Geo2c

[link_2]
defined_in = syst_1
coupling = var_8, var_17
path = Geo2c

[link_1]
defined_in = syst_1
coupling = var_3, var_12
path = Geo2c

[link_4]
defined_in = syst_1
coupling = var_4, var_7
```

```
path = Geo2c
```

Functional dependence table:

```
Columns:
 1: Geo2c.A1.z3
 2: Geo2c.B1.z3
 3: Geo2c.A2.z6
 4: Geo2c.B2.z6
 5: Geo2c.B1.z11
 6: Geo2c.B2.z11
 7: Geo2c.A1.z5
 8: Geo2c.A2.z5
 9: Geo2c.A1.z1
10: Geo2c.A1.z4
11: Geo2c.A2.z2
12: Geo2c.A2.z7
13: Geo2c.B1.z8
14: Geo2c.B1.z9
15: Geo2c.B1.z10
16: Geo2c.B2.z12
17: Geo2c.B2.z13
18: Geo2c.B2.z14

Rows:
 1: Geo2c.A1.z3 -- Geo2c.B1.z3
 2: Geo2c.A2.z6 -- Geo2c.B2.z6
 3: Geo2c.B1.z11 -- Geo2c.B2.z11
 4: Geo2c.A1.z5 -- Geo2c.A2.z5
 5: Geo2c.A1.f
 6: Geo2c.A1.g1
 7: Geo2c.A1.h1
 8: Geo2c.A2.f
 9: Geo2c.A2.g2
10: Geo2c.A2.h2
11: Geo2c.B1.g3
12: Geo2c.B1.g4
13: Geo2c.B1.h3
14: Geo2c.B2.g5
15: Geo2c.B2.g6
16: Geo2c.B2.h4

FDT:
1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
0, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
0, 0, 0, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0
1, 0, 0, 0, 0, 0, 1, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0
1, 0, 0, 0, 0, 0, 1, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0
0, 0, 1, 0, 0, 0, 0, 1, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0
0, 0, 1, 0, 0, 0, 0, 1, 0, 0, 1, 1, 0, 0, 0, 0, 0, 0
0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 0, 0
0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 1, 0, 1, 0, 0, 0
0, 1, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 1, 1, 1, 0, 0, 0
0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0
0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 1
0, 0, 0, 1, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 1
```

---

3 Speed reducer

First partition

Partition specification in Ψ:

```
comp Gear =
|[  extvar x1, x2, x3
    objfunc F1(x1,x2,x3)
```

```
    confunc g5(x1,x2,x3)
          , g6(x1,x2,x3)
          , g9(x2,x3)
          , g10(x1,x2)
          , g11(x1,x2)
]|

comp ShaftA =
|[  extvar x1, x2, x3
    intvar x4, x6
    objfunc F2(x1,x6)
          , F4(x6)
          , F6(x4,x6)
    confunc g1(x2,x3,x4,x6)
          , g3(x4,x6)
          , g7(x2,x3,x4,x6)
]|

comp ShaftB =
|[  extvar x1, x2, x3
    intvar x5, x7
    objfunc F3(x1,x7)
          , F5(x7)
          , F7(x5,x7)
    confunc g2(x2,x3,x5,x7)
          , g4(x5,x7)
          , g8(x2,x3,x5,x7)
]|

syst SpeedReducer =
|[  sub G: Gear, S1: ShaftA, S2: ShaftB
    link G.x1 -- {S1.x1, S2.x1}
       , G.x2 -- {S1.x2, S2.x2}
       , G.x3 -- {S1.x3, S2.x3}
]|

topsyst SpeedReducer
```

Generated normalized partition:

```
[topsyst]
system = syst_1

[func_9]
path = SpeedReducer.S1
argvars = var_4, var_5
name = F6
defined_in = ShaftA

[func_8]
path = SpeedReducer.S1
argvars = var_5
name = F4
defined_in = ShaftA

[func_5]
path = SpeedReducer.G
argvars = var_1, var_2
name = g10
defined_in = Gear

[func_4]
path = SpeedReducer.G
argvars = var_2, var_3
name = g9
defined_in = Gear

[func_7]
path = SpeedReducer.S1
argvars = var_6, var_5
name = F2
defined_in = ShaftA

[func_6]
path = SpeedReducer.G
argvars = var_1, var_2
name = g11
defined_in = Gear

[func_1]
path = SpeedReducer.G
argvars = var_1, var_2, var_3
name = F1
defined_in = Gear

[func_3]
path = SpeedReducer.G
argvars = var_1, var_2, var_3
```

```
name = g6
defined_in = Gear

[func_2]
path = SpeedReducer.G
argvars = var_1, var_2, var_3
name = g5
defined_in = Gear

[func_18]
path = SpeedReducer.S2
argvars = var_12, var_13, var_9, var_10
name = g8
defined_in = ShaftB

[func_11]
path = SpeedReducer.S1
argvars = var_4, var_5
name = g3
defined_in = ShaftA

[func_10]
path = SpeedReducer.S1
argvars = var_7, var_8, var_4, var_5
name = g1
defined_in = ShaftA

[func_13]
path = SpeedReducer.S2
argvars = var_11, var_10
name = F3
defined_in = ShaftB

[func_12]
path = SpeedReducer.S1
argvars = var_7, var_8, var_4, var_5
name = g7
defined_in = ShaftA

[func_15]
path = SpeedReducer.S2
argvars = var_9, var_10
name = F7
defined_in = ShaftB

[func_14]
path = SpeedReducer.S2
argvars = var_10
name = F5
defined_in = ShaftB

[func_17]
path = SpeedReducer.S2
argvars = var_9, var_10
name = g4
defined_in = ShaftB

[func_16]
path = SpeedReducer.S2
argvars = var_12, var_13, var_9, var_10
name = g2
defined_in = ShaftB

[var_9]
path = SpeedReducer.S2
name = x5
defined_in = ShaftB

[var_8]
path = SpeedReducer.S1
name = x3
defined_in = ShaftA

[var_7]
path = SpeedReducer.S1
name = x2
defined_in = ShaftA

[var_6]
path = SpeedReducer.S1
name = x1
defined_in = ShaftA

[var_5]
path = SpeedReducer.S1
name = x6
defined_in = ShaftA

[var_4]
path = SpeedReducer.S1
name = x4
defined_in = ShaftA

[var_3]
path = SpeedReducer.G
name = x3
defined_in = Gear

[var_2]
path = SpeedReducer.G
name = x2
defined_in = Gear

[var_1]
path = SpeedReducer.G
name = x1
defined_in = Gear

[var_13]
path = SpeedReducer.S2
name = x3
defined_in = ShaftB

[var_12]
path = SpeedReducer.S2
name = x2
defined_in = ShaftB

[var_11]
path = SpeedReducer.S2
name = x1
defined_in = ShaftB

[var_10]
path = SpeedReducer.S2
name = x7
defined_in = ShaftB

[comp_2]
name = S1
coupling_vars = var_7, var_6, var_8
objfuncs = func_7, func_8, func_9
local_vars = var_5, var_4
confuncs = func_10, func_11, func_12
path = SpeedReducer.S1
type = ShaftA

[comp_3]
name = S2
coupling_vars = var_13, var_12, var_11
objfuncs = func_13, func_14, func_15
local_vars = var_9, var_10
confuncs = func_16, func_17, func_18
path = SpeedReducer.S2
type = ShaftB

[syst_1]
path = SpeedReducer
type = SpeedReducer
name = SpeedReducer
links = link_1, link_2, link_3, link_4, link_5, link_6
subs = comp_1, comp_2, comp_3

[comp_1]
name = G
coupling_vars = var_3, var_2, var_1
objfuncs = func_1
path = SpeedReducer.G
confuncs = func_2, func_3, func_4, func_5, func_6
type = Gear

[link_3]
defined_in = syst_1
coupling = var_2, var_7
path = SpeedReducer

[link_2]
defined_in = syst_1
coupling = var_1, var_11
path = SpeedReducer

[link_1]
defined_in = syst_1
coupling = var_1, var_6
path = SpeedReducer

[link_6]
defined_in = syst_1
coupling = var_3, var_8
path = SpeedReducer

[link_5]
defined_in = syst_1
coupling = var_3, var_13
path = SpeedReducer

[link_4]
defined_in = syst_1
coupling = var_2, var_12
path = SpeedReducer
```

Functional dependence table:

```
Columns:
 1: SpeedReducer.G.x1
 2: SpeedReducer.S1.x1
 3: SpeedReducer.S2.x1
 4: SpeedReducer.G.x2
 5: SpeedReducer.S1.x2
 6: SpeedReducer.S2.x2
 7: SpeedReducer.G.x3
 8: SpeedReducer.S2.x3
 9: SpeedReducer.S1.x3
10: SpeedReducer.S1.x6
11: SpeedReducer.S1.x4
12: SpeedReducer.S2.x7
13: SpeedReducer.S2.x5

Rows:
 1: SpeedReducer.G.x1 -- SpeedReducer.S1.x1
 2: SpeedReducer.G.x1 -- SpeedReducer.S2.x1
 3: SpeedReducer.G.x2 -- SpeedReducer.S1.x2
 4: SpeedReducer.G.x2 -- SpeedReducer.S2.x2
 5: SpeedReducer.G.x3 -- SpeedReducer.S2.x3
 6: SpeedReducer.G.x3 -- SpeedReducer.S1.x3
 7: SpeedReducer.G.F1
 8: SpeedReducer.G.g5
 9: SpeedReducer.G.g6
10: SpeedReducer.G.g9
11: SpeedReducer.G.g10
12: SpeedReducer.G.g11
13: SpeedReducer.S1.F2
14: SpeedReducer.S1.F4
15: SpeedReducer.S1.F6
16: SpeedReducer.S1.g1
17: SpeedReducer.S1.g3
18: SpeedReducer.S1.g7
19: SpeedReducer.S2.F3
20: SpeedReducer.S2.F5
21: SpeedReducer.S2.F7
22: SpeedReducer.S2.g2
23: SpeedReducer.S2.g4
24: SpeedReducer.S2.g8

FDT:
1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
1, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
0, 0, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0
0, 0, 0, 1, 0, 1, 0, 0, 0, 0, 0, 0, 0
0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 0, 0, 0
0, 0, 0, 0, 0, 0, 1, 0, 1, 0, 0, 0, 0
1, 0, 0, 1, 0, 0, 1, 0, 0, 0, 0, 0, 0
1, 0, 0, 1, 0, 0, 1, 0, 0, 0, 0, 0, 0
1, 0, 0, 1, 0, 0, 1, 0, 0, 0, 0, 0, 0
0, 0, 0, 1, 0, 0, 1, 0, 0, 0, 0, 0, 0
1, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0
1, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0
0, 1, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0
0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0
0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0
0, 0, 0, 0, 1, 0, 0, 0, 1, 1, 1, 0, 0
0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0
0, 0, 0, 0, 1, 0, 0, 0, 1, 1, 1, 0, 0
0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1
0, 0, 0, 0, 0, 1, 0, 1, 0, 0, 0, 1, 1
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1
0, 0, 0, 0, 0, 1, 0, 1, 0, 0, 0, 1, 1
```

Second partition

Partition specification in Ψ:

```
comp Gear =
|[  extvar x1, x2, x3
    objfunc F1(x1,x2,x3)
    confunc g5(x1,x2,x3)
          , g6(x1,x2,x3)
          , g9(x2,x3)
```

```
          , g10(x1,x2)
          , g11(x1,x2)
]|

comp ShaftA2 =
|[  extvar x2, x3, x6
    intvar x4
    objfunc F4(x6)
          , F6(x4,x6)
    confunc g1(x2,x3,x4,x6)
          , g3(x4,x6)
          , g7(x2,x3,x4,x6)
]|

comp ShaftB2 =
|[  extvar x2, x3, x7
    intvar x5
    objfunc F5(x7)
          , F7(x5,x7)
    confunc g2(x2,x3,x5,x7)
          , g4(x5,x7)
          , g8(x2,x3,x5,x7)
]|

syst SpeedReducer2 =
|[  sub G: Gear, S1: ShaftA2, S2: ShaftB2
    objfunc F2(G.x1,S1.x6)
          , F3(G.x1,S2.x7)
    link G.x2 -- {S1.x2, S2.x2}
       , G.x3 -- {S1.x3, S2.x3}
]|

topsyst SpeedReducer2
```

Generated normalized partition:

```
[topsyst]
system = syst_1

[func_9]
path = SpeedReducer2.S1
argvars = var_5, var_6, var_4, var_7
name = g1
defined_in = ShaftA2

[func_8]
path = SpeedReducer2.S1
argvars = var_4, var_7
name = F6
defined_in = ShaftA2

[func_5]
path = SpeedReducer2.G
argvars = var_1, var_2
name = g10
defined_in = Gear

[func_4]
path = SpeedReducer2.G
argvars = var_2, var_3
name = g9
defined_in = Gear

[func_7]
path = SpeedReducer2.S1
argvars = var_7
name = F4
defined_in = ShaftA2

[func_6]
path = SpeedReducer2.G
argvars = var_1, var_2
name = g11
defined_in = Gear

[func_1]
path = SpeedReducer2.G
argvars = var_1, var_2, var_3
name = F1
defined_in = Gear

[func_3]
path = SpeedReducer2.G
argvars = var_1, var_2, var_3
name = g6
defined_in = Gear

[func_2]
path = SpeedReducer2.G
argvars = var_1, var_2, var_3
name = g5
defined_in = Gear

[func_18]
path = SpeedReducer2
argvars = var_1, var_11
name = F3
defined_in = SpeedReducer2
```

```
[func_11]
path = SpeedReducer2.S1
argvars = var_5, var_6, var_4, var_7
name = g7
defined_in = ShaftA2

[func_10]
path = SpeedReducer2.S1
argvars = var_4, var_7
name = g3
defined_in = ShaftA2

[func_13]
path = SpeedReducer2.S2
argvars = var_8, var_11
name = F7
defined_in = ShaftB2

[func_12]
path = SpeedReducer2.S2
argvars = var_11
name = F5
defined_in = ShaftB2

[func_15]
path = SpeedReducer2.S2
argvars = var_8, var_11
name = g4
defined_in = ShaftB2

[func_14]
path = SpeedReducer2.S2
argvars = var_9, var_10, var_8, var_11
name = g2
defined_in = ShaftB2

[func_17]
path = SpeedReducer2
argvars = var_1, var_7
name = F2
defined_in = SpeedReducer2

[func_16]
path = SpeedReducer2.S2
argvars = var_9, var_10, var_8, var_11
name = g8
defined_in = ShaftB2

[var_9]
path = SpeedReducer2.S2
name = x2
defined_in = ShaftB2

[var_8]
path = SpeedReducer2.S2
name = x5
defined_in = ShaftB2

[var_7]
path = SpeedReducer2.S1
name = x6
defined_in = ShaftA2

[var_6]
path = SpeedReducer2.S1
name = x3
defined_in = ShaftA2

[var_5]
path = SpeedReducer2.S1
name = x2
defined_in = ShaftA2

[var_4]
path = SpeedReducer2.S1
name = x4
defined_in = ShaftA2

[var_3]
path = SpeedReducer2.G
name = x3
defined_in = Gear

[var_2]
path = SpeedReducer2.G
name = x2
defined_in = Gear

[var_1]
path = SpeedReducer2.G
name = x1
defined_in = Gear

[var_11]
path = SpeedReducer2.S2
name = x7
defined_in = ShaftB2

[var_10]
path = SpeedReducer2.S2
name = x3
defined_in = ShaftB2

[comp_2]
name = S1
coupling_vars = var_6, var_5
objfuncs = func_7, func_8
local_vars = var_7, var_4
confuncs = func_9, func_10, func_11
path = SpeedReducer2.S1
type = ShaftA2

[comp_3]
name = S2
coupling_vars = var_9, var_10
objfuncs = func_12, func_13
local_vars = var_11, var_8
confuncs = func_14, func_15, func_16
path = SpeedReducer2.S2
type = ShaftB2

[syst_1]
subs = comp_1, comp_2, comp_3
links = link_1, link_2, link_3, link_4
objfuncs = func_17, func_18
path = SpeedReducer2
type = SpeedReducer2
name = SpeedReducer2

[comp_1]
name = G
coupling_vars = var_3, var_2
objfuncs = func_1
local_vars = var_1
confuncs = func_2, func_3, func_4, func_5, func_6
path = SpeedReducer2.G
type = Gear

[link_3]
defined_in = syst_1
coupling = var_3, var_10
path = SpeedReducer2

[link_2]
defined_in = syst_1
coupling = var_2, var_9
path = SpeedReducer2

[link_1]
defined_in = syst_1
coupling = var_2, var_5
path = SpeedReducer2

[link_4]
defined_in = syst_1
coupling = var_3, var_6
path = SpeedReducer2
```

Functional dependence table:

```
Columns:
 1: SpeedReducer2.G.x1
 2: SpeedReducer2.S1.x6
 3: SpeedReducer2.S2.x7
 4: SpeedReducer2.G.x2
 5: SpeedReducer2.S1.x2
 6: SpeedReducer2.S2.x2
 7: SpeedReducer2.G.x3
 8: SpeedReducer2.S2.x3
 9: SpeedReducer2.S1.x3
10: SpeedReducer2.S1.x4
```

```
11: SpeedReducer2.S2.x5

Rows:
 1: SpeedReducer2.F2
 2: SpeedReducer2.F3
 3: SpeedReducer2.G.x2 -- SpeedReducer2.S1.x2
 4: SpeedReducer2.G.x2 -- SpeedReducer2.S2.x2
 5: SpeedReducer2.G.x3 -- SpeedReducer2.S2.x3
 6: SpeedReducer2.G.x3 -- SpeedReducer2.S1.x3
 7: SpeedReducer2.G.F1
 8: SpeedReducer2.G.g5
 9: SpeedReducer2.G.g6
10: SpeedReducer2.G.g9
11: SpeedReducer2.G.g10
12: SpeedReducer2.G.g11
13: SpeedReducer2.S1.F4
14: SpeedReducer2.S1.F6
15: SpeedReducer2.S1.g1
16: SpeedReducer2.S1.g3
17: SpeedReducer2.S1.g7
18: SpeedReducer2.S2.F5
19: SpeedReducer2.S2.F7
20: SpeedReducer2.S2.g2
21: SpeedReducer2.S2.g4
22: SpeedReducer2.S2.g8

FDT:
1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0
1, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0
0, 0, 0, 1, 1, 0, 0, 0, 0, 0, 0
0, 0, 0, 1, 0, 1, 0, 0, 0, 0, 0
0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 0
0, 0, 0, 0, 0, 0, 1, 0, 1, 0, 0
1, 0, 0, 1, 0, 0, 1, 0, 0, 0, 0
1, 0, 0, 1, 0, 0, 1, 0, 0, 0, 0
1, 0, 0, 1, 0, 0, 1, 0, 0, 0, 0
0, 0, 0, 1, 0, 0, 1, 0, 0, 0, 0
1, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0
1, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0
0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0
0, 1, 0, 0, 0, 0, 0, 0, 0, 1, 0
0, 1, 0, 0, 1, 0, 0, 0, 1, 1, 0
0, 1, 0, 0, 0, 0, 0, 0, 0, 1, 0
0, 1, 0, 0, 1, 0, 0, 0, 1, 1, 0
0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0
0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 1
0, 0, 1, 0, 0, 1, 0, 1, 0, 0, 1
0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 1
0, 0, 1, 0, 0, 1, 0, 1, 0, 0, 1
```

4 Portal frame

First partition

Partition specification in Ψ:

```
comp Frame =
|[  extvar A1, A2, A3, I1, I2, I3, F1, F2, F3
    intvar u
    objfunc m(A1, A2, A3)
    confunc gd(u)
    resfunc (F1, F2, F3, u) = af(A1, A2, A3, I1, I2, I3)
]|

comp Beam =
|[  extvar A, I, F
    intvar z, s
    confunc gb(z, s)
          , gc(z)
```

```
    resfunc s = ab(F, z)
          , (A,I) = ac(z)
]|

syst Portal =
|[  sub F: Frame, B1,B2,B3: Beam
    link F.A1 -- B1.A, F.I1 -- B1.I, F.F1 -- B1.F
       , F.A2 -- B2.A, F.I2 -- B2.I, F.F2 -- B2.F
       , F.A3 -- B3.A, F.I3 -- B3.I, F.F3 -- B3.F
]|

topsyst Portal
```

Generated normalized partition:

```
[topsyst]
system = syst_1

[func_9]
path = Portal.B2
argvars = var_16
name = gc
defined_in = Beam

[func_8]
path = Portal.B2
argvars = var_16, var_17
name = gb
defined_in = Beam

[func_5]
path = Portal.B1
argvars = var_11
name = gc
defined_in = Beam

[func_4]
path = Portal.B1
argvars = var_11, var_12
name = gb
defined_in = Beam

[func_7]
path = Portal.B1
resvars = var_13, var_14
argvars = var_11
name = ac
defined_in = Beam

[func_6]
path = Portal.B1
resvars = var_12
argvars = var_15, var_11
name = ab
defined_in = Beam

[func_1]
path = Portal.F
argvars = var_2, var_3, var_4
name = m
defined_in = Frame

[func_3]
path = Portal.F
resvars = var_8, var_9, var_10, var_1
argvars = var_2, var_3, var_4, var_5, var_6, var_7
name = af
defined_in = Frame

[func_2]
path = Portal.F
argvars = var_1
name = gd
defined_in = Frame

[var_24]
path = Portal.B3
name = I
defined_in = Beam

[var_25]
path = Portal.B3
name = F
defined_in = Beam

[var_22]
path = Portal.B3
name = s
defined_in = Beam

[var_23]
path = Portal.B3
name = A
defined_in = Beam

[var_20]
path = Portal.B2
name = F
defined_in = Beam

[var_21]
path = Portal.B3
name = z
defined_in = Beam

[func_11]
path = Portal.B2
resvars = var_18, var_19
argvars = var_16
name = ac
defined_in = Beam

[func_10]
path = Portal.B2
resvars = var_17
argvars = var_20, var_16
name = ab
defined_in = Beam

