Design Structure Matrix (DSM) methods and its application in system engineering

Who am I and why am I here?

- B.E. in Electrical Engineering, Nanjing University of Aeronautics and Astronautics (NUAA), 2004
- Ph.D. in Guidance Navigation & Control, NUAA, 2011
- Research fellow
  - in Australian Centre for Space Research Center, UNSW Sydney, 2011-2014.
  - in Capability Systems Center (CSC), UNSW Canberra, 2014 – now
- Current research area: modular analysis, trade space exploration.
- Design Structure Matrix (DSM) method has been a subject of research at the Capability Systems Centre
- Share some interesting findings about DSM methods with a range of example applications
  - Matrix-based methods to solve some system engineering problems
Research motivation

Modularity in Design

- What are designs?
  - Instructions that turn knowledge into things
  - Span all artifacts and human activities
  - Some new designs create value forces that can change the structure of an industry
  - Small designs can “just get done” by one person or a small team; large designs require architecture

- Design Rules: The Power of Modularity – C. Baldwin & K. Clark [Baldwin 2000] “…IBM gave us the prototype of modularized design…The IBM System/370 was the first modularized mainframe…Baldwin & Clark claim that their principles apply equally to social and legal institutions as well as technologies. Modularization could improve the design of almost everything.” – W. Sheridan
Modularity in Design

- Design Rule Idea: have components depend on design parameters guaranteed to change
- Architectural design
  - Subsystem identification: services and constraints are specified
  - Module design: modular decomposition is performed; relationships specified
- Studying the designs and correlating their changes with the value changes, the finding is that modularity held the key.
- Modularity in design is a financial force that can change the structure of an industry
- Its virtues:
  - Makes complexity manageable
  - Enables parallel work
  - Welcomes experimentation
  - Creates options

What we are doing?

- Success depends on designers’ intuition and experience. Designers need to reason consequences of a change, options to accommodate a change, refactor or not, etc.
- Look for adequate design representations/algorithms to visualize and measure modularity in design
  - To help “experts” play with ideas
  - To help “beginners” learn about design
- Seek for a method/tool/algorithm for formal modeling and automated analysis which are
  - General enough: span language paradigm and system lifecycle
  - Explicitly represent decision: Design is a decision-making procedure
  - Computable
  - Analyzable
  - Scalable
  - Capture the essence of informal principles
Design Structure Matrix (DSM)

- A matrix representation of a complex system.
  - Static DSM: Represent system elements existing simultaneously
  - Time-Based DSM: Ordering of rows and columns represent a flow through time: upstream activities in a process precede downstream activities
- Well known technique to help define design decisions, elements/components/parameters, interfaces and element relationships.
- The rows and columns of a DSM are labeled by the design components and dependences between two parameters are marked.
- Ability to modularize

Powerful systems are built of many elements; power comes from elements’ interplay.

Hand-held massager*

Massager DSM*

* M. Kashkoush and H. ElMaraghy. Optimum overall product modularity. Procedia CIRP

DSM categories and analysis methods*

<table>
<thead>
<tr>
<th>Categories</th>
<th>Data types</th>
<th>Representation</th>
<th>Application</th>
<th>Analysis methods</th>
<th>DSMs</th>
<th>Time-based DSM</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Component based</td>
<td>Team-based</td>
<td>Task-based</td>
<td>Task-based</td>
<td>Task/Activity input/output relationships</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Team interface characteristics: e.g. communication frequency</td>
<td>Team interface characteristics: e.g. communication frequency</td>
<td>Task-based</td>
<td>Task/Activity input/output relationships</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Task-based</td>
<td>Task-based</td>
<td>Task-based</td>
<td>Task/Activity input/output relationships</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Parameter-based</td>
<td>Parameter-based</td>
<td>Parameter-based</td>
<td>Parameter decision points and necessary precedents</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Parameter decision points and necessary precedents</td>
<td>Parameter decision points and necessary precedents</td>
<td>Parameter-based</td>
<td>Parameter decision points and necessary precedents</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>DSM clustering</td>
<td>DSM clustering</td>
<td>DSM clustering</td>
<td>DSM clustering</td>
<td>DSM sequencing</td>
</tr>
<tr>
<td></td>
<td></td>
<td>DSM clustering</td>
<td>DSM clustering</td>
<td>DSM clustering</td>
<td>DSM clustering</td>
<td>DSM sequencing</td>
</tr>
<tr>
<td></td>
<td></td>
<td>DSM sequencing</td>
<td>DSM sequencing</td>
<td>DSM sequencing</td>
<td>DSM sequencing</td>
<td>DSM sequencing</td>
</tr>
<tr>
<td></td>
<td></td>
<td>DSM sequencing</td>
<td>DSM sequencing</td>
<td>DSM sequencing</td>
<td>DSM sequencing</td>
<td>DSM sequencing</td>
</tr>
</tbody>
</table>

