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

Never Stand Still UNSW Canberra School of Engineering and Information Telecommunication

Capability Systems Centre UNSW Canberra

Li Qiao
l.qiao@adfa.edu.au
Capability Systems Centre
School of Engineering and Information Technology
UNSW, Canberra

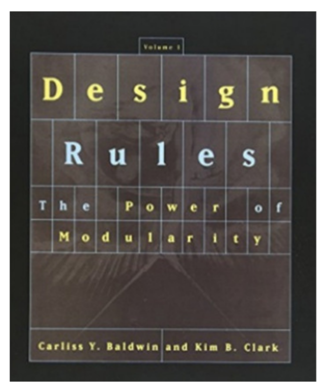
Who am I and why am I here?

- 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

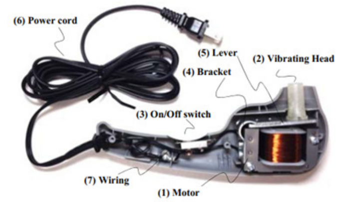
- Design Rule Idea: have components depend on design parameters guaranteed to change
- Architectural design
  - Subsystem identification: services and constraints are specified
  - Module design: modular decomposition is performed; relationships specified
- Studying the designs and correlating their 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



Hand-held massager*

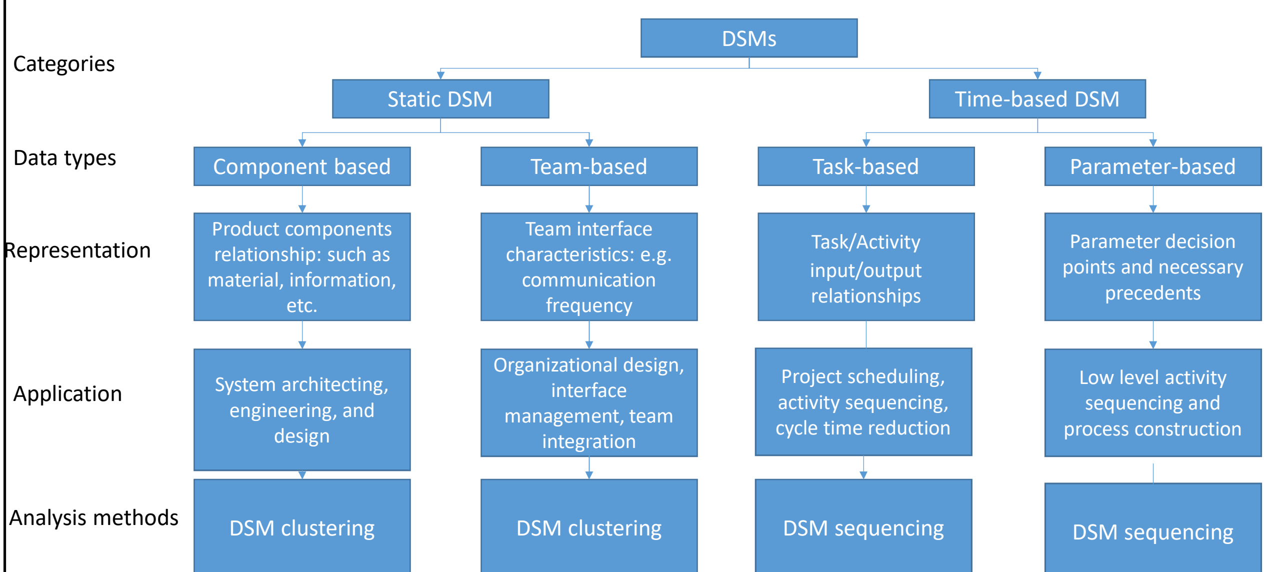
		Motor 1	Vibrating Head 2	On/Off Switch 3	Bracket 4	Lever 5	Power Cord 6	Wiring 7
Motor	1	1	0	0	1	0	0	1
Vibrating Head	2	0	1	0	0	1	0	0
On/Off Switch	3	0	0	1	0	0	1	1
Bracket	4	1	0	0	1	1	0	0
Lever	5	0	1	0	1	1	0	0
Power Cord	6	0	0	1	0	0	1	0
Wiring	7	1	0	1	0	0	0	1

Massager DSM*

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

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

DSM categories and analysis methods*



*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.