[func_13]
path = Portal.B3
argvars = var_21
name = gc
defined_in = Beam

[func_12]
path = Portal.B3
argvars = var_21, var_22
name = gb
defined_in = Beam

[func_15]
path = Portal.B3
resvars = var_23, var_24
argvars = var_21
name = ac
defined_in = Beam

[func_14]
path = Portal.B3
resvars = var_22
argvars = var_25, var_21
name = ab
defined_in = Beam

[var_9]
path = Portal.F
name = F2
defined_in = Frame

[var_8]
path = Portal.F
name = F1
defined_in = Frame
```

```
[var_7]
path = Portal.F
name = I3
defined_in = Frame

[var_6]
path = Portal.F
name = I2
defined_in = Frame

[var_5]
path = Portal.F
name = I1
defined_in = Frame

[var_4]
path = Portal.F
name = A3
defined_in = Frame

[var_3]
path = Portal.F
name = A2
defined_in = Frame

[var_2]
path = Portal.F
name = A1
defined_in = Frame

[var_1]
path = Portal.F
name = u
defined_in = Frame

[var_17]
path = Portal.B2
name = s
defined_in = Beam

[var_16]
path = Portal.B2
name = z
defined_in = Beam

[var_15]
path = Portal.B1
name = F
defined_in = Beam

[var_14]
path = Portal.B1
name = I
defined_in = Beam

[var_13]
path = Portal.B1
name = A
defined_in = Beam

[var_12]
path = Portal.B1
name = s
defined_in = Beam

[var_11]
path = Portal.B1
name = z
defined_in = Beam

[var_10]
path = Portal.F
name = F3
defined_in = Frame

[var_19]
path = Portal.B2
name = I
defined_in = Beam

[var_18]
path = Portal.B2
name = A
defined_in = Beam

[comp_4]
resfuncs = func_14, func_15
name = B3
local_resvars = var_22
coupling_resvars = var_24, var_23
local_vars = var_21
coupling_vars = var_25
confuncs = func_12, func_13
path = Portal.B3
type = Beam

[comp_2]
resfuncs = func_6, func_7
name = B1
local_resvars = var_12
coupling_resvars = var_14, var_13
local_vars = var_11
coupling_vars = var_15
confuncs = func_4, func_5
path = Portal.B1
type = Beam

[comp_3]
resfuncs = func_10, func_11
name = B2
local_resvars = var_17
coupling_resvars = var_19, var_18
local_vars = var_16
coupling_vars = var_20
confuncs = func_8, func_9
path = Portal.B2
type = Beam

[syst_1]
path = Portal
type = Portal
name = Portal
links = link_1, link_2, link_3, link_4, link_5, link_6, link_7, link_8, link_9
subs = comp_1, comp_2, comp_3, comp_4

[link_9]
defined_in = syst_1
coupling = var_10, var_25
path = Portal

[link_8]
defined_in = syst_1
coupling = var_7, var_24
path = Portal

[comp_1]
resfuncs = func_3
name = F
local_resvars = var_1
objfuncs = func_1
coupling_resvars = var_8, var_9, var_10
coupling_vars = var_7, var_6, var_5, var_4, var_3, var_2
confuncs = func_2
path = Portal.F
type = Frame

[link_3]
defined_in = syst_1
coupling = var_8, var_15
path = Portal

[link_2]
defined_in = syst_1
coupling = var_5, var_14
path = Portal

[link_1]
defined_in = syst_1
coupling = var_2, var_13
path = Portal

[link_7]
defined_in = syst_1
coupling = var_4, var_23
path = Portal

[link_6]
defined_in = syst_1
coupling = var_9, var_20
path = Portal

[link_5]
defined_in = syst_1
coupling = var_6, var_19
path = Portal

[link_4]
defined_in = syst_1
coupling = var_3, var_18
path = Portal
```

Functional dependence table:

```
Columns:
 1: Portal.F.A1
 2: Portal.B1.A
 3: Portal.F.I1
 4: Portal.B1.I
 5: Portal.F.F1
 6: Portal.B1.F
 7: Portal.F.A2
 8: Portal.B2.A
 9: Portal.F.I2
10: Portal.B2.I
11: Portal.F.F2
12: Portal.B2.F
13: Portal.F.A3
14: Portal.B3.A
15: Portal.F.I3
16: Portal.B3.I
17: Portal.F.F3
18: Portal.B3.F
19: Portal.F.u
20: Portal.B1.z
21: Portal.B1.s
22: Portal.B2.z
23: Portal.B2.s
24: Portal.B3.z
25: Portal.B3.s

Rows:
 1: Portal.F.A1 -- Portal.B1.A
 2: Portal.F.I1 -- Portal.B1.I
 3: Portal.F.F1 -- Portal.B1.F
 4: Portal.F.A2 -- Portal.B2.A
 5: Portal.F.I2 -- Portal.B2.I
 6: Portal.F.F2 -- Portal.B2.F
 7: Portal.F.A3 -- Portal.B3.A
 8: Portal.F.I3 -- Portal.B3.I
 9: Portal.F.F3 -- Portal.B3.F
10: Portal.F.m
11: Portal.F.gd
12: Portal.F.af
13: Portal.B1.gb
14: Portal.B1.gc
15: Portal.B1.ab
16: Portal.B1.ac
17: Portal.B2.gb
18: Portal.B2.gc
19: Portal.B2.ab
20: Portal.B2.ac
21: Portal.B3.gb
22: Portal.B3.gc
23: Portal.B3.ab
24: Portal.B3.ac

FDT:
1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
0, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
0, 0, 0, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0
1, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0
1, 0, 1, 0, 1, 0, 1, 0, 1, 0, 1, 0, 1, 0, 1, 0, 1, 0, 1, 0, 0, 0, 0, 0, 0
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 0, 0
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0
0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 0, 0
0, 1, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0
0, 0, 0, 0, 0, 0, 0, 1, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 1, 1
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 1, 0, 0, 0, 0, 0, 0, 0, 1, 0
```

Second partition

Partition specification in Ψ:

```
comp Beam =
```

```
|[  extvar A, I, F
    intvar z, s
    confunc gb(z, s)
          , gc(z)
    resfunc s = ab(F, z)
          , (A,I) = ac(z)
]|

syst Portal2 =
|[  intvar F1, F2, F3, u
    sub B1,B2,B3: Beam
    objfunc m(B1.A, B2.A, B3.A)
    confunc gd(u)
    resfunc (F1, F2, F3, u) = af(B1.A, B2.A, B3.A, B1.I, B2.I, B3.I)
    link F1 -- B1.F, F2 -- B2.F, F3 -- B3.F
]|

topsyst Portal2
```

Generated normalized partition:

```
[topsyst]
system = syst_1

[func_9]
path = Portal2.B3
argvars = var_15, var_16
name = gb
defined_in = Beam

[func_8]
path = Portal2.B2
resvars = var_12, var_13
argvars = var_10
name = ac
defined_in = Beam

[func_5]
path = Portal2.B2
argvars = var_10, var_11
name = gb
defined_in = Beam

[func_4]
path = Portal2.B1
resvars = var_7, var_8
argvars = var_5
name = ac
defined_in = Beam

[func_7]
path = Portal2.B2
resvars = var_11
argvars = var_14, var_10
name = ab
defined_in = Beam

[func_6]
path = Portal2.B2
argvars = var_10
name = gc
defined_in = Beam

[func_1]
path = Portal2.B1
argvars = var_5, var_6
name = gb
defined_in = Beam

[func_3]
path = Portal2.B1
resvars = var_6
argvars = var_9, var_5
name = ab
defined_in = Beam

[func_2]
path = Portal2.B1
argvars = var_5
name = gc
defined_in = Beam

[func_11]
path = Portal2.B3
resvars = var_16
argvars = var_19, var_15
name = ab
defined_in = Beam

[func_10]
path = Portal2.B3
argvars = var_15
name = gc
defined_in = Beam

[func_13]
path = Portal2
argvars = var_7, var_12, var_17
name = m
defined_in = Portal2

[func_12]
path = Portal2.B3
resvars = var_17, var_18
argvars = var_15
name = ac
defined_in = Beam

[func_15]
path = Portal2
resvars = var_1, var_2, var_3, var_4
argvars = var_7, var_12, var_17, var_8, var_13, var_18
name = af
defined_in = Portal2

[func_14]
path = Portal2
argvars = var_4
name = gd
defined_in = Portal2

[var_9]
path = Portal2.B1
name = F
defined_in = Beam

[var_8]
path = Portal2.B1
name = I
defined_in = Beam

[var_7]
path = Portal2.B1
name = A
defined_in = Beam

[var_6]
path = Portal2.B1
name = s
```

```
defined_in = Beam

[var_5]
path = Portal2.B1
name = z
defined_in = Beam

[var_4]
path = Portal2
name = u
defined_in = Portal2

[var_3]
path = Portal2
name = F3
defined_in = Portal2

[var_2]
path = Portal2
name = F2
defined_in = Portal2

[var_1]
path = Portal2
name = F1
defined_in = Portal2

[var_17]
path = Portal2.B3
name = A
defined_in = Beam

[var_16]
path = Portal2.B3
name = s
defined_in = Beam

[var_15]
path = Portal2.B3
name = z
defined_in = Beam

[var_14]
path = Portal2.B2
name = F
defined_in = Beam

[var_13]
path = Portal2.B2
name = I
defined_in = Beam

[var_12]
path = Portal2.B2
name = A
defined_in = Beam

[var_11]
path = Portal2.B2
name = s
defined_in = Beam

[var_10]
path = Portal2.B2
name = z
defined_in = Beam

[var_19]
path = Portal2.B3
name = F
defined_in = Beam

[var_18]
path = Portal2.B3
name = I
defined_in = Beam

[comp_2]
resfuncs = func_7, func_8
name = B2
local_resvars = var_13, var_12, var_11
local_vars = var_10
coupling_vars = var_14
confuncs = func_5, func_6
path = Portal2.B2
type = Beam

[comp_3]
resfuncs = func_11, func_12
name = B3
local_resvars = var_17, var_16, var_18
local_vars = var_15
coupling_vars = var_19
confuncs = func_9, func_10
path = Portal2.B3
type = Beam

[syst_1]
resfuncs = func_15
name = Portal2
links = link_1, link_2, link_3
local_resvars = var_4
objfuncs = func_13
coupling_resvars = var_3, var_2, var_1
confuncs = func_14
path = Portal2
type = Portal2
subs = comp_1, comp_2, comp_3

[comp_1]
resfuncs = func_3, func_4
name = B1
local_resvars = var_7, var_6, var_8
local_vars = var_5
coupling_vars = var_9
confuncs = func_1, func_2
path = Portal2.B1
type = Beam

[link_3]
defined_in = syst_1
coupling = var_3, var_19
path = Portal2

[link_2]
defined_in = syst_1
coupling = var_2, var_14
path = Portal2

[link_1]
defined_in = syst_1
coupling = var_1, var_9
path = Portal2
```

Functional dependence table:

```
Columns:
 1: Portal2.B1.A
 2: Portal2.B2.A
 3: Portal2.B3.A
 4: Portal2.u
 5: Portal2.B1.I
 6: Portal2.B2.I
 7: Portal2.B3.I
 8: Portal2.F1
 9: Portal2.F2
10: Portal2.F3
11: Portal2.B1.F
12: Portal2.B2.F
13: Portal2.B3.F
14: Portal2.B1.z
15: Portal2.B1.s
16: Portal2.B2.z
17: Portal2.B2.s
18: Portal2.B3.z
19: Portal2.B3.s

Rows:
 1: Portal2.m
```

```
 2: Portal2.gd
 3: Portal2.af
 4: Portal2.F1 -- Portal2.B1.F
 5: Portal2.F2 -- Portal2.B2.F
 6: Portal2.F3 -- Portal2.B3.F
 7: Portal2.B1.gb
 8: Portal2.B1.gc
 9: Portal2.B1.ab
10: Portal2.B1.ac
11: Portal2.B2.gb
12: Portal2.B2.gc
13: Portal2.B2.ab
14: Portal2.B2.ac
15: Portal2.B3.gb
16: Portal2.B3.gc
17: Portal2.B3.ab
18: Portal2.B3.ac

FDT:
1, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
1, 1, 1, 1, 1, 1, 1, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0
0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0
0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 1, 0, 0, 0, 0, 0, 0
0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 1, 0, 0, 0, 0, 0
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 0
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 1, 1, 0, 0, 0
1, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 1, 1, 0, 0
0, 1, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 1, 1
0, 0, 1, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0
```

5 Chassis design

First partition

Partition specification in Ψ:

```
comp Vehicle =
|[  extvar a, b, Ksf, Ksr, Ktf, Ktr, Caf, Car
    intvar wsf, wsr, wtf, wtr, kus
    objfunc f(wsf, wsr, wtf, wtr, kus)
    resfunc (wsf, wsr, wtf, wtr, kus) = a1(a, b, Ksf, Ksr, Ktf, Ktr, Caf, Car)
]|

comp Tire =
|[  extvar a, b, Ktf, Ktr, Pif, Pir
    resfunc (Ktf, Ktr) = a3(Pif, Pir, a, b)
]|

comp Corner =
|[  extvar a, b, Caf, Car, Pif, Pir
    resfunc (Caf, Car) = a4(Pif, Pir, a, b)
]|

comp Suspension =
|[  extvar Ks, KL, KB, L0
```

```
    intvar Zs
    confunc g1(Zs, KL, KB, L0)
    resfunc Ks = a2(Zs, KL, KB, L0)
]|

comp Spring =
|[  extvar KL, KB, L0
    intvar D, d, p
    confunc g2(D, d, p)
    resfunc (KL, KB) = a5(D, d, p, L0)
]|

syst Chassis =
|[  sub V: Vehicle, T: Tire, C: Corner
          , Sf, Sr: Suspension, Spf, Spr: Spring
    link V.a   -- {T.a, C.a}, T.Pif -- C.Pif
       , V.b   -- {T.b, C.b}, T.Pir -- C.Pir
       , V.Ktf -- T.Ktf     , V.Caf -- C.Caf
       , V.Ktr -- T.Ktr     , V.Car -- C.Car
       , V.Ksf -- Sf.Ks     , V.Ksr -- Sr.Ks
       , Sf.L0 -- Spf.L0    , Sr.L0 -- Spr.L0
       , Sf.KL -- Spf.KL    , Sr.KL -- Spr.KL
       , Sf.KB -- Spf.KB    , Sr.KB -- Spr.KB

]|

topsyst Chassis
```

Generated normalized partition:

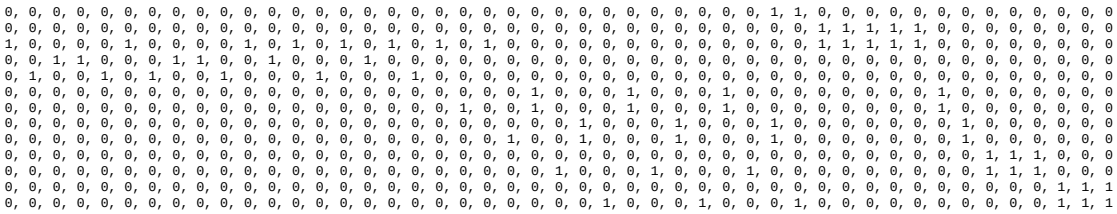
```
[var_44]
path = Chassis.Spr
name = p
defined_in = Spring

[var_45]
path = Chassis.Spr
name = KL
defined_in = Spring

[var_46]
path = Chassis.Spr
name = KB
defined_in = Spring

[var_47]
path = Chassis.Spr
name = L0
defined_in = Spring

[var_40]
path = Chassis.Spf
name = KB
defined_in = Spring

[var_41]
path = Chassis.Spf
name = L0
defined_in = Spring

[var_42]
path = Chassis.Spr
name = D
defined_in = Spring

[var_43]
path = Chassis.Spr
name = d
defined_in = Spring

[func_11]
path = Chassis.Spr
argvars = var_42, var_43, var_44
name = g2
defined_in = Spring

[func_10]
path = Chassis.Spf
resvars = var_39, var_40
argvars = var_36, var_37, var_38, var_41
name = a5
defined_in = Spring

[func_12]
path = Chassis.Spr
resvars = var_45, var_46
argvars = var_42, var_43, var_44, var_47
name = a5
defined_in = Spring

[var_9]
path = Chassis.V
name = Ksr
defined_in = Vehicle

[var_8]
path = Chassis.V
name = Ksf
defined_in = Vehicle

[var_7]
path = Chassis.V
name = b
defined_in = Vehicle

[var_6]
path = Chassis.V
name = a
defined_in = Vehicle

[var_5]
path = Chassis.V
name = kus
defined_in = Vehicle
```

```
[var_4]
path = Chassis.V
name = wtr
defined_in = Vehicle

[var_3]
path = Chassis.V
name = wtf
defined_in = Vehicle

[var_2]
path = Chassis.V
name = wsr
defined_in = Vehicle

[var_1]
path = Chassis.V
name = wsf
defined_in = Vehicle

[func_9]
path = Chassis.Spf
argvars = var_36, var_37, var_38
name = g2
defined_in = Spring

[func_8]
path = Chassis.Sr
resvars = var_32
argvars = var_31, var_33, var_34, var_35
name = a2
defined_in = Suspension

[func_5]
path = Chassis.Sf
argvars = var_26, var_28, var_29, var_30
name = g1
defined_in = Suspension

[func_4]
path = Chassis.C
resvars = var_22, var_23
argvars = var_24, var_25, var_20, var_21
name = a4
defined_in = Corner

[func_7]
path = Chassis.Sr
argvars = var_31, var_33, var_34, var_35
name = g1
defined_in = Suspension

[func_6]
path = Chassis.Sf
resvars = var_27
argvars = var_26, var_28, var_29, var_30
name = a2
defined_in = Suspension

[func_1]
path = Chassis.V
argvars = var_1, var_2, var_3, var_4, var_5
name = f
defined_in = Vehicle

[func_3]
path = Chassis.T
resvars = var_16, var_17
argvars = var_18, var_19, var_14, var_15
name = a3
defined_in = Tire

[func_2]
path = Chassis.V
resvars = var_1, var_2, var_3, var_4, var_5
argvars = var_6, var_7, var_8, var_9, var_10, var_11, var_12, var_13
name = a1
defined_in = Vehicle

[var_28]
path = Chassis.Sf
name = KL
defined_in = Suspension

[var_29]
path = Chassis.Sf
name = KB
defined_in = Suspension

[var_26]
path = Chassis.Sf
name = Zs
defined_in = Suspension

[var_27]
path = Chassis.Sf
name = Ks
defined_in = Suspension

[var_24]
path = Chassis.C
name = Pif
defined_in = Corner

[var_25]
path = Chassis.C
name = Pir
defined_in = Corner

[var_22]
path = Chassis.C
name = Caf
defined_in = Corner

[var_23]
path = Chassis.C
name = Car
defined_in = Corner

[var_20]
path = Chassis.C
name = a
defined_in = Corner

[var_21]
path = Chassis.C
name = b
defined_in = Corner

[var_39]
path = Chassis.Spf
name = KL
defined_in = Spring

[var_38]
path = Chassis.Spf
name = p
defined_in = Spring

[var_35]
path = Chassis.Sr
name = L0
defined_in = Suspension

[var_34]
path = Chassis.Sr
name = KB
defined_in = Suspension

[var_37]
path = Chassis.Spf
name = d
defined_in = Spring

[var_36]
path = Chassis.Spf
name = D
defined_in = Spring

[var_31]
path = Chassis.Sr
name = Zs
defined_in = Suspension

[var_30]
path = Chassis.Sf
name = L0
defined_in = Suspension

[var_33]
path = Chassis.Sr
name = KL
defined_in = Suspension

[var_32]
path = Chassis.Sr
name = Ks
defined_in = Suspension

[link_9]
defined_in = syst_1
coupling = var_11, var_17
path = Chassis

[link_8]
defined_in = syst_1
coupling = var_12, var_22
path = Chassis

[link_3]
defined_in = syst_1
coupling = var_18, var_24
path = Chassis

[link_2]
defined_in = syst_1
coupling = var_6, var_14
path = Chassis

[link_1]
defined_in = syst_1
coupling = var_6, var_20
path = Chassis

[link_7]
```

```
defined_in = syst_1
coupling = var_10, var_16
path = Chassis

[link_6]
defined_in = syst_1
coupling = var_19, var_25
path = Chassis

[link_5]
defined_in = syst_1
coupling = var_7, var_15
path = Chassis

[link_4]
defined_in = syst_1
coupling = var_7, var_21
path = Chassis

[topsyst]
system = syst_1

[var_17]
path = Chassis.T
name = Ktr
defined_in = Tire

[var_16]
path = Chassis.T
name = Ktf
defined_in = Tire

[var_15]
path = Chassis.T
name = b
defined_in = Tire

[var_14]
path = Chassis.T
name = a
defined_in = Tire

[var_13]
path = Chassis.V
name = Car
defined_in = Vehicle

[var_12]
path = Chassis.V
name = Caf
defined_in = Vehicle

[var_11]
path = Chassis.V
name = Ktr
defined_in = Vehicle

[var_10]
path = Chassis.V
name = Ktf
defined_in = Vehicle

[var_19]
path = Chassis.T
name = Pir
defined_in = Tire

[var_18]
path = Chassis.T
name = Pif
defined_in = Tire

[syst_1]
path = Chassis
type = Chassis
name = Chassis
links = link_1, link_2, link_3, link_4, link_5, link_6, link_7, link_8, link_9, link_10, link_11, link_12, link_13, link_14, link_15, link_16, link_17, link_18
subs = comp_1, comp_2, comp_3, comp_4, comp_5, comp_6, comp_7