*Design Structure Matrix Methods and Applications, Book by Steven D. Eppinger and Tyson R. Browning
Design Structure Matrix (DSM)

• Our previous work* regarding MDS clustering provides a rigorous study of its application to DSM modular analysis.
• Tailor our previous algorithm for various problems such as the structuring of design problem.
• We found
  • The ease of creating and editing design parameters enables the freedom to make changes (erasing and re-writing the matrix)
  • DSM based algorithms (e.g. clustering, sequencing derived from data mining, machine learning, graph theory, network analysis etc.) make some things automated and easier to manipulate.

---

DSM Application Examples

---

---

Product-DSM clustering

**Algorithm and Procedure**

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>SMACOF hierarchical clustering with DSM</th>
</tr>
</thead>
<tbody>
<tr>
<td>Input:</td>
<td>DSM (an $n \times n$ matrix)</td>
</tr>
<tr>
<td>Output:</td>
<td>Dendrogram, Partitions for a range of $k$ with Cost, optimal partitions</td>
</tr>
<tr>
<td></td>
<td>If the entries are not numbers, then digitize the DSM, and if</td>
</tr>
<tr>
<td></td>
<td>Use SMACOF, obtain embedded data (an $n \times m$ matrix)</td>
</tr>
<tr>
<td></td>
<td>Compute the distance matrix using Cosine,</td>
</tr>
<tr>
<td></td>
<td>hierarchical clustering the embedded data,</td>
</tr>
<tr>
<td></td>
<td>obtain the partitions for a range of $k$</td>
</tr>
<tr>
<td></td>
<td>Calculate the Cost for a range of $k$</td>
</tr>
<tr>
<td></td>
<td>If the $P_{opt}$ exits = True</td>
</tr>
<tr>
<td></td>
<td>Calculate the Jaccard index between the $P_{opt}$ and the obtained partitions</td>
</tr>
<tr>
<td></td>
<td>end</td>
</tr>
<tr>
<td></td>
<td>If there are no design constraints then choose the optimal partition when $k = k^*$</td>
</tr>
<tr>
<td></td>
<td>else if there are design constraints, modify the partition</td>
</tr>
<tr>
<td></td>
<td>end if</td>
</tr>
</tbody>
</table>

Input is DSM5 a $5 \times 5$ matrix
Set $m = 2$, output is a $5 \times 2$ matrix.

Representation in new 2D by SMACOF
Algorithm and Procedure (cont.)

### Algorithm

**Input:**
- DSM and its digitalization

**Step 1:** DSM and its digitalization

**Step 2:** SMACOF

**Step 3:** Hierarchical clustering

**Step 4:** Evaluation and knowledge deployment

**Output:**
- DSM (an \( n \times n \) matrix)
- Dendrogram, Partitions for a range of \( k \) with Cost, optimal partitions

If the entries are not numbers, then digitize the DSM, end if.