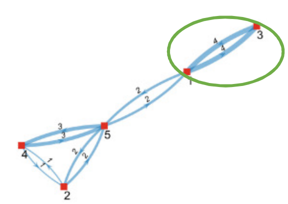
* Qiao, L., Efatmaneshnik, M., Ryan, M., Shoval, S.: Product modular analysis with design structure matrix using a hybrid approach based on MDS and clustering. J. Eng. Design **28**(6), 433–456 (2017)

DSM Application Examples

Product-DSM clustering

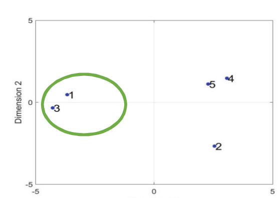
Algorithm and Procedure

Algorithm	SMCOF hierarchical clustering with DSM
Input:	DSM (an $n \times n$ matrix)
Output:	Dendrogram, Partitions for a range of k with Cost, optimal partitions
Step 1: DSM and its digitalization	**If** the entries are not numbers, **then** digitize the DSM, **end if**.
Step 2: SMACOF	Use SMACOF, obtain embedded data (an $n \times m$ matrix)
Step 3: Hierarchical clustering	Compute the distance matrix using *Cosine*, hierarchical clustering the embedded data, obtain the partitions for a range of k
Step 4: Evaluation and knowledge deployment	Calculate the *Cost* for a range of k **If** the P_{ref} exits = True Calculate the *Jaccard* index between the P_{ref} and the obtained partitions **end** **If** there are no design constraints **then** choose the optimal partition when $k = k^*$ **elseif** there are design constraints, modify the partition **end if**



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

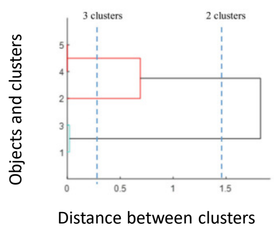
	①	②	③	④	⑤
①			4		2
②				1	2
③	4				
④		1			3
⑤	2	2		3	



Representation in new 2D by SMACOF

Algorithm and Procedure(cont.)

Algorithm	SMCOF hierarchical clustering with DSM
Input:	DSM (an $n \times n$ matrix)
Output:	Dendrogram, Partitions for a range of k with Cost, optimal partitions
Step 1: DSM and its digitalization	**If** the entries are not numbers, **then** digitize the DSM, **end if**.
Step 2: SMACOF	Use SMACOF, obtain embedded data (an $n \times m$ matrix)
Step 3: Hierarchical clustering	Compute the distance matrix using *Cosine*, hierarchical clustering the embedded data, obtain the partitions for a range of k
Step 4: Evaluation and knowledge deployment	Calculate the *Cost* for a range of k **If** the P_{ref} exits = True Calculate the *Jaccard* index between the P_{ref} and the obtained partitions **end** **If** there are no design constraints **then** choose the optimal partition when $k = k^*$ **elseif** there are design constraints, modify the partition **end if**



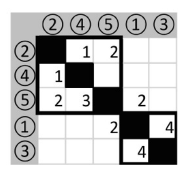
The solutions of hierarchical clustering are called dendrogram.
Represent nested clusters for DSM5.
A quantitative description of data properties.
A partition can be obtained by cutting the dendrogram at a certain level.

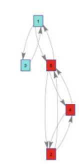
$P_{k=2} = \{\{1,3\},\{2,4,5\}\}$

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

Algorithm and Procedure(cont.)