[link_13]
defined_in = syst_1
coupling = var_30, var_41
path = Chassis

[link_12]
defined_in = syst_1
coupling = var_9, var_32
path = Chassis

[link_11]
defined_in = syst_1
coupling = var_8, var_27
path = Chassis

[link_10]
defined_in = syst_1
coupling = var_13, var_23
path = Chassis

[link_17]
defined_in = syst_1
coupling = var_29, var_40
path = Chassis

[link_16]
defined_in = syst_1
coupling = var_33, var_45
path = Chassis

[link_15]
defined_in = syst_1
coupling = var_28, var_39
path = Chassis

[link_14]
defined_in = syst_1
coupling = var_35, var_47
path = Chassis

[link_18]
defined_in = syst_1
coupling = var_34, var_46
path = Chassis

[comp_6]
resfuncs = func_10
name = Spf
coupling_resvars = var_39, var_40
local_vars = var_37, var_36, var_38
coupling_vars = var_41
confuncs = func_9
path = Chassis.Spf
type = Spring

[comp_7]
resfuncs = func_12
name = Spr
coupling_resvars = var_45, var_46
local_vars = var_44, var_42, var_43
coupling_vars = var_47
confuncs = func_11
path = Chassis.Spr
type = Spring

[comp_4]
resfuncs = func_6
name = Sf
coupling_resvars = var_27
local_vars = var_26
coupling_vars = var_30, var_28, var_29
confuncs = func_5
path = Chassis.Sf
type = Suspension

[comp_5]
resfuncs = func_8
name = Sr
coupling_resvars = var_32
local_vars = var_31
coupling_vars = var_35, var_34, var_33
confuncs = func_7
path = Chassis.Sr
type = Suspension

[comp_2]
resfuncs = func_3
name = T
coupling_resvars = var_17, var_16
coupling_vars = var_15, var_14, var_19, var_18
path = Chassis.T
type = Tire

[comp_3]
resfuncs = func_4
name = C
coupling_resvars = var_22, var_23
coupling_vars = var_24, var_25, var_20, var_21
path = Chassis.C
type = Corner

[comp_1]
resfuncs = func_2
name = V
local_resvars = var_5, var_4, var_3, var_2, var_1
objfuncs = func_1
coupling_vars = var_13, var_12, var_11, var_10, var_9, var_8, var_7, var_6
path = Chassis.V
type = Vehicle
```

Functional dependence table:

```
Columns:
 1: Chassis.V.a
 2: Chassis.C.a
 3: Chassis.T.a
 4: Chassis.T.Pif
 5: Chassis.C.Pif
 6: Chassis.V.b
 7: Chassis.C.b
 8: Chassis.T.b
 9: Chassis.T.Pir
10: Chassis.C.Pir
11: Chassis.V.Ktf
12: Chassis.T.Ktf
13: Chassis.V.Caf
14: Chassis.C.Caf
15: Chassis.V.Ktr
16: Chassis.T.Ktr
17: Chassis.V.Car
18: Chassis.C.Car
19: Chassis.V.Ksf
20: Chassis.Sf.Ks
21: Chassis.V.Ksr
22: Chassis.Sr.Ks
23: Chassis.Sf.L0
24: Chassis.Spf.L0
25: Chassis.Sr.L0
26: Chassis.Spr.L0
27: Chassis.Sf.KL
28: Chassis.Spf.KL
29: Chassis.Sr.KL
30: Chassis.Spr.KL
31: Chassis.Sf.KB
32: Chassis.Spf.KB
33: Chassis.Sr.KB
34: Chassis.Spr.KB
35: Chassis.V.wsf
36: Chassis.V.wsr
37: Chassis.V.wtf
38: Chassis.V.wtr
39: Chassis.V.kus
40: Chassis.Sf.Zs
41: Chassis.Sr.Zs
42: Chassis.Spf.D
43: Chassis.Spf.d
44: Chassis.Spf.p
45: Chassis.Spr.D
46: Chassis.Spr.d
47: Chassis.Spr.p

Rows:
 1: Chassis.V.a -- Chassis.C.a
 2: Chassis.V.a -- Chassis.T.a
 3: Chassis.T.Pif -- Chassis.C.Pif
 4: Chassis.V.b -- Chassis.C.b
 5: Chassis.V.b -- Chassis.T.b
 6: Chassis.T.Pir -- Chassis.C.Pir
 7: Chassis.V.Ktf -- Chassis.T.Ktf
 8: Chassis.V.Caf -- Chassis.C.Caf
 9: Chassis.V.Ktr -- Chassis.T.Ktr
10: Chassis.V.Car -- Chassis.C.Car
11: Chassis.V.Ksf -- Chassis.Sf.Ks
12: Chassis.V.Ksr -- Chassis.Sr.Ks
13: Chassis.Sf.L0 -- Chassis.Spf.L0
14: Chassis.Sr.L0 -- Chassis.Spr.L0
15: Chassis.Sf.KL -- Chassis.Spf.KL
16: Chassis.Sr.KL -- Chassis.Spr.KL
17: Chassis.Sf.KB -- Chassis.Spf.KB
18: Chassis.Sr.KB -- Chassis.Spr.KB
19: Chassis.V.f
20: Chassis.V.a1
21: Chassis.T.a3
22: Chassis.C.a4
23: Chassis.Sf.g1
24: Chassis.Sf.a2
25: Chassis.Sr.g1
26: Chassis.Sr.a2
27: Chassis.Spf.g2
28: Chassis.Spf.a5
29: Chassis.Spr.g2
30: Chassis.Spr.a5

FDT:
1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
1, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
0, 0, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
0, 0, 0, 0, 0, 0, 1, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 0, 0, 0, 0
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 0, 0
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
```



Second partition

Partition specification in Ψ:

```
comp Vehicle =
|[  extvar a, b, Ksf, Ksr, Ktf, Ktr, Caf, Car
    intvar wsf, wsr, wtf, wtr, kus
    objfunc f(wsf, wsr, wtf, wtr, kus)
    resfunc (wsf, wsr, wtf, wtr, kus) = a1(a, b, Ksf, Ksr, Ktf, Ktr, Caf, Car)
]|

comp Tire =
|[  extvar a, b, Ktf, Ktr, Pif, Pir
    resfunc (Ktf, Ktr) = a3(Pif, Pir, a, b)
]|

comp Corner =
|[  extvar a, b, Caf, Car, Pif, Pir
    resfunc (Caf, Car) = a4(Pif, Pir, a, b)
]|

comp Suspension =
|[  extvar Ks, KL, KB, L0
    intvar Zs
    confunc g1(Zs, KL, KB, L0)
    resfunc Ks = a2(Zs, KL, KB, L0)
]|

comp Spring =
|[  extvar KL, KB, L0
    intvar D, d, p
    confunc g2(D, d, p)
    resfunc (KL, KB) = a5(D, d, p, L0)
]|

syst SuspSpring =
|[  sub S: Suspension, Sp: Spring
    link S.L0 -- Sp.L0, S.KL -- Sp.KL, S.KB -- Sp.KB
    alias Ks = S.Ks
]|
```

```
syst Chassis2 =
|[  sub V: Vehicle, T: Tire, C: Corner
          , Sf, Sr: SuspSpring
    link V.a   -- {T.a, C.a}, T.Pif -- C.Pif
       , V.b   -- {T.b, C.b}, T.Pir -- C.Pir
       , V.Ktf -- T.Ktf     , V.Caf -- C.Caf
       , V.Ktr -- T.Ktr     , V.Car -- C.Car
       , V.Ksf -- Sf.Ks     , V.Ksr -- Sr.Ks

]|

topsyst Chassis2
```

Generated normalized partition:

```
[var_44]
path = Chassis2.Sr.Sp
name = p
defined_in = Spring

[var_45]
path = Chassis2.Sr.Sp
name = KL
defined_in = Spring

[var_46]
path = Chassis2.Sr.Sp
name = KB
defined_in = Spring

[var_47]
path = Chassis2.Sr.Sp
name = L0
defined_in = Spring

[var_40]
path = Chassis2.Sr.S
name = KB
defined_in = Suspension

[var_41]
path = Chassis2.Sr.S
name = L0
defined_in = Suspension

[var_42]
path = Chassis2.Sr.Sp
name = D
defined_in = Spring

[var_43]
path = Chassis2.Sr.Sp
name = d
defined_in = Spring

[func_11]
path = Chassis2.Sr.Sp
argvars = var_42, var_43, var_44
name = g2
defined_in = Spring

[func_10]
path = Chassis2.Sr.S
resvars = var_38
argvars = var_37, var_39, var_40, var_41
name = a2
defined_in = Suspension

[func_12]
path = Chassis2.Sr.Sp
resvars = var_45, var_46
argvars = var_42, var_43, var_44, var_47
name = a5
defined_in = Spring

[var_9]
path = Chassis2.V
name = Ksr
defined_in = Vehicle

[var_8]
path = Chassis2.V
name = Ksf
defined_in = Vehicle

[var_7]
path = Chassis2.V
name = b
defined_in = Vehicle

[var_6]
path = Chassis2.V
name = a
defined_in = Vehicle

[var_5]
path = Chassis2.V
name = kus
defined_in = Vehicle

[var_4]
path = Chassis2.V
name = wtr
defined_in = Vehicle

[var_3]
path = Chassis2.V
name = wtf
defined_in = Vehicle

[var_2]
path = Chassis2.V
name = wsr
defined_in = Vehicle

[var_1]
path = Chassis2.V
name = wsf
defined_in = Vehicle

[func_9]
path = Chassis2.Sr.S
argvars = var_37, var_39, var_40, var_41
name = g1
defined_in = Suspension

[func_8]
path = Chassis2.Sf.Sp
resvars = var_34, var_35
argvars = var_31, var_32, var_33, var_36
name = a5
defined_in = Spring

[func_5]
path = Chassis2.Sf.S
argvars = var_26, var_28, var_29, var_30
name = g1
defined_in = Suspension

[func_4]
path = Chassis2.C
resvars = var_22, var_23
argvars = var_24, var_25, var_20, var_21
name = a4
defined_in = Corner

[func_7]
path = Chassis2.Sf.Sp
argvars = var_31, var_32, var_33
name = g2
defined_in = Spring

[func_6]
path = Chassis2.Sf.S
resvars = var_27
```

```
argvars = var_26, var_28, var_29, var_30
name = a2
defined_in = Suspension

[func_1]
path = Chassis2.V
argvars = var_1, var_2, var_3, var_4, var_5
name = f
defined_in = Vehicle

[func_3]
path = Chassis2.T
resvars = var_16, var_17
argvars = var_18, var_19, var_14, var_15
name = a3
defined_in = Tire

[func_2]
path = Chassis2.V
resvars = var_1, var_2, var_3, var_4, var_5
argvars = var_6, var_7, var_8, var_9, var_10, var_11, var_12, var_13
name = a1
defined_in = Vehicle

[var_28]
path = Chassis2.Sf.S
name = KL
defined_in = Suspension

[var_29]
path = Chassis2.Sf.S
name = KB
defined_in = Suspension

[var_26]
path = Chassis2.Sf.S
name = Zs
defined_in = Suspension

[var_27]
path = Chassis2.Sf.S
name = Ks
defined_in = Suspension

[var_24]
path = Chassis2.C
name = Pif
defined_in = Corner

[var_25]
path = Chassis2.C
name = Pir
defined_in = Corner

[var_22]
path = Chassis2.C
name = Caf
defined_in = Corner

[var_23]
path = Chassis2.C
name = Car
defined_in = Corner

[var_20]
path = Chassis2.C
name = a
defined_in = Corner

[var_21]
path = Chassis2.C
name = b
defined_in = Corner

[var_39]
path = Chassis2.Sr.S
name = KL
defined_in = Suspension

[var_38]
path = Chassis2.Sr.S
name = Ks
defined_in = Suspension

[var_35]
path = Chassis2.Sf.Sp
name = KB
defined_in = Spring

[var_34]
path = Chassis2.Sf.Sp
name = KL
defined_in = Spring

[var_37]
path = Chassis2.Sr.S
name = Zs
defined_in = Suspension

[var_36]
path = Chassis2.Sf.Sp
name = L0
defined_in = Spring

[var_31]
path = Chassis2.Sf.Sp
name = D
defined_in = Spring

[var_30]
path = Chassis2.Sf.S
name = L0
defined_in = Suspension

[var_33]
path = Chassis2.Sf.Sp
name = p
defined_in = Spring

[var_32]
path = Chassis2.Sf.Sp
name = d
defined_in = Spring

[link_9]
defined_in = syst_1
coupling = var_18, var_24
path = Chassis2

[link_8]
defined_in = syst_1
coupling = var_6, var_14
path = Chassis2

[link_3]
defined_in = syst_2
coupling = var_29, var_35
path = Chassis2.Sf

[link_2]
defined_in = syst_2
coupling = var_28, var_34
path = Chassis2.Sf

[link_1]
defined_in = syst_2
coupling = var_30, var_36
path = Chassis2.Sf

[link_7]
defined_in = syst_1
coupling = var_6, var_20
path = Chassis2

[link_6]
defined_in = syst_3
coupling = var_40, var_46
path = Chassis2.Sr

[link_5]
defined_in = syst_3
coupling = var_39, var_45
path = Chassis2.Sr

[link_4]
defined_in = syst_3
coupling = var_41, var_47
path = Chassis2.Sr

[topsyst]
system = syst_1

[var_17]
path = Chassis2.T
name = Ktr
defined_in = Tire

[var_16]
path = Chassis2.T
name = Ktf
defined_in = Tire

[var_15]
path = Chassis2.T
name = b
defined_in = Tire

[var_14]
path = Chassis2.T
name = a
defined_in = Tire

[var_13]
path = Chassis2.V
name = Car
defined_in = Vehicle

[var_12]
path = Chassis2.V
name = Caf
defined_in = Vehicle

[var_11]
path = Chassis2.V
name = Ktr
defined_in = Vehicle
```

```
[var_10]
path = Chassis2.V
name = Ktf
defined_in = Vehicle

[var_19]
path = Chassis2.T
name = Pir
defined_in = Tire

[var_18]
path = Chassis2.T
name = Pif
defined_in = Tire

[syst_2]
path = Chassis2.Sf
type = SuspSpring
name = Sf
links = link_1, link_2, link_3
subs = comp_4, comp_5

[syst_3]
path = Chassis2.Sr
type = SuspSpring
name = Sr
links = link_4, link_5, link_6
subs = comp_6, comp_7

[syst_1]
path = Chassis2
type = Chassis2
name = Chassis2
links = link_7, link_8, link_9, link_10, link_11, link_12, link_13, link_14, link_15, link_16, link_17, link_18
subs = comp_1, comp_2, comp_3, syst_2, syst_3

[link_13]
defined_in = syst_1
coupling = var_10, var_16
path = Chassis2

[link_12]
defined_in = syst_1
coupling = var_19, var_25
path = Chassis2

[link_11]
defined_in = syst_1
coupling = var_7, var_15
path = Chassis2

[link_10]
defined_in = syst_1
coupling = var_7, var_21
path = Chassis2

[link_17]
defined_in = syst_1
coupling = var_8, var_27
path = Chassis2

[link_16]
defined_in = syst_1
coupling = var_13, var_23
path = Chassis2

[link_15]
defined_in = syst_1
coupling = var_11, var_17
path = Chassis2

[link_14]
defined_in = syst_1
coupling = var_12, var_22
path = Chassis2

[link_18]
defined_in = syst_1
coupling = var_9, var_38
path = Chassis2

[comp_6]
resfuncs = func_10
name = S
coupling_resvars = var_38
local_vars = var_37
coupling_vars = var_39, var_40, var_41
confuncs = func_9
path = Chassis2.Sr.S
type = Suspension

[comp_7]
resfuncs = func_12
name = Sp
coupling_resvars = var_45, var_46
local_vars = var_44, var_42, var_43
coupling_vars = var_47
confuncs = func_11
path = Chassis2.Sr.Sp
type = Spring

[comp_4]
resfuncs = func_6
name = S
coupling_resvars = var_27
local_vars = var_26
coupling_vars = var_30, var_28, var_29
confuncs = func_5
path = Chassis2.Sf.S
type = Suspension

[comp_5]
resfuncs = func_8
name = Sp
coupling_resvars = var_35, var_34
local_vars = var_31, var_33, var_32
coupling_vars = var_36
confuncs = func_7
path = Chassis2.Sf.Sp
type = Spring

[comp_2]
resfuncs = func_3
name = T
coupling_resvars = var_17, var_16
coupling_vars = var_15, var_14, var_19, var_18
path = Chassis2.T
type = Tire

[comp_3]
resfuncs = func_4
name = C
coupling_resvars = var_22, var_23
coupling_vars = var_24, var_25, var_20, var_21
path = Chassis2.C
type = Corner

[comp_1]
resfuncs = func_2
name = V
local_resvars = var_5, var_4, var_3, var_2, var_1
objfuncs = func_1
coupling_vars = var_13, var_12, var_11, var_10, var_9, var_8, var_7, var_6
path = Chassis2.V
type = Vehicle
```

Functional dependence table:

```
Columns:
 1: Chassis2.V.a
 2: Chassis2.C.a
 3: Chassis2.T.a
 4: Chassis2.T.Pif
 5: Chassis2.C.Pif
 6: Chassis2.V.b
 7: Chassis2.C.b
 8: Chassis2.T.b
 9: Chassis2.T.Pir
10: Chassis2.C.Pir
11: Chassis2.V.Ktf
12: Chassis2.T.Ktf
13: Chassis2.V.Caf
14: Chassis2.C.Caf
15: Chassis2.V.Ktr
16: Chassis2.T.Ktr
```

```
17: Chassis2.V.Car
18: Chassis2.C.Car
19: Chassis2.V.Ksf
20: Chassis2.Sf.S.Ks
21: Chassis2.V.Ksr
22: Chassis2.Sr.S.Ks
23: Chassis2.V.wsf
24: Chassis2.V.wsr
25: Chassis2.V.wtf
26: Chassis2.V.wtr
27: Chassis2.V.kus
28: Chassis2.Sf.S.L0
29: Chassis2.Sf.Sp.L0
30: Chassis2.Sf.S.KL
31: Chassis2.Sf.Sp.KL
32: Chassis2.Sf.S.KB
33: Chassis2.Sf.Sp.KB
34: Chassis2.Sf.S.Zs
35: Chassis2.Sf.Sp.D
36: Chassis2.Sf.Sp.d
37: Chassis2.Sf.Sp.p
38: Chassis2.Sr.S.L0
39: Chassis2.Sr.Sp.L0
40: Chassis2.Sr.S.KL
41: Chassis2.Sr.Sp.KL
42: Chassis2.Sr.S.KB
43: Chassis2.Sr.Sp.KB
44: Chassis2.Sr.S.Zs
45: Chassis2.Sr.Sp.D
46: Chassis2.Sr.Sp.d
47: Chassis2.Sr.Sp.p

Rows:
 1: Chassis2.V.a -- Chassis2.C.a
 2: Chassis2.V.a -- Chassis2.T.a
 3: Chassis2.T.Pif -- Chassis2.C.Pif
 4: Chassis2.V.b -- Chassis2.C.b
 5: Chassis2.V.b -- Chassis2.T.b
 6: Chassis2.T.Pir -- Chassis2.C.Pir
 7: Chassis2.V.Ktf -- Chassis2.T.Ktf
 8: Chassis2.V.Caf -- Chassis2.C.Caf
 9: Chassis2.V.Ktr -- Chassis2.T.Ktr
10: Chassis2.V.Car -- Chassis2.C.Car
11: Chassis2.V.Ksf -- Chassis2.Sf.S.Ks
12: Chassis2.V.Ksr -- Chassis2.Sr.S.Ks
13: Chassis2.V.f
14: Chassis2.V.a1
15: Chassis2.T.a3
16: Chassis2.C.a4
17: Chassis2.Sf.S.L0 -- Chassis2.Sf.Sp.L0
18: Chassis2.Sf.S.KL -- Chassis2.Sf.Sp.KL
19: Chassis2.Sf.S.KB -- Chassis2.Sf.Sp.KB
20: Chassis2.Sf.S.g1
21: Chassis2.Sf.S.a2
22: Chassis2.Sf.Sp.g2
23: Chassis2.Sf.Sp.a5
24: Chassis2.Sr.S.L0 -- Chassis2.Sr.Sp.L0
25: Chassis2.Sr.S.KL -- Chassis2.Sr.Sp.KL
26: Chassis2.Sr.S.KB -- Chassis2.Sr.Sp.KB
27: Chassis2.Sr.S.g1
28: Chassis2.Sr.S.a2
29: Chassis2.Sr.Sp.g2
30: Chassis2.Sr.Sp.a5

FDT:
1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
1, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
0, 0, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
0, 0, 0, 0, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
0, 0, 0, 0, 0, 1, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 1, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
1, 0, 0, 0, 0, 1, 0, 0, 0, 0, 1, 0, 1, 0, 1, 0, 1, 0, 1, 0, 1, 0, 1, 1, 1, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
0, 0, 1, 1, 0, 0, 0, 1, 1, 0, 0, 1, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
0, 1, 0, 0, 1, 0, 1, 0, 0, 1, 0, 0, 0, 1, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 1, 0, 1, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 1, 0, 1, 0, 1, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 1, 0, 1, 0, 1, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 0, 0, 0, 0
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 0, 0
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 1, 0, 1, 0, 1, 0, 0, 0
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 1, 0, 1, 0, 1, 0, 0, 0
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 1
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 1, 0, 1, 0, 1, 1, 1
```

6 Business jet

First partition

Partition specification in Ψ:

```
comp Range =
|[  extvar h, M, LD, SFC, Wt, Wf
    intvar R
    objfunc fr(R)
    resfunc R= ar(h, M, LD, SFC, Wt, Wf)
]|

comp Struc =
|[  extvar ysa, Wf, We, L, Wt, tw
    intvar xs
    confunc gs(ysa, xs, We, L)
    resfunc (Wt, Wf, tw) = as(ysa, xs, We, L)
]|

comp Aero =
|[  extvar ysa, h, M, LD, ESF, D, L, Wt, tw
    intvar xa
    confunc ga(ysa, xa, h, M, ESF, Wt, tw)
    resfunc (L, LD, D) = aa(ysa, xa, h, M, ESF, Wt, tw)
]|

comp Prop =
|[  extvar h, M, SFC, ESF, D, We
    intvar xp
    confunc gp(xp, h, M, D)
    resfunc (ESF, We, SFC) = ap(xp, h, M, D)
]|

syst Jet =
|[  sub R: Range, S: Struc, A: Aero, P: Prop
    link R.LD  -- A.LD  , R.h   -- {A.h, P.h}
       , R.Wf  -- S.Wf  , R.M   -- {A.M, P.M}
       , R.Wt  -- S.Wt  , R.SFC -- P.SFC
       , A.D   -- P.D   , A.ESF -- P.ESF
       , A.ysa -- S.ysa , A.L   -- S.L
       , A.Wt  -- S.Wt  , A.tw  -- S.tw
       , P.We  -- S.We