Use SMACOF, obtain embedded data (an \( n \times m \) matrix)

Compute the distance matrix using Cosine, hierarchical clustering the embedded data, obtain the partitions for a range of \( k \)

Calculate the Cost for a range of \( k \)

If the \( P_{opt} \) exits = True

Calculate the Jaccard index between the \( P_{opt} \) and the obtained partitions

end

If there are no design constraints then choose the optimal partition when \( k = k^* \)

else if there are design constraints, modify the partition

end if

---

**The solutions of hierarchical clustering are called dendrogram.**

Represent nested clusters for DSM5.

A quantitative description of data properties.

A partition can be obtained by cutting the dendrogram at a certain level.

\[
P_{k=2} = \{ \{1, 3\}, \{2, 4, 5\} \} \\
P_{k=3} = \{ \{1, 3\}, \{4, 5\}, \{2\} \}
\]

---

**Algorithm and Procedure (cont.)**

**Input:**
- DSM (an \( n \times n \) matrix)
- Dendrogram, Partitions for a range of \( k \) with Cost, optimal partitions

If the entries are not numbers, then digitize the DSM, end if.

Use SMACOF, obtain embedded data (an \( n \times m \) matrix)

Compute the distance matrix using Cosine, hierarchical clustering the embedded data, obtain the partitions for a range of \( k \)

Calculate the Cost for a range of \( k \)

If the \( P_{opt} \) exits = True

Calculate the Jaccard index between the \( P_{opt} \) and the obtained partitions

end

If there are no design constraints then choose the optimal partition when \( k = k^* \)

else if there are design constraints, modify the partition

end if

**Compare the quality of the partition solutions using**

\[
\text{Cost} = \sum \text{IntraClusterCost} + \sum \text{ExtraClusterCost} \\
= \sum_{i,j \in \text{same cluster}} [\text{DSM}(i,j) + \text{DSM}(j,i)] \times d_{ij} + \sum_{i,j \notin \text{same cluster}} [\text{DSM}(i,j) + \text{DSM}(j,i)] \times n
\]

When we calculate the cost for a range of partitions, the cost value first decrease as the number of cluster increase. Then, the cost value increase, as a result of an increase in the number of interactions outside the partitions. The minimum value for the cost is considered as the optimal partition.

\[
P_{k=2} = \{ \{1, 3\}, \{2, 4, 5\} \} \quad \text{Cost} = 72 \\
P_{k=3} = \{ \{1, 3\}, \{4, 5\}, \{2\} \} \quad \text{Cost} = 78
\]

The optimal \( k^* = 2 \). If there are no design constraints, we suggest the solution preference is \( P_{k^* = 2} \).
Algorithm and Procedure (cont.)

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>SMCOF hierarchical clustering with DSM</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Input:</strong></td>
<td>DSM (an $n \times n$ matrix)</td>
</tr>
<tr>
<td><strong>Output:</strong></td>
<td>DSM and its digitalization</td>
</tr>
<tr>
<td><strong>Step 1:</strong></td>
<td>Compute the distance matrix using Cosine, where $C_{ij}$ is the number of members shared between both sets/the total number of members in both sets</td>
</tr>
<tr>
<td><strong>Step 2:</strong></td>
<td>DSM and its digitalization</td>
</tr>
<tr>
<td><strong>Step 3:</strong></td>
<td>Compute the distance matrix using Cosine, where $C_{ij}$ is the number of members shared between both sets/the total number of members in both sets</td>
</tr>
<tr>
<td><strong>Step 4:</strong></td>
<td>Evaluate and knowledge deployment</td>
</tr>
</tbody>
</table>

Use Jaccard index to compare partitions

$$Jaccard (X, Y) = \frac{|X \cap Y|}{|X \cup Y|}$$

The number of members shared between both sets/the total number of members in both sets

1) Comparison of the obtained partitions with the reference partition $P_{ref}$
2) Comparison within the obtained partitions $P_2, P_3, ..., P_{n-1}$

Product modularization (hydraulic support system)