Algorithm	SMCOF hierarchical clustering with DSM
Input:	DSM (an $n \times n$ matrix)
Output:	Dendrogram, Partitions for a range of k with Cost, optimal partitions
Step 1: DSM and its digitalization	**If** the entries are not numbers, **then** digitize the DSM, **end if**.
Step 2: SMACOF	Use SMACOF, obtain embedded data (an $n \times m$ matrix)
Step 3: Hierarchical clustering	Compute the distance matrix using *Cosine*, hierarchical clustering the embedded data, obtain the partitions for a range of k
Step 4: Evaluation and knowledge deployment	Calculate the *Cost* for a range of k **If** the P_{ref} exits = True Calculate the *Jaccard* index between the P_{ref} and the obtained partitions **end** **If** there are no design constraints **then** choose the optimal partition when $k = k^*$ **elseif** there are design constraints, modify the partition **end if**





Compare the quality of the partition solutions using

$$Cost = \sum IntraClusterCost + \sum ExtraClusterCost$$
$$= \underbrace{\sum [DSM(i,j) + DSM(j,i)] \times d_k}_{i,j \text{ are in the same cluster}} + \underbrace{\sum [DSM(i,j) + DSM(j,i)] \times n}_{i,j \text{ are not in the same cluster}}$$

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.

$P_{k=2} = \{\{1,3\},\{2,4,5\}\}$ Cost = 72

$P_{k=3} = \{\{1,3\},\{4,5\},\{2\}\}$ Cost = 78

The optimal k^*=2. If there are no design constrains, We suggest the solution preference is $P_{k^*=2}$

Algorithm and Procedure(cont.)

Algorithm	SMCOF hierarchical clustering with DSM
Input:	DSM (an $n \times n$ matrix)
Output:	Dendrogram, Partitions for a range of k with Cost, optimal partitions
Step 1: DSM and its digitalization	**If** the entries are not numbers, **then** digitize the DSM, **end if**.
Step 2: SMACOF	Use SMACOF, obtain embedded data (an $n \times m$ matrix)
Step 3: Hierarchical clustering	Compute the distance matrix using *Cosine*, hierarchical clustering the embedded data, obtain the partitions for a range of k
Step 4: Evaluation and knowledge deployment	Calculate the *Cost* for a range of k **If** the P_{ref} exits = True Calculate the *Jaccard* index between the P_{ref} and the obtained partitions **end** **If** there are no design constraints **then** choose the optimal partition when $k = k^*$ **elseif** there are design constraints, modify the partition **end if**

Use Jaccard index to compare partitions $Jaccard\ (X, Y) = |X \cap Y| / |X \cup Y|$

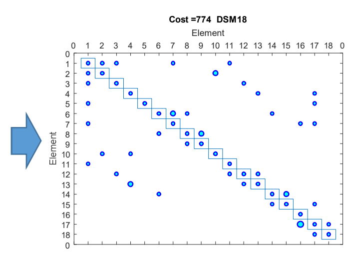
The number of members shared between both sets/the total number of members in both sets

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

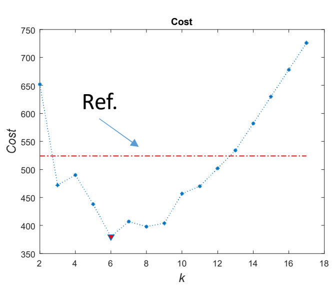
Product modularization(hydraulic support system)

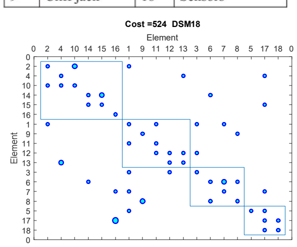
Hydraulic support components

No.	Components	No.	Components
1	Base	10	Pushing jack
2	Pushing bar	11	Front bar
3	Rear bar	12	Cover beam
4	Cover jack	13	Cover board
5	Balance jack	14	Side guard
6	Head beam	15	Side jack
7	Columns	16	Pipes
8	Cliff board	17	Valves
9	Cliff jack	18	Sensors

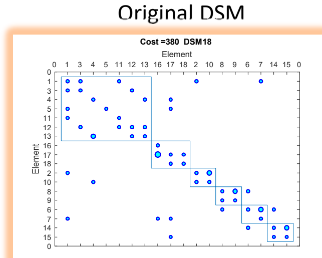


Original DSM





Referenced partitions

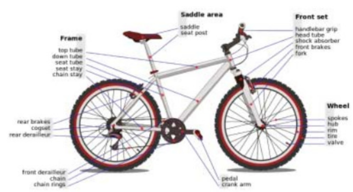


Optimal partitions with lowest cost

Li, Zhongkai, Zhihong Cheng, Yixiong Feng, and Jinyong Yang. 2013. "An Integrated Method for Flexible Platform Modular Architecture Design." Journal of Engineering Design 24 (1): 25–44.