]|

topsyst Jet
```

Generated normalized partition:

```
[link_13]
defined_in = syst_1
coupling = var_23, var_13
path = Jet

[topsyst]
system = syst_1

[func_8]
path = Jet.P
resvars = var_29, var_31, var_28
argvars = var_25, var_26, var_27, var_30
name = ap
defined_in = Prop

[link_12]
defined_in = syst_1
coupling = var_22, var_12
path = Jet

[func_5]
path = Jet.A
argvars = var_16, var_15, var_17, var_18, var_20, var_23, var_24
name = ga
defined_in = Aero

[func_4]
path = Jet.S
resvars = var_13, var_10, var_14
argvars = var_9, var_8, var_11, var_12
name = as
defined_in = Struc

[func_7]
path = Jet.P
argvars = var_25, var_26, var_27, var_30
name = gp
defined_in = Prop

[func_6]
path = Jet.A
resvars = var_22, var_19, var_21
argvars = var_16, var_15, var_17, var_18, var_20, var_23, var_24
name = aa
defined_in = Aero

[func_1]
path = Jet.R
argvars = var_1
name = fr
defined_in = Range

[link_11]
defined_in = syst_1
coupling = var_16, var_9
path = Jet

[func_3]
path = Jet.S
argvars = var_9, var_8, var_11, var_12
name = gs
defined_in = Struc

[func_2]
path = Jet.R
resvars = var_1
argvars = var_2, var_3, var_4, var_5, var_6, var_7
name = ar
defined_in = Range

[link_10]
defined_in = syst_1
coupling = var_20, var_29
path = Jet

[var_28]
path = Jet.P
name = SFC
defined_in = Prop

[var_29]
path = Jet.P
name = ESF
defined_in = Prop

[var_26]
path = Jet.P
name = h
defined_in = Prop

[var_27]
path = Jet.P
name = M
defined_in = Prop

[var_24]
path = Jet.A
name = tw
defined_in = Aero

[var_25]
path = Jet.P
name = xp
defined_in = Prop

[var_22]
path = Jet.A
name = L
defined_in = Aero

[var_23]
path = Jet.A
name = Wt
defined_in = Aero

[var_20]
path = Jet.A
name = ESF
defined_in = Aero

[var_21]
path = Jet.A
name = D
defined_in = Aero

[link_15]
defined_in = syst_1
coupling = var_31, var_11
path = Jet

[comp_2]
resfuncs = func_4
name = S
coupling_resvars = var_14, var_13, var_10
local_vars = var_8
coupling_vars = var_9, var_12, var_11
confuncs = func_3
path = Jet.S
type = Struc

[link_14]
defined_in = syst_1
coupling = var_24, var_14
path = Jet

[comp_3]
resfuncs = func_6
name = A
coupling_resvars = var_22, var_19, var_21
local_vars = var_15
coupling_vars = var_17, var_16, var_18, var_24, var_23, var_20
confuncs = func_5
path = Jet.A
type = Aero

[var_9]
path = Jet.S
name = ysa
defined_in = Struc

[var_8]
path = Jet.S
name = xs
defined_in = Struc

[var_7]
path = Jet.R
name = Wf
defined_in = Range

[var_6]
path = Jet.R
name = Wt
defined_in = Range

[var_5]
path = Jet.R
name = SFC
defined_in = Range

[var_4]
path = Jet.R
name = LD
defined_in = Range

[var_3]
path = Jet.R
name = M
defined_in = Range

[var_2]
path = Jet.R
name = h
defined_in = Range

[var_1]
path = Jet.R
name = R
defined_in = Range

[var_17]
path = Jet.A
name = h
defined_in = Aero

[var_16]
```

```
path = Jet.A
name = ysa
defined_in = Aero

[var_15]
path = Jet.A
name = xa
defined_in = Aero

[var_14]
path = Jet.S
name = tw
defined_in = Struc

[var_13]
path = Jet.S
name = Wt
defined_in = Struc

[var_12]
path = Jet.S
name = L
defined_in = Struc

[var_11]
path = Jet.S
name = We
defined_in = Struc

[var_10]
path = Jet.S
name = Wf
defined_in = Struc

[var_19]
path = Jet.A
name = LD
defined_in = Aero

[var_18]
path = Jet.A
name = M
defined_in = Aero

[comp_4]
resfuncs = func_8
name = P
coupling_resvars = var_31, var_28, var_29
local_vars = var_25
coupling_vars = var_26, var_27, var_30
confuncs = func_7
path = Jet.P
type = Prop

[var_31]
path = Jet.P
name = We
defined_in = Prop

[var_30]
path = Jet.P
name = D
defined_in = Prop
```

```
[syst_1]
path = Jet
type = Jet
name = Jet
links = link_1, link_2, link_3, link_4, link_5, link_6, link_7, link_8, link_9, link_10, link_11, link_
subs = comp_1, comp_2, comp_3, comp_4

[link_9]
defined_in = syst_1
coupling = var_21, var_30
path = Jet

[link_8]
defined_in = syst_1
coupling = var_5, var_28
path = Jet

[comp_1]
resfuncs = func_2
name = R
local_resvars = var_1
objfuncs = func_1
coupling_vars = var_7, var_6, var_5, var_4, var_3, var_2
path = Jet.R
type = Range

[link_3]
defined_in = syst_1
coupling = var_2, var_26
path = Jet

[link_2]
defined_in = syst_1
coupling = var_2, var_17
path = Jet

[link_1]
defined_in = syst_1
coupling = var_4, var_19
path = Jet

[link_7]
defined_in = syst_1
coupling = var_6, var_13
path = Jet

[link_6]
defined_in = syst_1
coupling = var_3, var_18
path = Jet

[link_5]
defined_in = syst_1
coupling = var_3, var_27
path = Jet

[link_4]
defined_in = syst_1
coupling = var_7, var_10
path = Jet
```

Functional dependence table:

```
Columns:
 1: Jet.R.LD
 2: Jet.A.LD
 3: Jet.R.h
 4: Jet.A.h
 5: Jet.P.h
 6: Jet.R.Wf
 7: Jet.S.Wf
 8: Jet.R.M
 9: Jet.P.M
10: Jet.A.M
11: Jet.R.Wt
12: Jet.S.Wt
13: Jet.R.SFC
14: Jet.P.SFC
15: Jet.A.D
16: Jet.P.D
17: Jet.A.ESF
18: Jet.P.ESF
19: Jet.A.ysa
20: Jet.S.ysa
21: Jet.A.L
22: Jet.S.L
23: Jet.A.Wt
24: Jet.A.tw
25: Jet.S.tw
26: Jet.P.We
27: Jet.S.We
```

```
28: Jet.R.R
29: Jet.S.xs
30: Jet.A.xa
31: Jet.P.xp

Rows:
 1: Jet.R.LD -- Jet.A.LD
 2: Jet.R.h -- Jet.A.h
 3: Jet.R.h -- Jet.P.h
 4: Jet.R.Wf -- Jet.S.Wf
 5: Jet.R.M -- Jet.P.M
 6: Jet.R.M -- Jet.A.M
 7: Jet.R.Wt -- Jet.S.Wt
 8: Jet.R.SFC -- Jet.P.SFC
 9: Jet.A.D -- Jet.P.D
10: Jet.A.ESF -- Jet.P.ESF
11: Jet.A.ysa -- Jet.S.ysa
12: Jet.A.L -- Jet.S.L
13: Jet.A.Wt -- Jet.S.Wt
14: Jet.A.tw -- Jet.S.tw
15: Jet.P.We -- Jet.S.We
16: Jet.R.fr
17: Jet.R.ar
18: Jet.S.gs
19: Jet.S.as
20: Jet.A.ga
21: Jet.A.aa
22: Jet.P.gp
23: Jet.P.ap

FDT:
1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
0, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
0, 0, 1, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
0, 0, 0, 0, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
0, 0, 0, 0, 0, 0, 0, 1, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 0, 0, 0, 0
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0
1, 0, 1, 0, 0, 1, 0, 1, 0, 0, 1, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 1, 0, 0
0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 1, 0, 1, 0, 0, 1, 0, 1, 0, 1, 0, 0
0, 0, 0, 1, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 1, 0, 1, 0, 0, 0, 1, 1, 0, 0, 0, 0, 0, 1, 0
0, 1, 0, 1, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 1, 0, 1, 0, 1, 0, 1, 0, 1, 1, 0, 0, 0, 0, 0, 1, 0
0, 0, 0, 0, 1, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1
0, 0, 0, 0, 1, 0, 0, 0, 1, 0, 0, 0, 0, 1, 0, 1, 0, 1, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 1
```

Second partition

Partition specification in Ψ:

```
comp Range =
|[  extvar h, M, LD, SFC, Wt, Wf
    intvar R
    objfunc fr(R)
    resfunc R= ar(h, M, LD, SFC, Wt, Wf)
]|

comp StrucAero =
|[  extvar h, M, ESF, We, LD, D, Wt, Wf
    intvar ysa, xs, xa, L, tw
    confunc gs(ysa, xs, We, L)
          , ga(ysa, xa, h, M, ESF, Wt, tw)
    resfunc (Wt, Wf, tw) = as(ysa, xs, We, L)
          , (L, LD, D) = aa(ysa, xa, h, M, ESF, Wt, tw)
]|
```

```
comp Prop =
|[  extvar h, M, SFC, ESF, D, We
    intvar xp
    confunc gp(xp, h, M, D)
    resfunc (ESF, We, SFC) = ap(xp, h, M, D)
]|

syst Jet2 =
|[  sub R: Range, SA: StrucAero, P: Prop
    link R.LD -- SA.LD , R.h    -- {SA.h, P.h}
       , R.Wf -- SA.Wf , R.M    -- {SA.M, P.M}
       , R.Wt -- SA.Wt , R.SFC  -- P.SFC
       , SA.D -- P.D   , SA.ESF -- P.ESF
       , P.We -- SA.We

]|

topsyst Jet2
```

Generated normalized partition:

```
[topsyst]
system = syst_1

[func_8]
path = Jet2.P
resvars = var_25, var_27, var_24
argvars = var_21, var_22, var_23, var_26
name = ap
defined_in = Prop

[func_5]
path = Jet2.SA
resvars = var_19, var_20, var_12
argvars = var_8, var_9, var_16, var_11
name = as
defined_in = StrucAero

[func_4]
path = Jet2.SA
argvars = var_8, var_10, var_13, var_14, var_15, var_19, var_12
name = ga
defined_in = StrucAero

[func_7]
path = Jet2.P
argvars = var_21, var_22, var_23, var_26
name = gp
defined_in = Prop

[func_6]
path = Jet2.SA
resvars = var_11, var_17, var_18
argvars = var_8, var_10, var_13, var_14, var_15, var_19, var_12
name = aa
defined_in = StrucAero

[func_1]
path = Jet2.R
argvars = var_1
name = fr
defined_in = Range

[link_11]
defined_in = syst_1
coupling = var_27, var_16
path = Jet2

[func_3]
path = Jet2.SA
argvars = var_8, var_9, var_16, var_11
name = gs
defined_in = StrucAero

[func_2]
path = Jet2.R
resvars = var_1
argvars = var_2, var_3, var_4, var_5, var_6, var_7
name = ar
defined_in = Range

[link_10]
defined_in = syst_1
coupling = var_15, var_25
path = Jet2

[var_26]
path = Jet2.P
name = D
defined_in = Prop

[var_27]
path = Jet2.P
name = We
defined_in = Prop

[var_24]
path = Jet2.P
name = SFC
defined_in = Prop

[var_25]
path = Jet2.P
name = ESF
defined_in = Prop

[var_22]
path = Jet2.P
name = h
defined_in = Prop

[var_23]
path = Jet2.P
name = M
defined_in = Prop

[var_20]
path = Jet2.SA
name = Wf
defined_in = StrucAero

[var_21]
path = Jet2.P
name = xp
defined_in = Prop

[var_9]
path = Jet2.SA
name = xs
defined_in = StrucAero

[var_8]
path = Jet2.SA
name = ysa
defined_in = StrucAero
```

```
[var_7]
path = Jet2.R
name = Wf
defined_in = Range

[var_6]
path = Jet2.R
name = Wt
defined_in = Range

[var_5]
path = Jet2.R
name = SFC
defined_in = Range

[var_4]
path = Jet2.R
name = LD
defined_in = Range

[var_3]
path = Jet2.R
name = M
defined_in = Range

[var_2]
path = Jet2.R
name = h
defined_in = Range

[var_1]
path = Jet2.R
name = R
defined_in = Range

[var_17]
path = Jet2.SA
name = LD
defined_in = StrucAero

[var_16]
path = Jet2.SA
name = We
defined_in = StrucAero

[var_15]
path = Jet2.SA
name = ESF
defined_in = StrucAero

[var_14]
path = Jet2.SA
name = M
defined_in = StrucAero

[var_13]
path = Jet2.SA
name = h
defined_in = StrucAero

[var_12]
path = Jet2.SA
name = tw
defined_in = StrucAero

[var_11]
path = Jet2.SA
name = L
defined_in = StrucAero

[var_10]
path = Jet2.SA
name = xa
defined_in = StrucAero

[var_19]
path = Jet2.SA
name = Wt
defined_in = StrucAero

[var_18]
path = Jet2.SA
name = D
defined_in = StrucAero

[comp_2]
resfuncs = func_5, func_6
name = SA
local_resvars = var_12, var_11
coupling_resvars = var_17, var_20, var_19, var_18
local_vars = var_10, var_9, var_8
coupling_vars = var_16, var_15, var_14, var_13
confuncs = func_3, func_4
path = Jet2.SA
type = StrucAero

[comp_3]
resfuncs = func_8
name = P
coupling_resvars = var_27, var_24, var_25
local_vars = var_21
coupling_vars = var_26, var_22, var_23
confuncs = func_7
path = Jet2.P
type = Prop

[syst_1]
path = Jet2
type = Jet2
name = Jet2
links = link_1, link_2, link_3, link_4, link_5, link_6, link_7, link_8, link_9, link_10, link_
subs = comp_1, comp_2, comp_3

[link_9]
defined_in = syst_1
coupling = var_18, var_26
path = Jet2

[link_8]
defined_in = syst_1
coupling = var_5, var_24
path = Jet2

[comp_1]
resfuncs = func_2
name = R
local_resvars = var_1
objfuncs = func_1
coupling_vars = var_7, var_6, var_5, var_4, var_3, var_2
path = Jet2.R
type = Range

[link_3]
defined_in = syst_1
coupling = var_2, var_13
path = Jet2

[link_2]
defined_in = syst_1
coupling = var_2, var_22
path = Jet2

[link_1]
defined_in = syst_1
coupling = var_4, var_17
path = Jet2

[link_7]
defined_in = syst_1
coupling = var_6, var_19
path = Jet2

[link_6]
defined_in = syst_1
coupling = var_3, var_14
path = Jet2

[link_5]
defined_in = syst_1
coupling = var_3, var_23
path = Jet2

[link_4]
defined_in = syst_1
coupling = var_7, var_20
path = Jet2
```

Functional dependence table:

```
Columns:
 1: Jet2.R.LD
 2: Jet2.SA.LD
 3: Jet2.R.h
 4: Jet2.P.h
 5: Jet2.SA.h
```

```
 6: Jet2.R.Wf
 7: Jet2.SA.Wf
 8: Jet2.R.M
 9: Jet2.P.M
10: Jet2.SA.M
11: Jet2.R.Wt
12: Jet2.SA.Wt
13: Jet2.R.SFC
14: Jet2.P.SFC
15: Jet2.SA.D
16: Jet2.P.D
17: Jet2.SA.ESF
18: Jet2.P.ESF
19: Jet2.P.We
20: Jet2.SA.We
21: Jet2.R.R
22: Jet2.SA.ysa
23: Jet2.SA.xs
24: Jet2.SA.L
25: Jet2.SA.xa
26: Jet2.SA.tw
27: Jet2.P.xp

Rows:
 1: Jet2.R.LD -- Jet2.SA.LD
 2: Jet2.R.h -- Jet2.P.h
 3: Jet2.R.h -- Jet2.SA.h
 4: Jet2.R.Wf -- Jet2.SA.Wf
 5: Jet2.R.M -- Jet2.P.M
 6: Jet2.R.M -- Jet2.SA.M
 7: Jet2.R.Wt -- Jet2.SA.Wt
 8: Jet2.R.SFC -- Jet2.P.SFC
 9: Jet2.SA.D -- Jet2.P.D
10: Jet2.SA.ESF -- Jet2.P.ESF
11: Jet2.P.We -- Jet2.SA.We
12: Jet2.R.fr
13: Jet2.R.ar
14: Jet2.SA.gs
15: Jet2.SA.ga
16: Jet2.SA.as
17: Jet2.SA.aa
18: Jet2.P.gp
19: Jet2.P.ap

FDT:
1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
0, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
0, 0, 1, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
0, 0, 0, 0, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
0, 0, 0, 0, 0, 0, 0, 1, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0
1, 0, 1, 0, 0, 1, 0, 1, 0, 0, 1, 0, 1, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 1, 1, 1, 0, 0, 0
0, 0, 0, 0, 1, 0, 0, 0, 0, 1, 0, 1, 0, 0, 0, 0, 1, 0, 0, 0, 0, 1, 0, 0, 1, 1, 0
0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 1, 0, 1, 1, 1, 0, 1, 0
0, 1, 0, 0, 1, 0, 0, 0, 0, 1, 0, 1, 0, 0, 1, 0, 1, 0, 0, 0, 0, 1, 0, 1, 1, 1, 0
0, 0, 0, 1, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1
0, 0, 0, 1, 0, 0, 0, 0, 1, 0, 0, 0, 0, 1, 0, 1, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 1
```

Third partition

Partition specification in Ψ:

```
comp Range =
|[  extvar h, M, LD, SFC, Wt, Wf
    intvar R
    objfunc fr(R)
    resfunc R= ar(h, M, LD, SFC, Wt, Wf)
]|

comp Struc =
|[  extvar ysa, Wf, We, L, Wt, tw
```

```
    intvar xs
    confunc gs(ysa, xs, We, L)
    resfunc (Wt, Wf, tw) = as(ysa, xs, We, L)
]|

comp Aero =
|[  extvar ysa, h, M, LD, ESF, D, L, Wt, tw
    intvar xa
    confunc ga(ysa, xa, h, M, ESF, Wt, tw)
    resfunc (L, LD, D) = aa(ysa, xa, h, M, ESF, Wt, tw)
]|

comp Prop =
|[  extvar h, M, SFC, ESF, D, We
    intvar xp
    confunc gp(xp, h, M, D)
    resfunc (ESF, We, SFC) = ap(xp, h, M, D)
]|

syst StrucAero2 =
|[  sub S: Struc, A: Aero
    link A.ysa -- S.ysa , A.L   -- S.L
       , A.Wt  -- S.Wt  , A.tw  -- S.tw
    alias Wf = S.Wf , ESF = A.ESF
        , We = S.We , LD  = A.LD
        , Wt = S.Wt , D   = A.D
        , h = A.h   , M   = A.M
]|

syst Jet2 =
|[  sub R: Range, SA: StrucAero2, P: Prop
    link R.LD  -- SA.LD  , R.h    -- {SA.h, P.h}
       , R.Wf  -- SA.Wf  , R.M    -- {SA.M, P.M}
       , R.Wt  -- SA.Wt  , R.SFC  -- P.SFC
       , SA.D  -- P.D    , SA.ESF -- P.ESF
       , P.We  -- SA.We

]|

topsyst Jet2
```

Generated normalized partition:

```
[link_13]
defined_in = syst_1
coupling = var_21, var_30
path = Jet2

[topsyst]
system = syst_1

[func_8]
path = Jet2.P
resvars = var_29, var_31, var_28
argvars = var_25, var_26, var_27, var_30
name = ap
defined_in = Prop

[link_12]
defined_in = syst_1
coupling = var_5, var_28
path = Jet2

[func_5]
path = Jet2.SA.A
argvars = var_16, var_15, var_17, var_18, var_20, var_23, var_24
name = ga
defined_in = Aero

[func_4]
path = Jet2.SA.S
resvars = var_13, var_10, var_14
argvars = var_9, var_8, var_11, var_12
name = as
defined_in = Struc

[func_7]
path = Jet2.P
argvars = var_25, var_26, var_27, var_30
```

```
name = gp
defined_in = Prop

[func_6]
path = Jet2.SA.A
resvars = var_22, var_19, var_21
argvars = var_16, var_15, var_17, var_18, var_20, var_23, var_24
name = aa
defined_in = Aero

[func_1]
path = Jet2.R
argvars = var_1
name = fr
defined_in = Range

[link_11]
defined_in = syst_1
coupling = var_6, var_13
path = Jet2

[func_3]
path = Jet2.SA.S
argvars = var_9, var_8, var_11, var_12
name = gs
defined_in = Struc

[func_2]
path = Jet2.R
resvars = var_1
argvars = var_2, var_3, var_4, var_5, var_6, var_7
name = ar
defined_in = Range