Hydraulic support components

<table>
<thead>
<tr>
<th>No.</th>
<th>Component</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Base</td>
</tr>
<tr>
<td>2</td>
<td>Pushing bar</td>
</tr>
<tr>
<td>3</td>
<td>Rear bar</td>
</tr>
<tr>
<td>4</td>
<td>Cover jack</td>
</tr>
<tr>
<td>5</td>
<td>Balance jack</td>
</tr>
<tr>
<td>6</td>
<td>Head beam</td>
</tr>
<tr>
<td>7</td>
<td>Columns</td>
</tr>
<tr>
<td>8</td>
<td>Cliff board</td>
</tr>
<tr>
<td>9</td>
<td>Cliff jack</td>
</tr>
</tbody>
</table>

Cost (Hydraulic support system)

Original DSM

Referenced partitions

Optimal partitions with lowest cost
Product modularization (Bicycle)

Bicycle components

<table>
<thead>
<tr>
<th>No.</th>
<th>Name</th>
<th>Element</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Handlebar</td>
<td>11</td>
</tr>
<tr>
<td>2</td>
<td>Seat post</td>
<td>12</td>
</tr>
<tr>
<td>3</td>
<td>Seat tube</td>
<td>14</td>
</tr>
<tr>
<td>4</td>
<td>Top tube</td>
<td>15</td>
</tr>
<tr>
<td>5</td>
<td>Seat stay</td>
<td>16</td>
</tr>
<tr>
<td>6</td>
<td>Down tube</td>
<td>17</td>
</tr>
<tr>
<td>7</td>
<td>Chain stay</td>
<td>18</td>
</tr>
<tr>
<td>8</td>
<td>Fork</td>
<td>19</td>
</tr>
<tr>
<td>9</td>
<td>Front brake</td>
<td>20</td>
</tr>
<tr>
<td>10</td>
<td>Head tube</td>
<td>21</td>
</tr>
</tbody>
</table>

Cost = 1532  DSM21

Referenced partitions


Cluster 6 (1) Saddle [1]

Cluster 7 (1) Saddle [1]

Dynamic Modular Architecture For Product Lifecycle: DSM³

Clustering functions

Clustering utilities

Jaccard similarity index

Similarity matrix

Dynamic Modular Architecture For Product Lifecycle: DSM³ (cont.)

- **Type I** – clusters that are identical in both the DSM¹ and the DSM².
- **Type II** – large clusters that consist of a mixture of elements from entire smaller clusters from the other layer of the DSM³.
- **Type III** – clusters that exist in one DSM layer and not in the other.

Characterized by a value of ‘1’ in the similarity matrix, indicating that the two clusters have 100% similarity according to the Jaccard similarity coefficient. No costs are associated with the transition of these clusters.

Characterized in the similarity matrix by a row or column that consists of a single non-zero value. Cluster has a single non-zero index with cluster, indicating that it is contained entirely within the larger cluster. Cluster is entirely contained within cluster. Two costs associated with it: the disconnect cost between the smaller clusters and the neighbouring clusters, which ideally are weak, and the reconnecting cost of the elements in the larger cluster.

Elements from a cluster in one layer of the DSM³ are transferred to another cluster in the other layer. Two costs associated with each element’s transition: the disconnection cost of leaving the current cluster, and the connection cost for joining the elements in the new cluster.

The upgraded internal hub-gearing mechanism, replaces the traditional derailleur gearing system. The internal hub gearing is expensive but is simpler for regular maintenance, in particular when frequent assembly and disassembly of the rear wheel is required.

### Organization - DSM clustering
Supply Chain Outsourcing Problem

- A common strategic decision in the area of supply chain management (SCM). Despite obvious advantages, no simple guide as to when to rely on an in-house solution, versus outsourcing to a partner. The decision is therefore difficult as outsourcing can pose a threat to a corporation if the outsourcing relationship is not clearly defined and practised.

- It is reported that some 90% of manufacturing companies have some level of outsourcing.

- One of the basic outsourcing rules: If something is deemed a core competency, then you keep it in-house; everything else you outsource. The unique business functions that allow an organization to be successful.

Supply Chain model

A supply chain is a complex network. There are different types of relationships (or called interactions) among these components, such as information, materiel, function and financial.