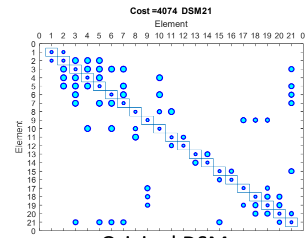
L. Qiao, M. Efatmaneshnik, M. Ryan, and S. Shoval. Product modular analysis with design structure matrix using a hybrid approach based on mds and clustering. *Journal of Engineering Design*, 28(6):433–456, 2017.

Product modularization(Bicycle)

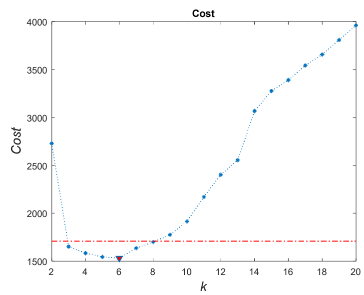


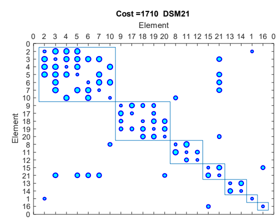
Bicycle components

No.	Name		
1	Saddle	11	Handlebar
2	Seat post	12	Handlebar grip
3	Seat tube	13	Front wheel
4	Top tube	14	Rear wheel
5	Seat stay	15	Crank arm
6	Down tube	16	Pedal
7	Chain stay	17	Rear brake
8	Fork	18	Rear derailleur
9	Front brake	19	Front derailleur
10	Head tube	20	Chain
		21	Chain Ring

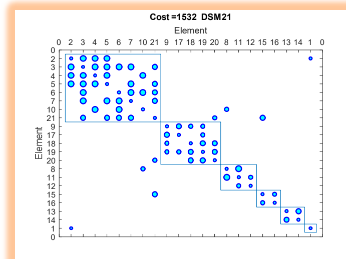


Original DSM





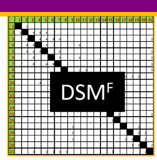
Referenced partitions



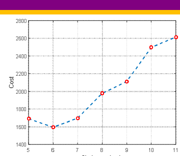
Optimal partitions with lowest cost

Shoval, Shraga, Li Qiao, Mahmoud Efatmaneshnik, and Michael Ryan. 2016. "Dynamic Modular Architecture for Product Lifecycle." *Procedia CIRP* 48: 271–276.

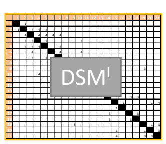
Dynamic Modular Architecture For Product Lifecycle: DSM^3



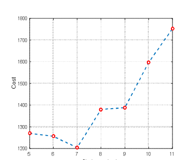
Clustering functions



Cluster 1 (8)	Seat post[2] Seat tube[3] Top tube[4] Seat stay[5] Down tube[6] Chain stay[7] Head tube[10] Chain ring[21]
Cluster 2 (5)	Front brake[9] Rear brake[17] Rear derailleur[18] Front derailleur[19] Chain[20]
Cluster 3 (3)	Fork[8] Handlebar[11] Handlebar grip[12]
Cluster 4 (2)	Crank arm[15] Pedal[16]
Cluster 5 (2)	Front wheel[13] Rear wheel[14]
Cluster 6 (1)	Saddle[1]