[link_10]
defined_in = syst_1
coupling = var_3, var_18
path = Jet2

[var_28]
path = Jet2.P
name = SFC
defined_in = Prop

[var_29]
path = Jet2.P
name = ESF
defined_in = Prop

[var_26]
path = Jet2.P
name = h
defined_in = Prop

[var_27]
path = Jet2.P
name = M
defined_in = Prop

[var_24]
path = Jet2.SA.A
name = tw
defined_in = Aero

[var_25]
path = Jet2.P
name = xp
defined_in = Prop

[var_22]
path = Jet2.SA.A
name = L
defined_in = Aero

[var_23]
path = Jet2.SA.A
name = Wt
defined_in = Aero

[var_20]
path = Jet2.SA.A
name = ESF
defined_in = Aero

[var_21]
path = Jet2.SA.A
name = D
defined_in = Aero

[link_15]
defined_in = syst_1
coupling = var_31, var_11
path = Jet2

[comp_2]
resfuncs = func_4
name = S
coupling_resvars = var_14, var_13, var_10
local_vars = var_8
coupling_vars = var_9, var_12, var_11
confuncs = func_3
path = Jet2.SA.S
type = Struc

[link_14]
defined_in = syst_1
coupling = var_20, var_29
path = Jet2

[comp_3]
resfuncs = func_6
name = A
coupling_resvars = var_22, var_19, var_21
local_vars = var_15
coupling_vars = var_17, var_16, var_18, var_24, var_23, var_20
confuncs = func_5
path = Jet2.SA.A
type = Aero

[var_9]
path = Jet2.SA.S
name = ysa
defined_in = Struc

[var_8]
path = Jet2.SA.S
name = xs
defined_in = Struc

[var_7]
path = Jet2.R
name = Wf
defined_in = Range

[var_6]
path = Jet2.R
name = Wt
defined_in = Range

[var_5]
path = Jet2.R
name = SFC
defined_in = Range

[var_4]
path = Jet2.R
name = LD
defined_in = Range

[var_3]
path = Jet2.R
name = M
defined_in = Range

[var_2]
path = Jet2.R
name = h
defined_in = Range

[var_1]
path = Jet2.R
name = R
defined_in = Range

[var_17]
path = Jet2.SA.A
name = h
defined_in = Aero

[var_16]
path = Jet2.SA.A
name = ysa
defined_in = Aero

[var_15]
path = Jet2.SA.A
name = xa
defined_in = Aero

[var_14]
path = Jet2.SA.S
name = tw
defined_in = Struc

[var_13]
path = Jet2.SA.S
name = Wt
defined_in = Struc

[var_12]
path = Jet2.SA.S
name = L
defined_in = Struc

[var_11]
path = Jet2.SA.S
name = We
defined_in = Struc

[var_10]
path = Jet2.SA.S
name = Wf
defined_in = Struc

[syst_2]
path = Jet2.SA
```

```
type = StrucAero2
name = SA
links = link_1, link_2, link_3, link_4
subs = comp_2, comp_3

[var_19]
path = Jet2.SA.A
name = LD
defined_in = Aero

[var_18]
path = Jet2.SA.A
name = M
defined_in = Aero

[syst_1]
path = Jet2
type = Jet2
name = Jet2
links = link_5, link_6, link_7, link_8, link_9, link_10, link_11, link_12, link_13, link_14, link_15
subs = comp_1, comp_4, syst_2

[comp_4]
resfuncs = func_8
name = P
coupling_resvars = var_31, var_28, var_29
local_vars = var_25
coupling_vars = var_26, var_27, var_30
confuncs = func_7
path = Jet2.P
type = Prop

[var_31]
path = Jet2.P
name = We
defined_in = Prop

[var_30]
path = Jet2.P
name = D
defined_in = Prop

[comp_1]
resfuncs = func_2
name = R
local_resvars = var_1
objfuncs = func_1
coupling_vars = var_7, var_6, var_5, var_4, var_3, var_2
path = Jet2.R
type = Range

[link_9]
defined_in = syst_1
coupling = var_3, var_27
path = Jet2

[link_8]
defined_in = syst_1
coupling = var_7, var_10
path = Jet2

[link_3]
defined_in = syst_2
coupling = var_23, var_13
path = Jet2.SA

[link_2]
defined_in = syst_2
coupling = var_22, var_12
path = Jet2.SA

[link_1]
defined_in = syst_2
coupling = var_16, var_9
path = Jet2.SA

[link_7]
defined_in = syst_1
coupling = var_2, var_17
path = Jet2

[link_6]
defined_in = syst_1
coupling = var_2, var_26
path = Jet2

[link_5]
defined_in = syst_1
coupling = var_4, var_19
path = Jet2

[link_4]
defined_in = syst_2
coupling = var_24, var_14
path = Jet2.SA
```

Functional dependence table:

```
Columns:
 1: Jet2.R.LD
 2: Jet2.SA.A.LD
 3: Jet2.R.h
 4: Jet2.P.h
 5: Jet2.SA.A.h
 6: Jet2.R.Wf
 7: Jet2.SA.S.Wf
 8: Jet2.R.M
 9: Jet2.P.M
10: Jet2.SA.A.M
11: Jet2.R.Wt
12: Jet2.SA.S.Wt
13: Jet2.R.SFC
14: Jet2.P.SFC
15: Jet2.SA.A.D
16: Jet2.P.D
17: Jet2.SA.A.ESF
18: Jet2.P.ESF
19: Jet2.P.We
20: Jet2.SA.S.We
21: Jet2.R.R
22: Jet2.P.xp
23: Jet2.SA.A.ysa
24: Jet2.SA.S.ysa
25: Jet2.SA.A.L
26: Jet2.SA.S.L
27: Jet2.SA.A.Wt
28: Jet2.SA.A.tw
29: Jet2.SA.S.tw
30: Jet2.SA.S.xs
31: Jet2.SA.A.xa

Rows:
 1: Jet2.R.LD -- Jet2.SA.A.LD
 2: Jet2.R.h -- Jet2.P.h
 3: Jet2.R.h -- Jet2.SA.A.h
 4: Jet2.R.Wf -- Jet2.SA.S.Wf
 5: Jet2.R.M -- Jet2.P.M
 6: Jet2.R.M -- Jet2.SA.A.M
 7: Jet2.R.Wt -- Jet2.SA.S.Wt
 8: Jet2.R.SFC -- Jet2.P.SFC
```

```
 9: Jet2.SA.A.D -- Jet2.P.D
10: Jet2.SA.A.ESF -- Jet2.P.ESF
11: Jet2.P.We -- Jet2.SA.S.We
12: Jet2.R.fr
13: Jet2.R.ar
14: Jet2.P.gp
15: Jet2.P.ap
16: Jet2.SA.A.ysa -- Jet2.SA.S.ysa
17: Jet2.SA.A.L -- Jet2.SA.S.L
18: Jet2.SA.A.Wt -- Jet2.SA.S.Wt
19: Jet2.SA.A.tw -- Jet2.SA.S.tw
20: Jet2.SA.S.gs
21: Jet2.SA.S.as
22: Jet2.SA.A.ga
23: Jet2.SA.A.aa

FDT:
1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
0, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
0, 0, 1, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
0, 0, 0, 0, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
1, 0, 1, 0, 0, 1, 0, 1, 0, 0, 1, 0, 1, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
0, 0, 0, 1, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0
0, 0, 0, 1, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 1, 0, 1, 0, 1, 1, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 0, 0, 0, 0
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 0, 0
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 1, 0, 1, 0, 0, 0, 1, 0, 0
0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 1, 0, 1, 0, 0, 0, 1, 0
0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 1, 0, 1, 0, 0, 0, 1
0, 0, 0, 0, 1, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 1, 1, 0, 0
0, 1, 0, 0, 1, 0, 0, 0, 0, 1, 0, 0, 0, 0, 1, 0, 1, 0, 0, 0, 0, 0, 1, 0, 1, 0, 1, 1, 0, 0, 1
```

Fourth partition

Partition specification in Ψ:

```
comp Struc =
|[  extvar ysa, Wf, We, L, Wt, tw
    intvar xs
    confunc gs(ysa, xs, We, L)
    resfunc (Wt, Wf, tw) = as(ysa, xs, We, L)
]|

comp Aero =
|[  extvar ysa, h, M, LD, ESF, D, L, Wt, tw
    intvar xa
    confunc ga(ysa, xa, h, M, ESF, Wt, tw)
    resfunc (L, LD, D) = aa(ysa, xa, h, M, ESF, Wt, tw)
]|

comp Prop =
|[  extvar h, M, SFC, ESF, D, We
    intvar xp
    confunc gp(xp, h, M, D)
    resfunc (ESF, We, SFC) = ap(xp, h, M, D)
]|

syst Jet3 =
|[  intvar R
```

```
    sub S: Struc, A: Aero, P: Prop
    objfunc fr(R)
    resfunc R= ar(A.h, A.M, A.LD, P.SFC, S.Wt, S.Wf)
    link A.h   -- P.h   , A.M   -- P.M
       , A.D   -- P.D   , A.ESF -- P.ESF
       , A.ysa -- S.ysa , A.L   -- S.L
       , A.Wt  -- S.Wt  , A.tw  -- S.tw
       , P.We  -- S.We

]|

topsyst Jet3
```

Generated normalized partition:

```
[topsyst]
system = syst_1

[func_8]
path = Jet3
resvars = var_1
argvars = var_11, var_12, var_13, var_22, var_7, var_4
name = ar
defined_in = Jet3

[func_5]
path = Jet3.P
argvars = var_19, var_20, var_21, var_24
name = gp
defined_in = Prop

[func_4]
path = Jet3.A
resvars = var_16, var_13, var_15
argvars = var_10, var_9, var_11, var_12, var_14, var_17, var_18
name = aa
defined_in = Aero

[func_7]
path = Jet3
argvars = var_1
name = fr
defined_in = Jet3

[func_6]
path = Jet3.P
resvars = var_23, var_25, var_22
argvars = var_19, var_20, var_21, var_24
name = ap
defined_in = Prop

[func_1]
path = Jet3.S
argvars = var_3, var_2, var_5, var_6
name = gs
defined_in = Struc

[func_3]
path = Jet3.A
argvars = var_10, var_9, var_11, var_12, var_14, var_17, var_18
name = ga
defined_in = Aero

[func_2]
path = Jet3.S
resvars = var_7, var_4, var_8
argvars = var_3, var_2, var_5, var_6
name = as
defined_in = Struc

[var_24]
path = Jet3.P
name = D
defined_in = Prop

[var_25]
path = Jet3.P
name = We
defined_in = Prop

[var_22]
path = Jet3.P
name = SFC
defined_in = Prop

[var_23]
path = Jet3.P
name = ESF
defined_in = Prop

[var_20]
path = Jet3.P
name = h
defined_in = Prop

[var_21]
path = Jet3.P
name = M
defined_in = Prop

[var_9]
path = Jet3.A
name = xa
defined_in = Aero

[var_8]
path = Jet3.S
name = tw
defined_in = Struc

[var_7]
path = Jet3.S
name = Wt
defined_in = Struc

[var_6]
path = Jet3.S
name = L
defined_in = Struc

[var_5]
path = Jet3.S
name = We
defined_in = Struc

[var_4]
path = Jet3.S
name = Wf
defined_in = Struc

[var_3]
path = Jet3.S
name = ysa
defined_in = Struc

[var_2]
path = Jet3.S
name = xs
defined_in = Struc

[var_1]
path = Jet3
name = R
defined_in = Jet3

[var_17]
path = Jet3.A
name = Wt
defined_in = Aero

[var_16]
path = Jet3.A
name = L
defined_in = Aero
```

```
[var_15]
path = Jet3.A
name = D
defined_in = Aero

[var_14]
path = Jet3.A
name = ESF
defined_in = Aero

[var_13]
path = Jet3.A
name = LD
defined_in = Aero

[var_12]
path = Jet3.A
name = M
defined_in = Aero

[var_11]
path = Jet3.A
name = h
defined_in = Aero

[var_10]
path = Jet3.A
name = ysa
defined_in = Aero

[var_19]
path = Jet3.P
name = xp
defined_in = Prop

[var_18]
path = Jet3.A
name = tw
defined_in = Aero

[comp_2]
resfuncs = func_4
name = A
local_resvars = var_13
coupling_resvars = var_16, var_15
local_vars = var_9
coupling_vars = var_17, var_14, var_12, var_11, var_10, var_18
confuncs = func_3
path = Jet3.A
type = Aero

[comp_3]
resfuncs = func_6
name = P
local_resvars = var_22
coupling_resvars = var_25, var_23
local_vars = var_19
coupling_vars = var_24, var_20, var_21
confuncs = func_5
path = Jet3.P
type = Prop

[syst_1]
resfuncs = func_8
name = Jet3
links = link_1, link_2, link_3, link_4, link_5, link_6, link_7, link_8, link_9
local_resvars = var_1
objfuncs = func_7
path = Jet3
type = Jet3
subs = comp_1, comp_2, comp_3

[link_9]
defined_in = syst_1
coupling = var_25, var_5
path = Jet3

[link_8]
defined_in = syst_1
coupling = var_18, var_8
path = Jet3

[comp_1]
resfuncs = func_2
name = S
local_resvars = var_4
coupling_resvars = var_7, var_8
local_vars = var_2
coupling_vars = var_6, var_5, var_3
confuncs = func_1
path = Jet3.S
type = Struc

[link_3]
defined_in = syst_1
coupling = var_15, var_24
path = Jet3

[link_2]
defined_in = syst_1
coupling = var_12, var_21
path = Jet3

[link_1]
defined_in = syst_1
coupling = var_11, var_20
path = Jet3

[link_7]
defined_in = syst_1
coupling = var_17, var_7
path = Jet3

[link_6]
defined_in = syst_1
coupling = var_16, var_6
path = Jet3

[link_5]
defined_in = syst_1
coupling = var_10, var_3
path = Jet3

[link_4]
defined_in = syst_1
coupling = var_14, var_23
path = Jet3
```

Functional dependence table:

```
Columns:
 1: Jet3.R
 2: Jet3.A.h
 3: Jet3.A.M
 4: Jet3.A.LD
 5: Jet3.P.SFC
 6: Jet3.S.Wt
 7: Jet3.S.Wf
 8: Jet3.P.h
 9: Jet3.P.M
10: Jet3.A.D
11: Jet3.P.D
12: Jet3.A.ESF
13: Jet3.P.ESF
14: Jet3.A.ysa
15: Jet3.S.ysa
16: Jet3.A.L
17: Jet3.S.L
18: Jet3.A.Wt
19: Jet3.A.tw
20: Jet3.S.tw
21: Jet3.P.We
22: Jet3.S.We
23: Jet3.S.xs
24: Jet3.A.xa
25: Jet3.P.xp
```

```
Rows:
 1: Jet3.fr
 2: Jet3.ar
 3: Jet3.A.h -- Jet3.P.h
 4: Jet3.A.M -- Jet3.P.M
 5: Jet3.A.D -- Jet3.P.D
 6: Jet3.A.ESF -- Jet3.P.ESF
 7: Jet3.A.ysa -- Jet3.S.ysa
 8: Jet3.A.L -- Jet3.S.L
 9: Jet3.A.Wt -- Jet3.S.Wt
10: Jet3.A.tw -- Jet3.S.tw
11: Jet3.P.We -- Jet3.S.We
12: Jet3.S.gs
13: Jet3.S.as
14: Jet3.A.ga
15: Jet3.A.aa
16: Jet3.P.gp
17: Jet3.P.ap

FDT:
1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
1, 1, 1, 1, 1, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
0, 1, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
0, 0, 1, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0
0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 0, 0, 0, 0
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 0, 0
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 1, 0, 0, 0, 0, 1, 1, 0, 0
0, 0, 0, 0, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 1, 0, 1, 0, 0, 1, 0, 1, 1, 0, 0, 0
0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 1, 0, 0, 0, 1, 1, 0, 0, 0, 0, 1, 0, 0
0, 1, 1, 1, 0, 0, 0, 0, 0, 1, 0, 1, 0, 1, 0, 1, 0, 1, 1, 0, 0, 0, 0, 1, 0, 0
0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1
0, 0, 0, 0, 1, 0, 0, 1, 1, 0, 1, 0, 1, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 1, 0, 1
```

7 Micro-accelerometer

First partition

Partition specification in Ψ:

```
comp Circuit =
|[  extvar Sm, anm, Sd, Cp, Vs0
    intvar Ca, Gni, Sc, S, an, afs
    confunc gdevice(S, an, afs)
          , gcirc(Ca, Cp)
    resfunc (S, an, afs) = adevice(Sm, Sd, Sc, anm, Vs0)
          , Sc = acirc(Vs0, Ca, Gni)
]|

comp Dynamics =
|[  extvar Sm, anm, m, J, b, kxm, kxe, ky, kth
    intvar wx, wy, wth, w3dB
    confunc gdyn(wx, wy, wth, w3dB)
    resfunc (Sm, anm, wx, wy, wth, w3dB) = adyn(m, J, b, kxm, kxe, ky, kth)
]|

comp Geometry =
|[  extvar m, J, b, kxm, kxe, ky, kth, Sd, Cp, Vs0
    intvar A, lp, wp, lb1, lb2, wb, wb2, lf, lov, gs, gsu, gf, gfu, gx, wf, ws, Vd
         , amaxpi, amaxls, amaxst, xpull, l, wspr, wfin, lfin
```

```
    objfunc f(A)
    confunc gmass(lp, wp, lf, lov, gs, gsu, gf, gfu, gx, wf, ws)
          , gspr(lp, wp, lb1, lb2, wb, wb2)
          , garea(gx, lb2, l, wspr, wfin, lfin, A)
          , gelec(gs, gx, xpull, kxe, kxm, amaxpi, amaxls, amaxst)
    resfunc (A, m, J, b, wfin, lfin) = amass(lp, wp, lf, lov, gs, gsu, gf, gfu, gx, wf, ws)
          , (kxm, ky, kth, wspr, l) = aspr(lp, wp, lb1, lb2, wb, wb2)
          , (Sd, kxe, amaxpi, amaxls, amaxst, xpull, Cp)
                = aelec(m, kxm, lp, wp, lf, lov, gs, gf, gx, wf, ws, Vs0, Vd)
]|

syst Accelerometer =
|[  sub C: Circuit, D: Dynamics, G: Geometry
    link C.Sm -- D.Sm , C.anm -- D.anm
       , C.Sd -- G.Sd , C.Vs0 -- G.Vs0
       , C.Cp -- G.Cp , D.kxm -- G.kxm
       , D.m  -- G.m  , D.kxe -- G.kxe
       , D.J  -- G.J  , D.ky  -- G.ky
       , D.b  -- G.b  , D.kth -- G.kth
]|

topsyst Accelerometer
```

Generated normalized partition:

```
[var_48]
path = Accelerometer.G
name = wfin
defined_in = Geometry

[var_49]
path = Accelerometer.G
name = lfin
defined_in = Geometry

[var_44]
path = Accelerometer.G
name = amaxst
defined_in = Geometry

[var_45]
path = Accelerometer.G
name = xpull
defined_in = Geometry

[var_46]
path = Accelerometer.G
name = l
defined_in = Geometry

[var_47]
path = Accelerometer.G
name = wspr
defined_in = Geometry

[var_40]
path = Accelerometer.G
name = ws
defined_in = Geometry

[var_41]
path = Accelerometer.G
name = Vd
defined_in = Geometry

[var_42]
path = Accelerometer.G
name = amaxpi
defined_in = Geometry

[var_43]
path = Accelerometer.G
name = amaxls
defined_in = Geometry

[func_11]
path = Accelerometer.G
argvars = var_34, var_38, var_45, var_54, var_53, var_42, var_43, var_44
name = gelec
defined_in = Geometry

[func_10]
path = Accelerometer.G
argvars = var_38, var_29, var_46, var_47, var_48, var_49, var_25
name = garea
defined_in = Geometry

[func_13]
path = Accelerometer.G
resvars = var_53, var_55, var_56, var_47, var_46
argvars = var_26, var_27, var_28, var_29, var_30, var_31
name = aspr
defined_in = Geometry

[func_12]
path = Accelerometer.G
resvars = var_25, var_50, var_51, var_52, var_48, var_49
argvars = var_26, var_27, var_32, var_33, var_34, var_35, var_36, var_37, var_38, var_39, var_40
name = amass
defined_in = Geometry

[func_14]
path = Accelerometer.G
resvars = var_57, var_54, var_42, var_43, var_44, var_45, var_58
argvars = var_50, var_53, var_26, var_27, var_32, var_33, var_34, var_36, var_38, var_39, var_40, var_5
name = aelec
defined_in = Geometry

[var_9]
path = Accelerometer.C
name = Sd
defined_in = Circuit

[var_8]
path = Accelerometer.C
name = anm
defined_in = Circuit

[var_7]
path = Accelerometer.C
name = Sm
defined_in = Circuit

[var_6]
path = Accelerometer.C
name = afs
defined_in = Circuit
```

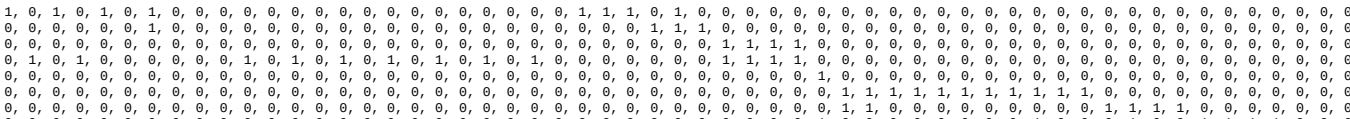
```
[var_5]
path = Accelerometer.C
name = an
defined_in = Circuit

[var_4]
path = Accelerometer.C
name = S
defined_in = Circuit

[var_3]
path = Accelerometer.C
name = Sc
defined_in = Circuit

[var_2]
path = Accelerometer.C
name = Gni
defined_in = Circuit

[var_1]
path = Accelerometer.C
name = Ca
defined_in = Circuit

[var_59]
path = Accelerometer.G
name = Vs0
defined_in = Geometry

[var_58]
path = Accelerometer.G
name = Cp
defined_in = Geometry

[var_53]
path = Accelerometer.G
name = kxm
defined_in = Geometry

[var_52]
path = Accelerometer.G
name = b
defined_in = Geometry

[var_51]
path = Accelerometer.G
name = J
defined_in = Geometry

[var_50]
path = Accelerometer.G
name = m
defined_in = Geometry

[var_57]
path = Accelerometer.G
name = Sd
defined_in = Geometry

[var_56]
path = Accelerometer.G
name = kth
defined_in = Geometry

[var_55]
path = Accelerometer.G
name = ky
defined_in = Geometry

[var_54]
path = Accelerometer.G
name = kxe
defined_in = Geometry

[func_9]
path = Accelerometer.G
argvars = var_26, var_27, var_28, var_29, var_30, var_31
name = gspr
defined_in = Geometry