What we do: developing a systematic approach to identify and quantify the interactions and cluster the large network to reasonable number of subgroups of core competencies and ancillary groups.
Various flows modelling of a supply chain DSM

Scales for determine the interaction strength

Original matrix: from Chen and Huang, 2007

Core competency index

Modular analysis of supply chain network

The segmented aggregated DSM with the lowest Cost value.

A diagonalization pattern
Support supply chain outsourcing decision making

This partitioning helps to identify the core competency elements in the supply chain system.

- **Group 1** is the core competency element as it has the Produce and Source subsystems. (We assume that the core competency uses up the company’s most resources.)

- **Group 4** (Return stage) contains the ancillary parts as they have less influence to affect the core product.

Supply Chain resource allocation problem

Network graph for the supply chain functional structure with 22 nodes

- 22 functional components
- 87 arcs directed between pairs of nodes
- The value of weight is taken from Chen 2007.
- The width of the edges are proportional to the weights

Algorithm: Centrality measures in graph theory and clustering algorithm

DSM + social network analysis (graph theory)

<table>
<thead>
<tr>
<th>Centrality measures</th>
<th>Values</th>
</tr>
</thead>
<tbody>
<tr>
<td>PageRank</td>
<td>3</td>
</tr>
<tr>
<td>Betweenness</td>
<td>2</td>
</tr>
<tr>
<td>OutCloseness</td>
<td>1</td>
</tr>
<tr>
<td>InCloseness</td>
<td>-1</td>
</tr>
<tr>
<td>OutDegree</td>
<td>-2</td>
</tr>
<tr>
<td>InDegree</td>
<td>-3</td>
</tr>
</tbody>
</table>

Different centralities distribution

If the distribution of the nodes is not even which suggests the existence of something interesting.

### E.g. Betweenness

- High betweenness nodes act as critical intermediates.
- Have the potential to influence others near them, through both direct and indirect pathways.
- The nodes with high betweenness and low degree mean their few ties are crucial for network flow.

<table>
<thead>
<tr>
<th>Network characteristics</th>
<th>Network interpretation</th>
<th>Resource recommendations</th>
<th>Example nodes</th>
</tr>
</thead>
<tbody>
<tr>
<td>High in-degree</td>
<td>Prestigious/dependent, critical and vulnerable</td>
<td>Upstream information alignment and sharing service</td>
<td>9, 4, 5, 10, 8, 11</td>
</tr>
<tr>
<td>High out-degree</td>
<td>Conductive, critical and vulnerable</td>
<td>Downstream information alignment and sharing service</td>
<td>2, 1, 18, 7</td>
</tr>
<tr>
<td>High out-degree, low in-degree</td>
<td>A beginning</td>
<td>Introductory resources, such as sales oriented information</td>
<td>1</td>
</tr>
<tr>
<td>High in-closeness</td>
<td>Highly accessible from all other nodes</td>
<td>Regulation to improve the access efficiency, such as new coordination and communication model</td>
<td>3, 17, 19, 18, 1</td>
</tr>
<tr>
<td>High out-closeness</td>
<td>Highly accessible to all other nodes, outbound gateway</td>
<td>Resources to facilitate the downstream flows</td>
<td>2, 15, 14, 10, 7, 12</td>
</tr>
<tr>
<td>High Betweenness</td>
<td>Critical intermediate</td>
<td>Service to control the flows across the network</td>
<td>1, 7, 2, 8, 12, 14</td>
</tr>
<tr>
<td>Low closeness, high in-degree</td>
<td>Far from the rest of the network</td>
<td>Opportunities for improvement</td>
<td>6</td>
</tr>
<tr>
<td>Low degree, low closeness, low betweenness</td>
<td>Border</td>
<td>Promotion-related information</td>
<td>20, 21, 22, 17, 18</td>
</tr>
<tr>
<td>High PageRank</td>
<td>Influential nodes</td>
<td>Customer fulfillment related information</td>
<td>8, 4, 5, 11, 7, 9</td>
</tr>
</tbody>
</table>
Team-based DSM clustering

Team-based satellite design problem

Binary Team-based satellite design problem. 16 subsystems/disciplines, marked as DSM16. X presents the interaction among members. Transfer it to a binary matrix. New-Girvan algorithm is adopted to obtain $P_{ref}\{\{1\}, \{2, 3, 4, 5, 6, 7\}, \{8, 9, 10, 11, 12, 13\}, \{14\}, \{15\}, \{16\}\}

Optimal partition with lower cost value of 264.