Clustering ilities



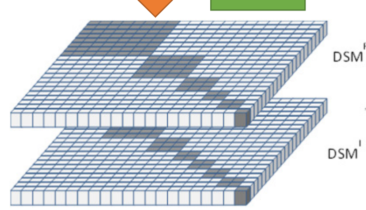
Cluster 1 (6)	Seat post[2] Seat tube[3] Top tube[4] Seat stay[5] Down tube[6] Chain stay[7]
Cluster 2 (5)	Crank arm[15] Pedal[16] Front derailleur[19] Chain[20] Chain ring[21]
Cluster 3 (3)	Rear wheel[14] Rear brake[17] Rear derailleur[18]
Cluster 4 (2)	Fork[8] Head tube[10]
Cluster 5 (2)	Front brake[9] Front wheel[13]
Cluster 6 (2)	Handlebar[11] Handlebar grip[12]
Cluster 7 (1)	Saddle[1]

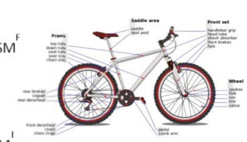
DSM^3

	M_1^F	M_2^F	M_3^F	M_4^F	M_5^F	M_6^F
M_1^I	$^6/_8$	$^1/_{12}$	0	0	0	0
M_2^I	$^1/_{12}$	$^2/_8$	0	$^2/_5$	0	0
M_3^I	0	$^2/_6$	0	0	$^1/_4$	0
M_4^I	$^1/_{98}$	0	$^1/_4$	0	0	0
M_5^I	0	$^1/_6$	0	0	$^1/_3$	0
M_6^I	0	0	$^2/_3$	0	0	0
M_7^I	0	0	0	0	0	1

Similarity matrix

Jaccard similarity index





Shoval, Shraga, Li Qiao, Mahmoud Efatmaneshnik, and Michael Ryan. 2016. "Dynamic Modular Architecture for Product Lifecycle." *Procedia CIRP* 48: 271–2[illegible]

Dynamic Modular Architecture For Product Lifecycle: DSM^3(cont.)

- ***Type I*** – clusters that are identical in both the DSM^I and the DSM^F.
- ***Type II*** – large clusters that consist of a mixture of elements from entire smaller clusters from the other layer of the DSM^3.
- ***Type III*** – clusters that exist in one DSM layer and not in the other.

Internal hub gearing mechanism and conventional derailleur gearing mechanism

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

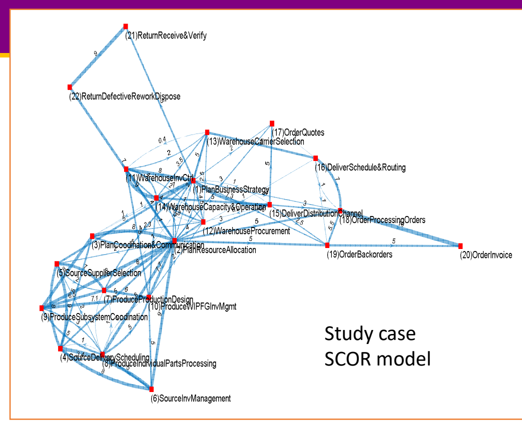
Supply Chain Outsourcing Problem

- 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 today
Source: A. Subramaniam Supply Chain Risk Management

Supply Chain model

Plan	Source	Produce	Ware-house	Deliver	Order	Return
Business Strategy (1)	Delivery Scheduling (4)	Production Design (7)	Inventory Control (11)	Distribution Channel (15)	Quotes (17)	Receive & Verify (21)
Resource Allocation (2)	Supply Selection (5)	Individual Parts Processing (8)	Procurement (12)	Schedule & Routing (16)	Processing Orders (18)	Defective Rework/ Dispose (22)
Coordination & Communica-tion (3)	Inventory Management (6)	Subsystem Coordination (9)	Carrier Selection (13)		Back Orders (19)	
		WIP/FG Inventory Management (10)	Capacity & Operation (14)		Invoice (20)	