[func_8]
path = Accelerometer.G
argvars = var_26, var_27, var_32, var_33, var_34, var_35, var_36,
name = gmass
defined_in = Geometry

[func_5]
path = Accelerometer.D
argvars = var_12, var_13, var_14, var_15
name = gdyn
defined_in = Dynamics

[func_4]
path = Accelerometer.C
resvars = var_3
argvars = var_11, var_1, var_2
name = acirc
defined_in = Circuit

[func_7]
path = Accelerometer.G
argvars = var_25
name = f
defined_in = Geometry

[func_6]
path = Accelerometer.D
resvars = var_16, var_17, var_12, var_13, var_14, var_15
argvars = var_18, var_19, var_20, var_21, var_22, var_23, var_24
name = adyn
defined_in = Dynamics

[func_1]
path = Accelerometer.C
argvars = var_4, var_5, var_6
name = gdevice
defined_in = Circuit

[func_3]
path = Accelerometer.C
resvars = var_4, var_5, var_6
argvars = var_7, var_9, var_3, var_8, var_11
name = adevice
defined_in = Circuit

[func_2]
path = Accelerometer.C
argvars = var_1, var_10
name = gcirc
defined_in = Circuit

[var_28]
path = Accelerometer.G
name = lb1
defined_in = Geometry

[var_29]
path = Accelerometer.G
name = lb2
defined_in = Geometry

[var_26]
path = Accelerometer.G
name = lp
defined_in = Geometry

[var_27]
path = Accelerometer.G
name = wp
defined_in = Geometry

[var_24]
path = Accelerometer.D
name = kth
defined_in = Dynamics

[var_25]
path = Accelerometer.G
name = A
defined_in = Geometry

[var_22]
path = Accelerometer.D
name = kxe
defined_in = Dynamics

[var_23]
path = Accelerometer.D
name = ky
defined_in = Dynamics

[var_20]
path = Accelerometer.D
name = b
defined_in = Dynamics

[var_21]
path = Accelerometer.D
name = kxm
defined_in = Dynamics

[var_39]
path = Accelerometer.G
name = wf
defined_in = Geometry
var_37, var_38, var_39, var_40
[var_38]
path = Accelerometer.G
name = gx
defined_in = Geometry

[var_35]
path = Accelerometer.G
name = gsu
defined_in = Geometry

[var_34]
path = Accelerometer.G
name = gs
defined_in = Geometry

[var_37]
path = Accelerometer.G
name = gfu
```

```
defined_in = Geometry

[var_36]
path = Accelerometer.G
name = gf
defined_in = Geometry

[var_31]
path = Accelerometer.G
name = wb2
defined_in = Geometry

[var_30]
path = Accelerometer.G
name = wb
defined_in = Geometry

[var_33]
path = Accelerometer.G
name = lov
defined_in = Geometry

[var_32]
path = Accelerometer.G
name = lf
defined_in = Geometry

[link_9]
defined_in = syst_1
coupling = var_19, var_51
path = Accelerometer

[link_8]
defined_in = syst_1
coupling = var_22, var_54
path = Accelerometer

[link_3]
defined_in = syst_1
coupling = var_9, var_57
path = Accelerometer

[link_2]
defined_in = syst_1
coupling = var_8, var_17
path = Accelerometer

[link_1]
defined_in = syst_1
coupling = var_7, var_16
path = Accelerometer

[link_7]
defined_in = syst_1
coupling = var_18, var_50
path = Accelerometer

[link_6]
defined_in = syst_1
coupling = var_21, var_53
path = Accelerometer

[link_5]
defined_in = syst_1
coupling = var_10, var_58
path = Accelerometer

[link_4]
defined_in = syst_1
coupling = var_11, var_59
path = Accelerometer

[topsyst]
system = syst_1

[var_17]
path = Accelerometer.D
name = anm
defined_in = Dynamics

[var_16]
path = Accelerometer.D
name = Sm
defined_in = Dynamics

[var_15]
path = Accelerometer.D
name = w3dB
defined_in = Dynamics

[var_14]
path = Accelerometer.D
name = wth
defined_in = Dynamics

[var_13]
path = Accelerometer.D
name = wy
defined_in = Dynamics

[var_12]
path = Accelerometer.D
name = wx
defined_in = Dynamics

[var_11]
path = Accelerometer.C
name = Vs0
defined_in = Circuit

[var_10]
path = Accelerometer.C
name = Cp
defined_in = Circuit

[var_19]
path = Accelerometer.D
name = J
defined_in = Dynamics

[var_18]
path = Accelerometer.D
name = m
defined_in = Dynamics

[syst_1]
path = Accelerometer
type = Accelerometer
name = Accelerometer
links = link_1, link_2, link_3, link_4, link_5, link_6, link_7, link_8, link_9, link_10, link_11, link_
subs = comp_1, comp_2, comp_3

[link_12]
defined_in = syst_1
coupling = var_24, var_56
path = Accelerometer

[link_11]
defined_in = syst_1
coupling = var_20, var_52
path = Accelerometer

[link_10]
defined_in = syst_1
coupling = var_23, var_55
path = Accelerometer

[comp_2]
resfuncs = func_6
name = D
local_resvars = var_15, var_14, var_13, var_12
coupling_resvars = var_17, var_16
coupling_vars = var_19, var_18, var_24, var_22, var_23, var_20, var_21
confuncs = func_5
path = Accelerometer.D
type = Dynamics

[comp_3]
resfuncs = func_12, func_13, func_14
name = G
local_resvars = var_48, var_49, var_44, var_45, var_46, var_47, var_42, var_43, var_25
objfuncs = func_7
coupling_resvars = var_58, var_53, var_52, var_51, var_50, var_57, var_56, var_55, var_54
local_vars = var_34, var_40, var_41, var_35, var_39, var_38, var_28, var_29, var_26, var_27, var_37, va
coupling_vars = var_59
confuncs = func_8, func_9, func_10, func_11
path = Accelerometer.G
type = Geometry

[comp_1]
resfuncs = func_3, func_4
name = C
local_resvars = var_6, var_5, var_4, var_3
local_vars = var_2, var_1
coupling_vars = var_7, var_8, var_9, var_11, var_10
confuncs = func_1, func_2
path = Accelerometer.C
type = Circuit
```

Functional dependence table:

```
Columns:
```

```
 1: Accelerometer.C.Sm
 2: Accelerometer.D.Sm
 3: Accelerometer.C.anm
 4: Accelerometer.D.anm
 5: Accelerometer.C.Sd
 6: Accelerometer.G.Sd
 7: Accelerometer.C.Vs0
 8: Accelerometer.G.Vs0
 9: Accelerometer.C.Cp
10: Accelerometer.G.Cp
11: Accelerometer.D.kxm
12: Accelerometer.G.kxm
13: Accelerometer.D.m
14: Accelerometer.G.m
15: Accelerometer.D.kxe
16: Accelerometer.G.kxe
17: Accelerometer.D.J
18: Accelerometer.G.J
19: Accelerometer.D.ky
20: Accelerometer.G.ky
21: Accelerometer.D.b
22: Accelerometer.G.b
23: Accelerometer.D.kth
24: Accelerometer.G.kth
25: Accelerometer.C.S
26: Accelerometer.C.an
27: Accelerometer.C.afs
28: Accelerometer.C.Ca
29: Accelerometer.C.Sc
30: Accelerometer.C.Gni
31: Accelerometer.D.wx
32: Accelerometer.D.wy
33: Accelerometer.D.wth
34: Accelerometer.D.w3dB
35: Accelerometer.G.A
36: Accelerometer.G.lp
37: Accelerometer.G.wp
38: Accelerometer.G.lf
39: Accelerometer.G.lov
40: Accelerometer.G.gs
41: Accelerometer.G.gsu
42: Accelerometer.G.gf
43: Accelerometer.G.gfu
44: Accelerometer.G.gx
45: Accelerometer.G.wf
46: Accelerometer.G.ws
47: Accelerometer.G.lb1
48: Accelerometer.G.lb2
49: Accelerometer.G.wb
50: Accelerometer.G.wb2
51: Accelerometer.G.l
52: Accelerometer.G.wspr
53: Accelerometer.G.wfin
54: Accelerometer.G.lfin
55: Accelerometer.G.xpull
56: Accelerometer.G.amaxpi
57: Accelerometer.G.amaxls
58: Accelerometer.G.amaxst
59: Accelerometer.G.Vd

Rows:
 1: Accelerometer.C.Sm -- Accelerometer.D.Sm
 2: Accelerometer.C.anm -- Accelerometer.D.anm
 3: Accelerometer.C.Sd -- Accelerometer.G.Sd
 4: Accelerometer.C.Vs0 -- Accelerometer.G.Vs0
 5: Accelerometer.C.Cp -- Accelerometer.G.Cp
 6: Accelerometer.D.kxm -- Accelerometer.G.kxm
 7: Accelerometer.D.m -- Accelerometer.G.m
 8: Accelerometer.D.kxe -- Accelerometer.G.kxe
 9: Accelerometer.D.J -- Accelerometer.G.J
10: Accelerometer.D.ky -- Accelerometer.G.ky
11: Accelerometer.D.b -- Accelerometer.G.b
12: Accelerometer.D.kth -- Accelerometer.G.kth
13: Accelerometer.C.gdevice
14: Accelerometer.C.gcirc
15: Accelerometer.C.adevice
16: Accelerometer.C.acirc
17: Accelerometer.D.gdyn
18: Accelerometer.D.adyn
19: Accelerometer.G.f
20: Accelerometer.G.gmass
21: Accelerometer.G.gspr
22: Accelerometer.G.garea
23: Accelerometer.G.gelec
24: Accelerometer.G.amass
25: Accelerometer.G.aspr
26: Accelerometer.G.aelec

FDT:
1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
0, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
0, 0, 0, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
```



Second partition

Partition specification in Ψ:

```
comp Circuit =
|[  extvar Sm, anm, Sd, Cp, Vs0
    intvar Ca, Gni, Sc, S, an, afs
    confunc gdevice(S, an, afs)
          , gcirc(Ca, Cp)
    resfunc (S, an, afs) = adevice(Sm, Sd, Sc, anm, Vs0)
          , Sc = acirc(Vs0, Ca, Gni)
]|

comp Dynamics =
|[  extvar Sm, anm, m, J, b, kxm, kxe, ky, kth
    intvar wx, wy, wth, w3dB
    confunc gdyn(wx, wy, wth, w3dB)
    resfunc (Sm, anm, wx, wy, wth, w3dB) = adyn(m, J, b, kxm, kxe, ky, kth)
]|

comp Mechanics =
|[  extvar m, J, b, kxm, ky, kth, lp, wp, lf, lov, gx, wf, ws, gs, gf
    intvar A, lb1, lb2, wb, wb2, gsu, gfu, l, wspr, wfin, lfin
    objfunc f(A)
    confunc gmass(lp, wp, lf, lov, gs, gsu, gf, gfu, gx, wf, ws)
          , gspr(lp, wp, lb1, lb2, wb, wb2)
          , garea(gx, lb2, l, wspr, wfin, lfin, A)
    resfunc (A, m, J, b, wfin, lfin) = amass(lp, wp, lf, lov, gs, gsu, gf, gfu, gx, wf, ws)
          , (kxm, ky, kth, wspr, l) = aspr(lp, wp, lb1, lb2, wb, wb2)
]|

comp Electrostatics =
|[  extvar m, kxm, kxe, Sd, Cp, Vs0, lp, wp, lf, lov, gs, gf, gx, wf, ws
    intvar Vd, amaxpi, amaxls, amaxst, xpull
    confunc gelec(gs, gx, xpull, kxe, kxm, amaxpi, amaxls, amaxst)
    resfunc (Sd, kxe, amaxpi, amaxls, amaxst, xpull, Cp)
                = aelec(m, kxm, lp, wp, lf, lov, gs, gf, gx, wf, ws, Vs0, Vd)
]|

syst Accelerometer2 =
|[  sub C: Circuit, D: Dynamics, M: Mechanics, E: Electrostatics
    link M.m  -- {D.m, E.m} , M.kxm -- {D.kxm, E.kxm}
```

```
       , C.Sm -- D.Sm      , C.anm -- D.anm
       , C.Sd -- E.Sd       , D.kxe -- E.kxe
       , C.Cp -- E.Cp       , C.Vs0 -- E.Vs0
       , M.lp -- E.lp       , D.ky  -- M.ky
       , M.lf -- E.lf       , D.kth -- M.kth
       , M.gs -- E.gs       , M.lov -- E.lov
       , M.gf -- E.gf       , M.wp  -- E.wp
       , M.gx -- E.gx       , M.wf  -- E.wf
       , D.b  -- M.b        , M.ws  -- E.ws
       , D.J  -- M.J
]|

topsyst Accelerometer2
```

Generated normalized partition:

```
[var_48]
path = Accelerometer2.M
name = ws
defined_in = Mechanics

[var_49]
path = Accelerometer2.M
name = gs
defined_in = Mechanics

[var_44]
path = Accelerometer2.M
name = lf
defined_in = Mechanics

[var_45]
path = Accelerometer2.M
name = lov
defined_in = Mechanics

[var_46]
path = Accelerometer2.M
name = gx
defined_in = Mechanics

[var_47]
path = Accelerometer2.M
name = wf
defined_in = Mechanics

[var_40]
path = Accelerometer2.M
name = ky
defined_in = Mechanics

[var_41]
path = Accelerometer2.M
name = kth
defined_in = Mechanics

[var_42]
path = Accelerometer2.M
name = lp
defined_in = Mechanics

[var_43]
path = Accelerometer2.M
name = wp
defined_in = Mechanics

[func_11]
path = Accelerometer2.M
resvars = var_25, var_36, var_37, var_38, var_34, var_35
argvars = var_42, var_43, var_44, var_45, var_49, var_30, var_50, var_31, var_46, var_47, var_48
name = amass
defined_in = Mechanics

[func_10]
path = Accelerometer2.M
argvars = var_46, var_27, var_32, var_33, var_34, var_35, var_25
name = garea
defined_in = Mechanics

[func_13]
path = Accelerometer2.E
argvars = var_66, var_68, var_55, var_58, var_57, var_52, var_53, var_54
name = gelec
defined_in = Electrostatics

[func_12]
path = Accelerometer2.M
resvars = var_39, var_40, var_41, var_33, var_32
argvars = var_42, var_43, var_26, var_27, var_28, var_29
name = aspr
defined_in = Mechanics

[func_14]
path = Accelerometer2.E
resvars = var_59, var_58, var_52, var_53, var_54, var_55, var_60
argvars = var_56, var_57, var_62, var_63, var_64, var_65, var_66, var_67, var_68, var_69, var_
name = aelec
defined_in = Electrostatics

[var_9]
path = Accelerometer2.C
name = Sd
defined_in = Circuit

[var_8]
path = Accelerometer2.C
name = anm
defined_in = Circuit

[var_7]
path = Accelerometer2.C
name = Sm
defined_in = Circuit

[var_6]
path = Accelerometer2.C
name = afs
defined_in = Circuit

[var_5]
path = Accelerometer2.C
name = an
defined_in = Circuit

[var_4]
path = Accelerometer2.C
name = S
defined_in = Circuit

[var_3]
path = Accelerometer2.C
name = Sc
defined_in = Circuit

[var_2]
path = Accelerometer2.C
name = Gni
defined_in = Circuit

[var_1]
path = Accelerometer2.C
name = Ca
defined_in = Circuit

[var_59]
path = Accelerometer2.E
name = Sd
defined_in = Electrostatics

[var_58]
path = Accelerometer2.E
name = kxe
```

```
defined_in = Electrostatics

[var_53]
path = Accelerometer2.E
name = amaxls
defined_in = Electrostatics

[var_52]
path = Accelerometer2.E
name = amaxpi
defined_in = Electrostatics

[var_51]
path = Accelerometer2.E
name = Vd
defined_in = Electrostatics

[var_50]
path = Accelerometer2.M
name = gf
defined_in = Mechanics

[var_57]
path = Accelerometer2.E
name = kxm
defined_in = Electrostatics

[var_56]
path = Accelerometer2.E
name = m
defined_in = Electrostatics

[var_55]
path = Accelerometer2.E
name = xpull
defined_in = Electrostatics

[var_54]
path = Accelerometer2.E
name = amaxst
defined_in = Electrostatics

[func_9]
path = Accelerometer2.M
argvars = var_42, var_43, var_26, var_27, var_28, var_29
name = gspr
defined_in = Mechanics

[func_8]
path = Accelerometer2.M
argvars = var_42, var_43, var_44, var_45, var_49, var_30, var_50,
var_31, var_46, var_47, var_48
name = gmass
defined_in = Mechanics

[link_12]
defined_in = syst_1
coupling = var_23, var_40
path = Accelerometer2

[func_5]
path = Accelerometer2.D
argvars = var_12, var_13, var_14, var_15
name = gdyn
defined_in = Dynamics

[func_4]
path = Accelerometer2.C
resvars = var_3
argvars = var_11, var_1, var_2
name = acirc
defined_in = Circuit

[func_7]
path = Accelerometer2.M
argvars = var_25
name = f
defined_in = Mechanics

[func_6]
path = Accelerometer2.D
resvars = var_16, var_17, var_12, var_13, var_14, var_15
argvars = var_18, var_19, var_20, var_21, var_22, var_23, var_24
name = adyn
defined_in = Dynamics

[func_1]
path = Accelerometer2.C
argvars = var_4, var_5, var_6
name = gdevice
defined_in = Circuit

[func_3]
path = Accelerometer2.C
resvars = var_4, var_5, var_6
argvars = var_7, var_9, var_3, var_8, var_11
name = adevice
defined_in = Circuit

[func_2]
path = Accelerometer2.C
argvars = var_1, var_10
name = gcirc
defined_in = Circuit

[var_28]
path = Accelerometer2.M
name = wb
defined_in = Mechanics

[var_29]
path = Accelerometer2.M
name = wb2
defined_in = Mechanics

[var_26]
path = Accelerometer2.M
name = lb1
defined_in = Mechanics

[var_27]
path = Accelerometer2.M
name = lb2
defined_in = Mechanics

[var_24]
path = Accelerometer2.D
name = kth
defined_in = Dynamics

[var_25]
path = Accelerometer2.M
name = A
defined_in = Mechanics

[var_22]
path = Accelerometer2.D
name = kxe
defined_in = Dynamics

[var_23]
path = Accelerometer2.D
name = ky
defined_in = Dynamics

[var_20]
path = Accelerometer2.D
name = b
defined_in = Dynamics

[var_21]
path = Accelerometer2.D
name = kxm
defined_in = Dynamics
[var_39]
path = Accelerometer2.M
name = kxm
defined_in = Mechanics

[var_38]
path = Accelerometer2.M
name = b
defined_in = Mechanics

[var_35]
path = Accelerometer2.M
name = lfin
defined_in = Mechanics

[var_34]
path = Accelerometer2.M
name = wfin
defined_in = Mechanics

[var_37]
path = Accelerometer2.M
name = J
defined_in = Mechanics

[var_36]
path = Accelerometer2.M
name = m
defined_in = Mechanics
[var_31]
path = Accelerometer2.M
name = gfu
defined_in = Mechanics

[var_30]
path = Accelerometer2.M
name = gsu
defined_in = Mechanics

[var_33]
path = Accelerometer2.M
name = wspr
defined_in = Mechanics

[var_32]
path = Accelerometer2.M
name = l
defined_in = Mechanics

[link_9]
defined_in = syst_1
```

```
coupling = var_10, var_60
path = Accelerometer2

[link_8]
defined_in = syst_1
coupling = var_22, var_58
path = Accelerometer2

[link_3]
defined_in = syst_1
coupling = var_39, var_57
path = Accelerometer2

[link_2]
defined_in = syst_1
coupling = var_36, var_56
path = Accelerometer2

[link_1]
defined_in = syst_1
coupling = var_36, var_18
path = Accelerometer2

[link_7]
defined_in = syst_1
coupling = var_9, var_59
path = Accelerometer2

[link_6]
defined_in = syst_1
coupling = var_8, var_17
path = Accelerometer2

[link_5]
defined_in = syst_1
coupling = var_7, var_16
path = Accelerometer2

[link_4]
defined_in = syst_1
coupling = var_39, var_21
path = Accelerometer2

[topsyst]
system = syst_1

[link_22]
defined_in = syst_1
coupling = var_48, var_70
path = Accelerometer2

[link_23]
defined_in = syst_1
coupling = var_19, var_37
path = Accelerometer2

[link_20]
defined_in = syst_1
coupling = var_47, var_69
path = Accelerometer2

[var_17]
path = Accelerometer2.D
name = anm
defined_in = Dynamics

[var_16]
path = Accelerometer2.D
name = Sm
defined_in = Dynamics

[var_15]
path = Accelerometer2.D
name = w3dB
defined_in = Dynamics

[var_14]
path = Accelerometer2.D
name = wth
defined_in = Dynamics

[var_13]
path = Accelerometer2.D
name = wy
defined_in = Dynamics

[var_12]
path = Accelerometer2.D
name = wx
defined_in = Dynamics

[var_11]
path = Accelerometer2.C
name = Vs0
defined_in = Circuit

[var_10]
path = Accelerometer2.C
name = Cp
defined_in = Circuit

[var_19]
path = Accelerometer2.D
name = J
defined_in = Dynamics

[var_18]
path = Accelerometer2.D
name = m
defined_in = Dynamics

[syst_1]
path = Accelerometer2
type = Accelerometer2
name = Accelerometer2
links = link_1, link_2, link_3, link_4, link_5, link_6, link_7, link_8, link_9, link_10, link_
subs = comp_1, comp_2, comp_3, comp_4