Dendrogram identify the related elements, such as Thermal[12] and Electrical power[14].
Team-based satellite design problem (cont.)

- The optimal number of cluster $k^* = 4$.
- Launch Vehicle (1), Integration and Test (15) and Parametric Cost (16) are included though there are not any dependencies to other disciplines. Set the design constraints here that (1), (15) and (16) must be located in independent groups. Then we revise the partition with the same lowest cost. $k^* = 6$
- Use a colour-coded graph to indicate the hierarchical modularity of the optimal partition.
- The teams with the same colour should work more closely together.

Activity-based DSM clustering
Activity-based DSM for aircraft design problem

Activity-based DSM15 for aircraft design problem.
15 elements, presenting various decision-making activities of the whole design problem.
Entities represent the information flow between elements.
Larger dots denote stronger coupling between the disciplines.
The dendrogram represents nested clusters for DSM.

Activity-based DSM for aircraft design problem(cont.)

- $P_{k=3}$ is not satisfied as element Optimization(1) is a bus-like element which has links to most of the rest activities.
- Look for a partition with low cost and in which the Optimization[1] is isolated.
- Revised our preferred partition to $P_{k=4}$.
- The rest elements group very well as very few interactions are outside the cluster.

Parameter-based DSM sequencing

- Project management is hard... We need tools to help...
- Network-based (graph theory) methods
  - CPM (Critical Path Method),
  - PERT (Program Evaluation and Review Technique)
- Design Structure Matrix: A system engineering tool.
  - Simple
  - Compact
  - Visual representation of a complex system
### 3 Possible Sequences for 2 tasks

- **Dependent (Series)**: A → B → A → B
- **Independent (Parallel)**: A → A → B → B
- **Interdependent (Coupled)**: A ↔ B ↔ A ↔ B

**Sequencing analysis**: analysis of the process DSM through logical ordering of the activities, identifying sequential, parallel, and coupled sets of activities.

### Satellite Formation flying design problem

- Parameter-based DSM
- Sequencing algorithm identifies the concurrent activities and cycles.
- Use the values of the marks to order to rows into a lower triangular form
- Division is based on the concurrency while the high level activities are grouped based on function

Matrices have 7 levels.
The coupled cycle highlight in pink

29 parameters in satellite formation flying problem
Satellite design tasks problem

The temperature (warm/cool) of the colour shows the precedence relationship between the elements. “2 Flight dynamics” has the warmest colour, which suggests its importance as the start work.

The rest 11 elements forms a big feedback loop. The elements in this loop have different colours, which means this is not a simple loop.
Satellite design tasks problem (cont.)

- The driver sector elements influence many other elements. The dependent sector elements are influenced by other elements. For ideal case, the warmer nodes are located in the upper left part and the cooler nodes are located in the lower right. The distribution of the dots indicates the degree of complexity. The satellite design problem is much more complex than the ideal case.

- These three isolated nodes (1,15,16) are located in the equal criticality line (Criticality=1). We can eliminate these nodes for investigation in the next step.

Conclusion of our DSM research

- Knowledge about the structure of a problem is important for deepening understanding of design.

- Useful tool to model analysis and manage the complexity of design problems.

- Simple and compact
  - The ease of creating and editing to make changes (erasing and re-writing the matrix)
  - The ease of manipulating the matrix (various DSM based algorithms can support its analysis such as clustering, sequencing derived from data mining, graph theory etc.)

- “This really should be done with a computer”. Think about DSM based method. It may help...