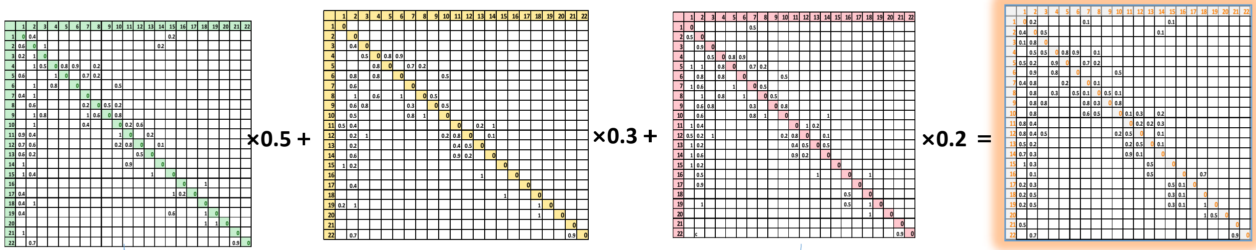


Study case
SCOR model

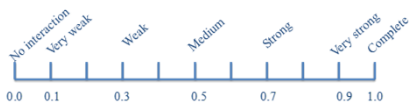
A supply chain is a complex network. There are different types of relationships (or called interactions) among these components, such as information, materiel, function and financial.
What we do: developing a systematic approach to identify and quantify the interactions and cluster the large network to reasonable number of subgroups of core competencies and ancillary groups.

SCOR model structure has
7 stages/subsystems
22 functional component

Various flows modelling of a supply chain DSM



Information Material Functional Aggregated DSM



Scales for determine the interaction strength

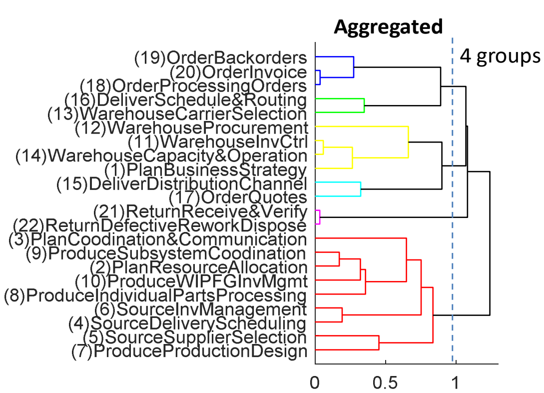
Original matrix: from Chen and Huang, 2007

Core_competency_index

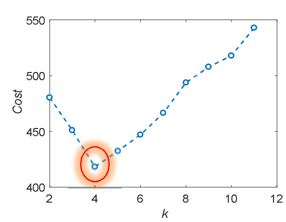
[0.5 0.3 0.2] are weights

$$DSM_{Aggregated} = DSM_{Flow\_1} \times cc\_index_{Flow\_1} + DSM_{Flow\_2} \times cc\_index_{Flow\_2} + \cdots = \sum_i DSM_{Flow\_i} \times cc\_index_{Flow\_i}$$

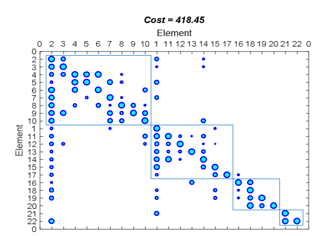
Modular analysis of supply chain network



Dendrogram of aggregated DSM



Cost vs k

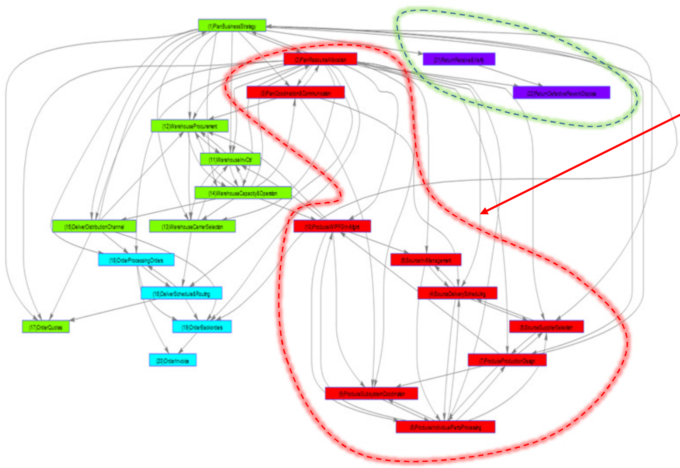


The segmented aggregated DSM with the lowest Cost value.