[link_21]
defined_in = syst_1
coupling = var_20, var_38
path = Accelerometer2

[var_62]
path = Accelerometer2.E
name = lp
defined_in = Electrostatics

[var_63]
path = Accelerometer2.E
name = wp
defined_in = Electrostatics

[var_60]
path = Accelerometer2.E
name = Cp
defined_in = Electrostatics

[var_61]
path = Accelerometer2.E
name = Vs0
defined_in = Electrostatics

[var_66]
path = Accelerometer2.E
name = gs
defined_in = Electrostatics

[var_67]
path = Accelerometer2.E
name = gf
defined_in = Electrostatics

[var_64]
path = Accelerometer2.E
name = lf
defined_in = Electrostatics

[var_65]
path = Accelerometer2.E
name = lov
defined_in = Electrostatics

[var_68]
path = Accelerometer2.E
name = gx
defined_in = Electrostatics

[var_69]
path = Accelerometer2.E
name = wf
defined_in = Electrostatics

[link_13]
defined_in = syst_1
coupling = var_44, var_64
path = Accelerometer2

[var_70]
path = Accelerometer2.E
name = ws
defined_in = Electrostatics

[link_11]
defined_in = syst_1
coupling = var_42, var_62
path = Accelerometer2

[link_10]
defined_in = syst_1
coupling = var_11, var_61
path = Accelerometer2

[link_17]
defined_in = syst_1
coupling = var_50, var_67
path = Accelerometer2

[link_16]
defined_in = syst_1
coupling = var_45, var_65
path = Accelerometer2

[link_15]
defined_in = syst_1
coupling = var_49, var_66
```

```
path = Accelerometer2

[link_14]
defined_in = syst_1
coupling = var_24, var_41
path = Accelerometer2

[link_19]
defined_in = syst_1
coupling = var_46, var_68
path = Accelerometer2

[link_18]
defined_in = syst_1
coupling = var_43, var_63
path = Accelerometer2

[comp_4]
resfuncs = func_14
name = E
local_resvars = var_53, var_52, var_55, var_54
coupling_resvars = var_59, var_58, var_60
local_vars = var_51
coupling_vars = var_62, var_63, var_70, var_61, var_66, var_67, var_64, var_65, var_68, var_69, var_57, var_56
confuncs = func_13
path = Accelerometer2.E
type = Electrostatics

[comp_2]
resfuncs = func_6
name = D
local_resvars = var_15, var_14, var_13, var_12
coupling_resvars = var_17, var_16
coupling_vars = var_19, var_18, var_24, var_22, var_23, var_20, var_21
confuncs = func_5
path = Accelerometer2.D
type = Dynamics

[comp_3]
resfuncs = func_11, func_12
name = M
local_resvars = var_35, var_34, var_25, var_33, var_32
objfuncs = func_7
coupling_resvars = var_40, var_41, var_39, var_38, var_37, var_36
local_vars = var_28, var_29, var_26, var_27, var_31, var_30
coupling_vars = var_50, var_48, var_49, var_44, var_45, var_46, var_47, var_42, var_43
confuncs = func_8, func_9, func_10
path = Accelerometer2.M
type = Mechanics

[comp_1]
resfuncs = func_3, func_4
name = C
local_resvars = var_4, var_3
local_vars = var_2, var_1
coupling_vars = var_7, var_8, var_9, var_11, var_10
confuncs = func_1, func_2
path = Accelerometer2.C
type = Circuit
```

Functional dependence table:

```
Columns:
 1: Accelerometer2.M.m
 2: Accelerometer2.D.m
 3: Accelerometer2.E.m
 4: Accelerometer2.M.kxm
 5: Accelerometer2.E.kxm
 6: Accelerometer2.D.kxm
 7: Accelerometer2.C.Sm
 8: Accelerometer2.D.Sm
 9: Accelerometer2.C.anm
10: Accelerometer2.D.anm
11: Accelerometer2.C.Sd
12: Accelerometer2.E.Sd
13: Accelerometer2.D.kxe
14: Accelerometer2.E.kxe
15: Accelerometer2.C.Cp
16: Accelerometer2.E.Cp
17: Accelerometer2.C.Vs0
18: Accelerometer2.E.Vs0
19: Accelerometer2.M.lp
20: Accelerometer2.E.lp
21: Accelerometer2.D.ky
22: Accelerometer2.M.ky
23: Accelerometer2.M.lf
24: Accelerometer2.E.lf
25: Accelerometer2.D.kth
26: Accelerometer2.M.kth
27: Accelerometer2.M.gs
28: Accelerometer2.E.gs
29: Accelerometer2.M.lov
30: Accelerometer2.E.lov
31: Accelerometer2.M.gf
32: Accelerometer2.E.gf
33: Accelerometer2.M.wp
34: Accelerometer2.E.wp
35: Accelerometer2.M.gx
36: Accelerometer2.E.gx
37: Accelerometer2.M.wf
38: Accelerometer2.E.wf
39: Accelerometer2.D.b
40: Accelerometer2.M.b
41: Accelerometer2.M.ws
42: Accelerometer2.E.ws
43: Accelerometer2.D.J
44: Accelerometer2.M.J
45: Accelerometer2.C.S
46: Accelerometer2.C.an
47: Accelerometer2.C.afs
48: Accelerometer2.C.Ca
49: Accelerometer2.C.Sc
50: Accelerometer2.C.Gni
51: Accelerometer2.D.wx
52: Accelerometer2.D.wy
53: Accelerometer2.D.wth
54: Accelerometer2.D.w3dB
55: Accelerometer2.M.A
56: Accelerometer2.M.gsu
57: Accelerometer2.M.gfu
58: Accelerometer2.M.lb1
59: Accelerometer2.M.lb2
```

```
60: Accelerometer2.M.wb
61: Accelerometer2.M.wb2
62: Accelerometer2.M.l
63: Accelerometer2.M.wspr
64: Accelerometer2.M.wfin
65: Accelerometer2.M.lfin
66: Accelerometer2.E.xpull
67: Accelerometer2.E.amaxpi
68: Accelerometer2.E.amaxls
69: Accelerometer2.E.amaxst
70: Accelerometer2.E.Vd

Rows:
 1: Accelerometer2.M.m -- Accelerometer2.D.m
 2: Accelerometer2.M.m -- Accelerometer2.E.m
 3: Accelerometer2.M.kxm -- Accelerometer2.E.kxm
 4: Accelerometer2.M.kxm -- Accelerometer2.D.kxm
 5: Accelerometer2.C.Sm -- Accelerometer2.D.Sm
 6: Accelerometer2.C.anm -- Accelerometer2.D.anm
 7: Accelerometer2.C.Sd -- Accelerometer2.E.Sd
 8: Accelerometer2.D.kxe -- Accelerometer2.E.kxe
 9: Accelerometer2.C.Cp -- Accelerometer2.E.Cp
10: Accelerometer2.C.Vs0 -- Accelerometer2.E.Vs0
11: Accelerometer2.M.lp -- Accelerometer2.E.lp
12: Accelerometer2.D.ky -- Accelerometer2.M.ky
13: Accelerometer2.M.lf -- Accelerometer2.E.lf
14: Accelerometer2.D.kth -- Accelerometer2.M.kth
15: Accelerometer2.M.gs -- Accelerometer2.E.gs
16: Accelerometer2.M.lov -- Accelerometer2.E.lov
17: Accelerometer2.M.gf -- Accelerometer2.E.gf
18: Accelerometer2.M.wp -- Accelerometer2.E.wp
19: Accelerometer2.M.gx -- Accelerometer2.E.gx
20: Accelerometer2.M.wf -- Accelerometer2.E.wf
21: Accelerometer2.D.b -- Accelerometer2.M.b
22: Accelerometer2.M.ws -- Accelerometer2.E.ws
23: Accelerometer2.D.J -- Accelerometer2.M.J
24: Accelerometer2.C.gdevice
25: Accelerometer2.C.gcirc
26: Accelerometer2.C.adevice
27: Accelerometer2.C.acirc
28: Accelerometer2.D.gdyn
29: Accelerometer2.D.adyn
30: Accelerometer2.M.f
31: Accelerometer2.M.gmass
32: Accelerometer2.M.gspr
33: Accelerometer2.M.garea
34: Accelerometer2.M.amass
35: Accelerometer2.M.aspr
36: Accelerometer2.E.gelec
37: Accelerometer2.E.aelec

FDT:
1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
1, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
0, 0, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
0, 0, 0, 1, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 1, 0, 0, 0, 0, 0, 0, 0
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0
0, 0, 0, 0, 0, 0, 1, 0, 1, 0, 1, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 1, 0, 1, 0, 0, 0, 0, 0
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 1, 0, 0, 0, 0
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 1, 1
0, 1, 0, 0, 0, 1, 0, 1, 0, 1, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 1, 1, 1, 1
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 1, 0, 0, 0, 1, 0, 1, 0, 1, 0, 1, 0, 1, 0, 1, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 1, 0, 0, 0, 1, 0, 1, 0, 1, 0, 1, 0, 1, 0, 1, 0, 0, 1, 1, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 1, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
0, 0, 1, 0, 1, 0, 0, 0, 0, 0, 0, 1, 0, 1, 0, 1, 0, 1, 0, 1, 0, 0, 0, 1, 0, 0, 0, 1, 0, 1, 0, 1, 0, 1, 0, 1, 0, 1, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
```

Third partition

Partition specification in Ψ:

```
comp Circuit =
|[  extvar Sm, anm, Sd, Cp, Vs0
    intvar Ca, Gni, Sc, S, an, afs
    confunc gdevice(S, an, afs)
          , gcirc(Ca, Cp)
    resfunc (S, an, afs) = adevice(Sm, Sd, Sc, anm, Vs0)
          , Sc = acirc(Vs0, Ca, Gni)
]|

comp Dynamics =
|[  extvar Sm, anm, m, J, b, kxm, kxe, ky, kth
    intvar wx, wy, wth, w3dB
    confunc gdyn(wx, wy, wth, w3dB)
    resfunc (Sm, anm, wx, wy, wth, w3dB) = adyn(m, J, b, kxm, kxe, ky, kth)
]|

comp Spring =
|[  extvar kxm, ky, kth, lb2, lp, wp, l, wspr
    intvar lb1, wb, wb2
    confunc gspr(lp, wp, lb1, lb2, wb, wb2)
    resfunc (kxm, ky, kth, wspr, l) = aspr(lp, wp, lb1, lb2, wb, wb2)
]|

comp Mass =
|[  extvar A, m, J, b, kxm, kxe, Sd, Cp, Vs0, lp, wp, gx, wfin, lfin
    intvar lf, lov, gs, gsu, gf, gfu, wf, ws, Vd, amaxpi, amaxls, amaxst, xpull
    objfunc f(A)
    confunc gmass(lp, wp, lf, lov, gs, gsu, gf, gfu, gx, wf, ws)
          , gelec(gs, gx, xpull, kxe, kxm, amaxpi, amaxls, amaxst)
    resfunc (A, m, J, b, wfin, lfin) = amass(lp, wp, lf, lov, gs, gsu, gf, gfu, gx, wf, ws)
          , (Sd, kxe, amaxpi, amaxls, amaxst, xpull, Cp)
                = aelec(m, kxm, lp, wp, lf, lov, gs, gf, gx, wf, ws, Vs0, Vd)
]|

syst Accelerometer3 =
|[  sub C: Circuit, D: Dynamics, S: Spring, M: Mass
    confunc garea(M.gx, S.lb2, S.l, S.wspr, M.wfin, M.lfin, M.A)
    link S.kxm -- {D.kxm, M.kxm} , C.Vs0 -- M.Vs0
       , C.Sm -- D.Sm            , C.anm -- D.anm
       , C.Sd -- M.Sd            , D.kxe -- M.kxe
       , C.Cp -- M.Cp            , D.kth -- S.kth
       , S.wp -- M.wp            , D.ky  -- S.ky
       , S.lp -- M.lp            , D.J   -- M.J
       , D.m  -- M.m             , D.b   -- M.b
]|

topsyst Accelerometer3
```

Generated normalized partition:

```
[var_48]
path = Accelerometer3.M
name = xpull
defined_in = Mass

[var_49]
path = Accelerometer3.M
name = A
defined_in = Mass

[var_44]
path = Accelerometer3.M
name = Vd
defined_in = Mass

[var_45]
path = Accelerometer3.M
name = amaxpi
defined_in = Mass

[var_46]
path = Accelerometer3.M
name = amaxls
defined_in = Mass

[var_47]
path = Accelerometer3.M
name = amaxst
defined_in = Mass

[var_40]
path = Accelerometer3.M
name = gf
defined_in = Mass

[var_41]
path = Accelerometer3.M
name = gfu
defined_in = Mass

[var_42]
path = Accelerometer3.M
name = wf
defined_in = Mass

[var_43]
path = Accelerometer3.M
name = ws
defined_in = Mass

[func_11]
path = Accelerometer3.M
argvars = var_38, var_60, var_48, var_54, var_53, var_45, var_46, var_47
name = gelec
defined_in = Mass

[func_10]
path = Accelerometer3.M
argvars = var_58, var_59, var_36, var_37, var_38, var_39, var_40,
var_41, var_60, var_42, var_43
name = gmass
defined_in = Mass

[func_13]
path = Accelerometer3.M
resvars = var_55, var_54, var_45, var_46, var_47, var_48, var_56
argvars = var_50, var_53, var_58, var_59, var_36, var_37, var_38,
var_40, var_60, var_42, var_43, var_57, var_44
name = aelec
defined_in = Mass

[func_12]
path = Accelerometer3.M
resvars = var_49, var_50, var_51, var_52, var_61, var_62
argvars = var_58, var_59, var_36, var_37, var_38, var_39, var_40,
var_41, var_60, var_42, var_43
name = amass
defined_in = Mass

[func_14]
path = Accelerometer3
argvars = var_60, var_31, var_34, var_35, var_61, var_62, var_49
name = garea
defined_in = Accelerometer3

[var_9]
path = Accelerometer3.C
name = Sd
defined_in = Circuit

[var_8]
path = Accelerometer3.C
name = anm
defined_in = Circuit

[var_7]
path = Accelerometer3.C
name = Sm
defined_in = Circuit

[var_6]
path = Accelerometer3.C
name = afs
defined_in = Circuit

[var_5]
path = Accelerometer3.C
name = an
defined_in = Circuit

[var_4]
path = Accelerometer3.C
name = S
defined_in = Circuit

[var_3]
path = Accelerometer3.C
name = Sc
defined_in = Circuit

[var_2]
path = Accelerometer3.C
name = Gni
defined_in = Circuit

[var_1]
path = Accelerometer3.C
name = Ca
defined_in = Circuit

[var_59]
path = Accelerometer3.M
name = wp
defined_in = Mass

[var_58]
path = Accelerometer3.M
name = lp
defined_in = Mass

[var_53]
path = Accelerometer3.M
name = kxm
defined_in = Mass

[var_52]
path = Accelerometer3.M
name = b
defined_in = Mass

[var_51]
path = Accelerometer3.M
name = J
defined_in = Mass

[var_50]
path = Accelerometer3.M
name = m
defined_in = Mass

[var_57]
path = Accelerometer3.M
name = Vs0
defined_in = Mass

[var_56]
path = Accelerometer3.M
name = Cp
defined_in = Mass

[var_55]
path = Accelerometer3.M
name = Sd
defined_in = Mass

[var_54]
path = Accelerometer3.M
name = kxe
defined_in = Mass

[func_9]
path = Accelerometer3.M
argvars = var_49
name = f
defined_in = Mass

[func_8]
path = Accelerometer3.S
resvars = var_28, var_29, var_30, var_35, var_34
argvars = var_32, var_33, var_25, var_31, var_26, var_27
name = aspr
defined_in = Spring

[func_5]
path = Accelerometer3.D
argvars = var_12, var_13, var_14, var_15
name = gdyn
defined_in = Dynamics

[func_4]
path = Accelerometer3.C
resvars = var_3
argvars = var_11, var_1, var_2
name = acirc
defined_in = Circuit

[func_7]
path = Accelerometer3.S
argvars = var_32, var_33, var_25, var_31, var_26, var_27
```

```
name = gspr
defined_in = Spring

[func_6]
path = Accelerometer3.D
resvars = var_16, var_17, var_12, var_13, var_14, var_15
argvars = var_18, var_19, var_20, var_21, var_22, var_23, var_24
name = adyn
defined_in = Dynamics

[func_1]
path = Accelerometer3.C
argvars = var_4, var_5, var_6
name = gdevice
defined_in = Circuit

[func_3]
path = Accelerometer3.C
resvars = var_4, var_5, var_6
argvars = var_7, var_9, var_3, var_8, var_11
name = adevice
defined_in = Circuit

[func_2]
path = Accelerometer3.C
argvars = var_1, var_10
name = gcirc
defined_in = Circuit

[var_28]
path = Accelerometer3.S
name = kxm
defined_in = Spring

[var_29]
path = Accelerometer3.S
name = ky
defined_in = Spring

[var_26]
path = Accelerometer3.S
name = wb
defined_in = Spring

[var_27]
path = Accelerometer3.S
name = wb2
defined_in = Spring

[var_24]
path = Accelerometer3.D
name = kth
defined_in = Dynamics

[var_25]
path = Accelerometer3.S
name = lb1
defined_in = Spring

[var_22]
path = Accelerometer3.D
name = kxe
defined_in = Dynamics

[var_23]
path = Accelerometer3.D
name = ky
defined_in = Dynamics

[var_20]
path = Accelerometer3.D
name = b
defined_in = Dynamics

[var_21]
path = Accelerometer3.D
name = kxm
defined_in = Dynamics

[var_39]
path = Accelerometer3.M
name = gsu
defined_in = Mass

[var_38]
path = Accelerometer3.M
name = gs
defined_in = Mass

[var_35]
path = Accelerometer3.S
name = wspr
defined_in = Spring

[var_34]
path = Accelerometer3.S
name = l
defined_in = Spring

[var_37]
path = Accelerometer3.M
name = lov
defined_in = Mass

[var_36]
path = Accelerometer3.M
name = lf
defined_in = Mass

[var_31]
path = Accelerometer3.S
name = lb2
defined_in = Spring

[var_30]
path = Accelerometer3.S
name = kth
defined_in = Spring

[var_33]
path = Accelerometer3.S
name = wp
defined_in = Spring

[var_32]
path = Accelerometer3.S
name = lp
defined_in = Spring

[link_9]
defined_in = syst_1
coupling = var_24, var_30
path = Accelerometer3

[link_8]
defined_in = syst_1
coupling = var_10, var_56
path = Accelerometer3

[link_3]
defined_in = syst_1
coupling = var_11, var_57
path = Accelerometer3

[link_2]
defined_in = syst_1
coupling = var_28, var_53
path = Accelerometer3

[link_1]
defined_in = syst_1
coupling = var_28, var_21
path = Accelerometer3

[link_7]
defined_in = syst_1
coupling = var_22, var_54
path = Accelerometer3

[link_6]
defined_in = syst_1
coupling = var_9, var_55
path = Accelerometer3

[link_5]
defined_in = syst_1
coupling = var_8, var_17
path = Accelerometer3

[link_4]
defined_in = syst_1
coupling = var_7, var_16
path = Accelerometer3

[topsyst]
system = syst_1

[var_17]
path = Accelerometer3.D
name = anm
defined_in = Dynamics

[var_16]
path = Accelerometer3.D
name = Sm
defined_in = Dynamics

[var_15]
path = Accelerometer3.D
name = w3dB
defined_in = Dynamics

[var_14]
path = Accelerometer3.D
name = wth
defined_in = Dynamics

[var_13]
path = Accelerometer3.D
name = wy
defined_in = Dynamics

[var_12]
path = Accelerometer3.D
name = wx
defined_in = Dynamics
```

```
[var_11]
path = Accelerometer3.C
name = Vs0
defined_in = Circuit

[var_10]
path = Accelerometer3.C
name = Cp
defined_in = Circuit

[var_19]
path = Accelerometer3.D
name = J
defined_in = Dynamics

[var_18]
path = Accelerometer3.D
name = m
defined_in = Dynamics

[syst_1]
subs = comp_1, comp_2, comp_3, comp_4
links = link_1, link_2, link_3, link_4, link_5, link_6, link_7, link_8, link_9, link_10, link_11, link_12, link_13, link_14, link_15
confuncs = func_14
path = Accelerometer3
type = Accelerometer3
name = Accelerometer3

[var_62]
path = Accelerometer3.M
name = lfin
defined_in = Mass

[var_60]
path = Accelerometer3.M
name = gx
defined_in = Mass

[var_61]
path = Accelerometer3.M
name = wfin
defined_in = Mass

[link_13]
defined_in = syst_1
coupling = var_19, var_51
path = Accelerometer3

[link_12]
defined_in = syst_1
coupling = var_32, var_58
path = Accelerometer3

[link_11]
defined_in = syst_1
coupling = var_23, var_29
path = Accelerometer3

[link_10]
defined_in = syst_1
coupling = var_33, var_59
path = Accelerometer3

[link_15]
defined_in = syst_1
coupling = var_20, var_52
path = Accelerometer3

[link_14]
defined_in = syst_1
coupling = var_18, var_50
path = Accelerometer3

[comp_4]
resfuncs = func_12, func_13
name = M
local_resvars = var_62, var_61, var_48, var_49, var_45, var_46, var_47
objfuncs = func_9
coupling_resvars = var_52, var_51, var_50, var_56, var_55, var_54
local_vars = var_60, var_44, var_40, var_41, var_42, var_43, var_39, var_38, var_37, var_36
coupling_vars = var_53, var_59, var_58, var_57
confuncs = func_10, func_11
path = Accelerometer3.M
type = Mass

[comp_2]
resfuncs = func_6
name = D
local_resvars = var_15, var_14, var_13, var_12
coupling_resvars = var_17, var_16
coupling_vars = var_19, var_18, var_24, var_22, var_23, var_20, var_21
confuncs = func_5
path = Accelerometer3.D
type = Dynamics

[comp_3]
resfuncs = func_8
name = S
local_resvars = var_35, var_34
coupling_resvars = var_30, var_28, var_29
local_vars = var_26, var_27, var_25, var_31
coupling_vars = var_33, var_32
confuncs = func_7
path = Accelerometer3.S
type = Spring

[comp_1]
resfuncs = func_3, func_4
name = C
local_resvars = var_6, var_5, var_4, var_3
local_vars = var_2, var_1
coupling_vars = var_7, var_8, var_11, var_9, var_10
confuncs = func_1, func_2
path = Accelerometer3.C
type = Circuit
```

Functional dependence table:

```
Columns:
 1: Accelerometer3.M.gx
 2: Accelerometer3.S.lb2
 3: Accelerometer3.S.l
 4: Accelerometer3.S.wspr
 5: Accelerometer3.M.wfin
 6: Accelerometer3.M.lfin
 7: Accelerometer3.M.A
 8: Accelerometer3.S.kxm
 9: Accelerometer3.D.kxm
10: Accelerometer3.M.kxm
11: Accelerometer3.C.Vs0
12: Accelerometer3.M.Vs0
13: Accelerometer3.C.Sm
14: Accelerometer3.D.Sm
15: Accelerometer3.C.anm
16: Accelerometer3.D.anm
17: Accelerometer3.C.Sd
18: Accelerometer3.M.Sd
19: Accelerometer3.D.kxe
20: Accelerometer3.M.kxe
21: Accelerometer3.C.Cp
22: Accelerometer3.M.Cp
23: Accelerometer3.D.kth
24: Accelerometer3.S.kth
25: Accelerometer3.S.wp
26: Accelerometer3.M.wp
27: Accelerometer3.D.ky
28: Accelerometer3.S.ky
29: Accelerometer3.S.lp
30: Accelerometer3.M.lp
31: Accelerometer3.D.J
```

```
32: Accelerometer3.M.J
33: Accelerometer3.D.m
34: Accelerometer3.M.m
35: Accelerometer3.D.b
36: Accelerometer3.M.b
37: Accelerometer3.C.S
38: Accelerometer3.C.an
39: Accelerometer3.C.afs
40: Accelerometer3.C.Ca
41: Accelerometer3.C.Sc
42: Accelerometer3.C.Gni
43: Accelerometer3.D.wx
44: Accelerometer3.D.wy
45: Accelerometer3.D.wth
46: Accelerometer3.D.w3dB
47: Accelerometer3.S.lb1
48: Accelerometer3.S.wb
49: Accelerometer3.S.wb2
50: Accelerometer3.M.lf
51: Accelerometer3.M.lov
52: Accelerometer3.M.gs
53: Accelerometer3.M.gsu
54: Accelerometer3.M.gf
55: Accelerometer3.M.gfu
56: Accelerometer3.M.wf
57: Accelerometer3.M.ws
58: Accelerometer3.M.xpull
59: Accelerometer3.M.amaxpi
60: Accelerometer3.M.amaxls
61: Accelerometer3.M.amaxst
62: Accelerometer3.M.Vd

Rows:
 1: Accelerometer3.garea
 2: Accelerometer3.S.kxm -- Accelerometer3.D.kxm
 3: Accelerometer3.S.kxm -- Accelerometer3.M.kxm
 4: Accelerometer3.C.Vs0 -- Accelerometer3.M.Vs0
 5: Accelerometer3.C.Sm -- Accelerometer3.D.Sm
 6: Accelerometer3.C.anm -- Accelerometer3.D.anm
 7: Accelerometer3.C.Sd -- Accelerometer3.M.Sd
 8: Accelerometer3.D.kxe -- Accelerometer3.M.kxe
 9: Accelerometer3.C.Cp -- Accelerometer3.M.Cp
10: Accelerometer3.D.kth -- Accelerometer3.S.kth
11: Accelerometer3.S.wp -- Accelerometer3.M.wp
12: Accelerometer3.D.ky -- Accelerometer3.S.ky
13: Accelerometer3.S.lp -- Accelerometer3.M.lp
14: Accelerometer3.D.J -- Accelerometer3.M.J
15: Accelerometer3.D.m -- Accelerometer3.M.m
16: Accelerometer3.D.b -- Accelerometer3.M.b
17: Accelerometer3.C.gdevice
18: Accelerometer3.C.gcirc
19: Accelerometer3.C.adevice
20: Accelerometer3.C.acirc
21: Accelerometer3.D.gdyn
22: Accelerometer3.D.adyn
23: Accelerometer3.S.gspr
24: Accelerometer3.S.aspr
25: Accelerometer3.M.f
26: Accelerometer3.M.gmass
27: Accelerometer3.M.gelec
28: Accelerometer3.M.amass
29: Accelerometer3.M.aelec

FDT:
1, 1, 1, 1, 1, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
0, 0, 0, 0, 0, 0, 0, 1, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 1, 0, 1, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 1, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 1, 0, 1, 0, 0, 1, 0, 0, 0, 1, 0, 0, 0, 1, 0, 0, 0, 1, 0, 1, 0, 1, 0, 0, 0, 0, 0, 0, 0, 1, 1, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0
0, 1, 1, 1, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0
0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 1, 1, 1, 1, 1, 1
1, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0
1, 0, 0, 0, 1, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 1, 0, 1, 0, 1, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 1, 1, 1, 1, 1, 1
1, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 1, 0, 0, 0, 0, 0, 1, 0, 1, 0, 1, 0, 0, 0, 1, 0, 0, 0, 1, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 1, 0, 1, 0, 1, 1
```

Fourth partition

Partition specification in Ψ:

```
comp Circuit2 =
|[  extvar Sc, Ca, Vs0
    intvar Gni
    resfunc Sc = acirc(Vs0, Ca, Gni)
]|

comp Dynamics =
|[  extvar Sm, anm, m, J, b, kxm, kxe, ky, kth
    intvar wx, wy, wth, w3dB
    confunc gdyn(wx, wy, wth, w3dB)
    resfunc (Sm, anm, wx, wy, wth, w3dB) = adyn(m, J, b, kxm, kxe, ky, kth)
]|

comp Spring =
|[  extvar kxm, ky, kth, lb2, lp, wp, l, wspr
    intvar lb1, wb, wb2
    confunc gspr(lp, wp, lb1, lb2, wb, wb2)
    resfunc (kxm, ky, kth, wspr, l) = aspr(lp, wp, lb1, lb2, wb, wb2)
]|

comp Mass =
|[  extvar A, m, J, b, kxm, kxe, Sd, Cp, Vs0, lp, wp, gx, wfin, lfin
    intvar lf, lov, gs, gsu, gf, gfu, wf, ws, Vd, amaxpi, amaxls, amaxst, xpull
    objfunc f(A)
    confunc gmass(lp, wp, lf, lov, gs, gsu, gf, gfu, gx, wf, ws)
          , gelec(gs, gx, xpull, kxe, kxm, amaxpi, amaxls, amaxst)
    resfunc (A, m, J, b, wfin, lfin) = amass(lp, wp, lf, lov, gs, gsu, gf, gfu, gx, wf, w
          , (Sd, kxe, amaxpi, amaxls, amaxst, xpull, Cp)
                = aelec(m, kxm, lp, wp, lf, lov, gs, gf, gx, wf, ws, Vs0, Vd)
]|

syst Accelerometer4 =
|[  intvar S, an, afs
    sub C: Circuit2, D: Dynamics, S: Spring, M: Mass
    confunc garea(M.gx, S.lb2, S.l, S.wspr, M.wfin, M.lfin, M.A)
          , gdevice(S, an, afs)
          , gcirc(C.Ca, M.Cp)
    resfunc (S, an, afs) = adevice(D.Sm, M.Sd, C.Sc, D.anm, C.Vs0)
    link S.kxm -- {D.kxm, M.kxm} , D.kxe -- M.kxe
       , S.wp  -- M.wp           , D.kth -- S.kth
       , S.lp  -- M.lp           , D.ky  -- S.ky
       , D.m   -- M.m            , D.J   -- M.J
       , D.b   -- M.b            , C.Vs0 -- M.Vs0
]|

topsyst Accelerometer4
```

Generated normalized partition:

```
[var_48]
path = Accelerometer4.M
name = b
defined_in = Mass

[var_49]
path = Accelerometer4.M
name = kxm
defined_in = Mass

[var_44]
path = Accelerometer4.M
name = xpull
defined_in = Mass

[var_45]
path = Accelerometer4.M
name = A
defined_in = Mass

[var_46]
path = Accelerometer4.M
name = m
defined_in = Mass

[var_47]
path = Accelerometer4.M
name = J
defined_in = Mass

[var_40]
path = Accelerometer4.M
name = Vd
defined_in = Mass

[var_41]
path = Accelerometer4.M
name = amaxpi
defined_in = Mass

[var_42]
path = Accelerometer4.M
name = amaxls
defined_in = Mass

[var_43]
path = Accelerometer4.M
name = amaxst
defined_in = Mass

[func_11]
path = Accelerometer4
argvars = var_56, var_27, var_30, var_31, var_57, var_58, var_45
name = garea
defined_in = Accelerometer4

[func_10]
path = Accelerometer4.M
resvars = var_51, var_50, var_41, var_42, var_43, var_44, var_52
argvars = var_46, var_49, var_54, var_55, var_32, var_33, var_34, var_36, var_56, var_38, var_39, var_53, var_40
name = aelec
defined_in = Mass

[func_13]
path = Accelerometer4
argvars = var_6, var_52
name = gcirc
defined_in = Accelerometer4

[func_12]
path = Accelerometer4
argvars = var_1, var_2, var_3
name = gdevice
defined_in = Accelerometer4

[func_14]
path = Accelerometer4
resvars = var_1, var_2, var_3
argvars = var_12, var_51, var_5, var_13, var_7
name = adevice
defined_in = Accelerometer4

[var_9]
path = Accelerometer4.D
name = wy
defined_in = Dynamics

[var_8]
path = Accelerometer4.D
name = wx
defined_in = Dynamics

[var_7]
path = Accelerometer4.C
name = Vs0
defined_in = Circuit2

[var_6]
path = Accelerometer4.C
name = Ca
defined_in = Circuit2

[var_5]
path = Accelerometer4.C
name = Sc
defined_in = Circuit2

[var_4]
path = Accelerometer4.C
name = Gni
defined_in = Circuit2

[var_3]
path = Accelerometer4
name = afs
defined_in = Accelerometer4

[var_2]
path = Accelerometer4
name = an
defined_in = Accelerometer4

[var_1]
path = Accelerometer4
name = S
defined_in = Accelerometer4

[var_58]
path = Accelerometer4.M
name = lfin
defined_in = Mass

[var_53]
path = Accelerometer4.M
name = Vs0
defined_in = Mass

[var_52]
path = Accelerometer4.M
name = Cp
defined_in = Mass

[var_51]
path = Accelerometer4.M
name = Sd
defined_in = Mass

[var_50]
path = Accelerometer4.M
name = kxe
defined_in = Mass

[var_57]
path = Accelerometer4.M
name = wfin
defined_in = Mass

[var_56]
path = Accelerometer4.M
name = gx
defined_in = Mass

[var_55]
path = Accelerometer4.M
name = wp
defined_in = Mass

[var_54]
path = Accelerometer4.M
name = lp
defined_in = Mass

[func_9]
path = Accelerometer4.M
resvars = var_45, var_46, var_47, var_48, var_57, var_58
argvars = var_54, var_55, var_32, var_33, var_34, var_35, var_36, var_37, var_56, var_38, var_39
name = amass
defined_in = Mass

[func_8]
path = Accelerometer4.M
argvars = var_34, var_56, var_44, var_50, var_49, var_41, var_42, var_43
name = gelec
defined_in = Mass

[func_5]
path = Accelerometer4.S
resvars = var_24, var_25, var_26, var_31, var_30
argvars = var_28, var_29, var_21, var_27, var_22, var_23
name = aspr
defined_in = Spring

[func_4]
path = Accelerometer4.S
argvars = var_28, var_29, var_21, var_27, var_22, var_23
name = gspr
defined_in = Spring
```

```
[func_7]
path = Accelerometer4.M
argvars = var_54, var_55, var_32, var_33, var_34, var_35, var_36, var_37, var_38, var_39
name = gmass
defined_in = Mass

[func_6]
path = Accelerometer4.M
argvars = var_45
name = f
defined_in = Mass

[func_1]
path = Accelerometer4.C
resvars = var_5
argvars = var_7, var_6, var_4
name = acirc
defined_in = Circuit2

[func_3]
path = Accelerometer4.D
resvars = var_12, var_13, var_8, var_9, var_10, var_11
argvars = var_14, var_15, var_16, var_17, var_18, var_19, var_20
name = adyn
defined_in = Dynamics

[func_2]
path = Accelerometer4.D
argvars = var_8, var_9, var_10, var_11
name = gdyn
defined_in = Dynamics

[var_28]
path = Accelerometer4.S
name = lp
defined_in = Spring

[var_29]
path = Accelerometer4.S
name = wp
defined_in = Spring

[var_26]
path = Accelerometer4.S
name = kth
defined_in = Spring

[var_27]
path = Accelerometer4.S
name = lb2
defined_in = Spring

[var_24]
path = Accelerometer4.S
name = kxm
defined_in = Spring

[var_25]
path = Accelerometer4.S
name = ky
defined_in = Spring

[var_22]
path = Accelerometer4.S
name = wb
defined_in = Spring

[var_23]
path = Accelerometer4.S
name = wb2
defined_in = Spring

[var_20]
path = Accelerometer4.D
name = kth
defined_in = Dynamics

[var_21]
path = Accelerometer4.S
name = lb1
defined_in = Spring

[var_39]
path = Accelerometer4.M
name = ws
defined_in = Mass

[var_38]
path = Accelerometer4.M
name = wf
defined_in = Mass

[var_35]
path = Accelerometer4.M
name = gsu
defined_in = Mass

[var_34]
path = Accelerometer4.M
name = gs
defined_in = Mass

[var_37]
path = Accelerometer4.M
name = gfu
defined_in = Mass

[var_36]
path = Accelerometer4.M
name = gf
defined_in = Mass

[var_31]
path = Accelerometer4.S
name = wspr
defined_in = Spring

[var_30]
path = Accelerometer4.S
name = l
defined_in = Spring

[var_33]
path = Accelerometer4.M
name = lov
defined_in = Mass

[var_32]
path = Accelerometer4.M
name = lf
defined_in = Mass

[link_9]
defined_in = syst_1
coupling = var_15, var_47
path = Accelerometer4

[link_8]
defined_in = syst_1
coupling = var_14, var_46
path = Accelerometer4

[link_3]
defined_in = syst_1
coupling = var_18, var_50
path = Accelerometer4

[link_2]
defined_in = syst_1
coupling = var_24, var_49
path = Accelerometer4

[link_1]
defined_in = syst_1
coupling = var_24, var_17
path = Accelerometer4

[link_7]
defined_in = syst_1
coupling = var_19, var_25
path = Accelerometer4

[link_6]
defined_in = syst_1
coupling = var_28, var_54
path = Accelerometer4

[link_5]
defined_in = syst_1
coupling = var_20, var_26
path = Accelerometer4

[link_4]
defined_in = syst_1
coupling = var_29, var_55
path = Accelerometer4

[topsyst]
system = syst_1

[var_17]
path = Accelerometer4.D
name = kxm
defined_in = Dynamics

[var_16]
path = Accelerometer4.D
name = b
defined_in = Dynamics

[var_15]
path = Accelerometer4.D
name = J
defined_in = Dynamics

[var_14]
path = Accelerometer4.D
name = m
defined_in = Dynamics

[var_13]
path = Accelerometer4.D
name = anm
defined_in = Dynamics

[var_12]
```

```
path = Accelerometer4.D
name = Sm
defined_in = Dynamics

[var_11]
path = Accelerometer4.D
name = w3dB
defined_in = Dynamics

[var_10]
path = Accelerometer4.D
name = wth
defined_in = Dynamics

[var_19]
path = Accelerometer4.D
name = ky
defined_in = Dynamics

[var_18]
path = Accelerometer4.D
name = kxe
defined_in = Dynamics

[syst_1]
resfuncs = func_14
name = Accelerometer4
links = link_1, link_2, link_3, link_4, link_5, link_6, link_7, link_8, link_9, link_10, link_11
local_resvars = var_3, var_2, var_1
path = Accelerometer4
confuncs = func_11, func_12, func_13
type = Accelerometer4
subs = comp_1, comp_2, comp_3, comp_4

[link_11]
defined_in = syst_1
coupling = var_7, var_53
path = Accelerometer4

[link_10]
defined_in = syst_1
coupling = var_16, var_48
path = Accelerometer4

[comp_4]
resfuncs = func_9, func_10
name = M
local_resvars = var_58, var_44, var_45, var_51, var_57, var_41, var_42, var_43, var_52
objfuncs = func_6
coupling_resvars = var_47, var_46, var_50, var_48
local_vars = var_40, var_56, var_39, var_38, var_35, var_34, var_37, var_36, var_33, var_32
coupling_vars = var_53, var_49, var_55, var_54
confuncs = func_7, func_8
path = Accelerometer4.M
type = Mass

[comp_2]
resfuncs = func_3
name = D
local_resvars = var_13, var_12, var_11, var_10, var_9, var_8
coupling_vars = var_17, var_16, var_15, var_14, var_19, var_18, var_20
confuncs = func_2
path = Accelerometer4.D
type = Dynamics

[comp_3]
resfuncs = func_5
name = S
local_resvars = var_31, var_30
coupling_resvars = var_26, var_24, var_25
local_vars = var_27, var_22, var_23, var_21
coupling_vars = var_28, var_29
confuncs = func_4
path = Accelerometer4.S
type = Spring

[comp_1]
resfuncs = func_1
name = C
local_resvars = var_5
local_vars = var_6, var_4
coupling_vars = var_7
path = Accelerometer4.C
type = Circuit2
```

Functional dependence table:

```
Columns:
 1: Accelerometer4.M.gx
 2: Accelerometer4.S.lb2
 3: Accelerometer4.S.l
 4: Accelerometer4.S.wspr
 5: Accelerometer4.M.wfin
 6: Accelerometer4.M.lfin
 7: Accelerometer4.M.A
 8: Accelerometer4.S
 9: Accelerometer4.an
10: Accelerometer4.afs
11: Accelerometer4.C.Ca
12: Accelerometer4.M.Cp
13: Accelerometer4.D.Sm
14: Accelerometer4.M.Sd
15: Accelerometer4.C.Sc
16: Accelerometer4.D.anm
17: Accelerometer4.C.Vs0
18: Accelerometer4.S.kxm
19: Accelerometer4.D.kxm
20: Accelerometer4.M.kxm
21: Accelerometer4.D.kxe
22: Accelerometer4.M.kxe
23: Accelerometer4.S.wp
24: Accelerometer4.M.wp
25: Accelerometer4.D.kth
26: Accelerometer4.S.kth
27: Accelerometer4.S.lp
28: Accelerometer4.M.lp
29: Accelerometer4.D.ky
30: Accelerometer4.S.ky
31: Accelerometer4.D.m
32: Accelerometer4.M.m
33: Accelerometer4.D.J
34: Accelerometer4.M.J
35: Accelerometer4.D.b
36: Accelerometer4.M.b
37: Accelerometer4.M.Vs0
38: Accelerometer4.C.Gni
39: Accelerometer4.D.wx
40: Accelerometer4.D.wy
41: Accelerometer4.D.wth
42: Accelerometer4.D.w3dB
43: Accelerometer4.S.lb1
44: Accelerometer4.S.wb
45: Accelerometer4.S.wb2
46: Accelerometer4.M.lf
47: Accelerometer4.M.lov
```

```
48: Accelerometer4.M.gs
49: Accelerometer4.M.gsu
50: Accelerometer4.M.gf
51: Accelerometer4.M.gfu
52: Accelerometer4.M.wf
53: Accelerometer4.M.ws
54: Accelerometer4.M.xpull
55: Accelerometer4.M.amaxpi
56: Accelerometer4.M.amaxls
57: Accelerometer4.M.amaxst
58: Accelerometer4.M.Vd

Rows:
 1: Accelerometer4.garea
 2: Accelerometer4.gdevice
 3: Accelerometer4.gcirc
 4: Accelerometer4.adevice
 5: Accelerometer4.S.kxm -- Accelerometer4.D.kxm
 6: Accelerometer4.S.kxm -- Accelerometer4.M.kxm
 7: Accelerometer4.D.kxe -- Accelerometer4.M.kxe
 8: Accelerometer4.S.wp -- Accelerometer4.M.wp
 9: Accelerometer4.D.kth -- Accelerometer4.S.kth
10: Accelerometer4.S.lp -- Accelerometer4.M.lp
11: Accelerometer4.D.ky -- Accelerometer4.S.ky
12: Accelerometer4.D.m -- Accelerometer4.M.m
13: Accelerometer4.D.J -- Accelerometer4.M.J
14: Accelerometer4.D.b -- Accelerometer4.M.b
15: Accelerometer4.C.Vs0 -- Accelerometer4.M.Vs0
16: Accelerometer4.C.acirc
17: Accelerometer4.D.gdyn
18: Accelerometer4.D.adyn
19: Accelerometer4.S.gspr
20: Accelerometer4.S.aspr
21: Accelerometer4.M.f
22: Accelerometer4.M.gmass
23: Accelerometer4.M.gelec
24: Accelerometer4.M.amass
25: Accelerometer4.M.aelec

FDT:
1, 1, 1, 1, 1, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
0, 0, 0, 0, 0, 0, 0, 1, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
0, 0, 0, 0, 0, 0, 0, 1, 1, 1, 0, 0, 1, 1, 1, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 1, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 1, 0, 0, 1, 0, 1, 0, 0, 0, 1, 0, 0, 0, 1, 0, 1, 0, 1, 0, 1, 0, 0, 0, 1, 1, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0,
0, 1, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 1, 0, 0, 1, 1, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0,
0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 1, 1, 1, 1, 1, 1,
1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0,
1, 0, 0, 0, 1, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 1, 0, 0, 0, 1, 0, 1, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 1, 1, 1, 1, 1, 1,
1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 1, 0, 0, 0, 0, 0, 1, 0, 1, 0, 1, 0, 0, 0, 1, 0, 0, 0, 1, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 1, 0, 1, 0, 1, 1,
```