A diagonalization pattern

$$P_{k^*=4} = \{\underbrace{\{2,3,4,5,6,7,8,9,10\}}_{Group\ 1}, \underbrace{\{1,11,12,13,14,15,17\}}_{Group\ 2}, \underbrace{\{16,18,19,20\}}_{Group\ 3}, \underbrace{\{21,22\}}_{Group\ 4}\}$$

Support supply chain outsourcing decision making

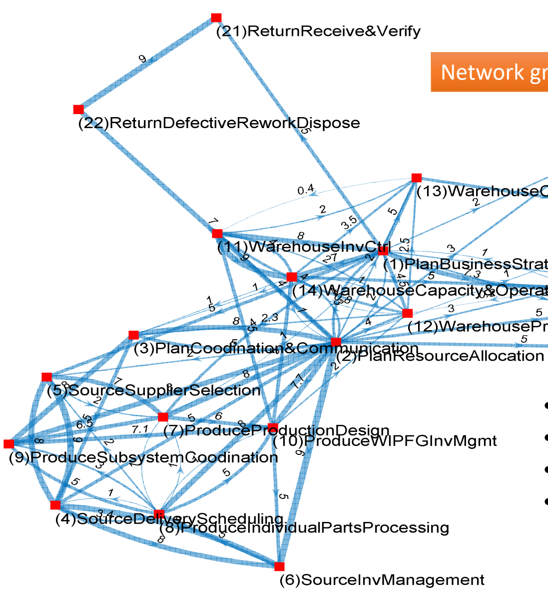


$$P_{k^*=4} = \{\underbrace{\{2,3,4,5,6,7,8,9,10\}}_{Group\ 1}, \underbrace{\{1,11,12,13,14,15,17\}}_{Group\ 2}, \underbrace{\{16,18,19,20\}}_{Group\ 3}, \underbrace{\{21,22\}}_{Group\ 4}\}$$

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



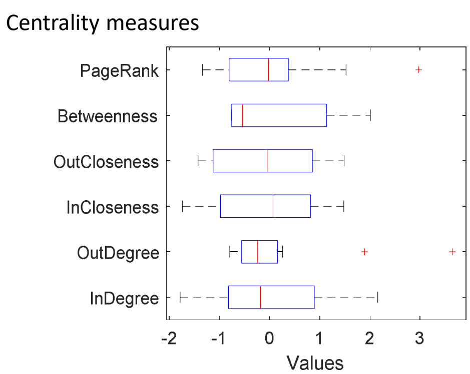
Network graph for the supply chain functional structure with 22 nodes

- 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

L. Qiao and M. Ryan. A hybrid approach for supply chain analysis: An application of network and cluster analysis. In *INCOSE International Symposium*, volume 27, pages 746–762. Wiley Online Library, 2017.

Algorithm: Centrality measures in graph theory and clustering algorithm

DSM + social network analysis (graph theory)

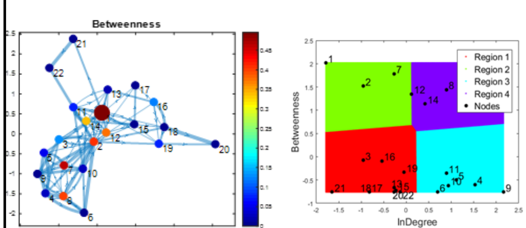


Different centralities distribution

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

Components	Degree		Closeness		Betweenness	PageRank
	In-degree	Out-degree	In-closeness	Out-closeness		
1	4.0	57.4	0.0073	0.0087	104.0	0.0255
2	10.0	94.7	0.0056	0.0115	85.0	0.0221
3	10.0	20.3	0.0078	0.0068	26.0	0.0190
4	28.0	20.4	0.0056	0.0055	6.0	0.0832
5	25.0	10.0	0.0065	0.0069	10.5	0.0677
6	22.0	14.0	0.0029	0.0047	0	0.0434
7	15.0	21.5	0.0060	0.0094	96.0	0.0623
8	23.4	12.0	0.0057	0.0086	83.0	0.1189
9	32.6	5.0	0.0040	0.0046	0	0.0547
10	23.7	20.1	0.0040	0.0101	5.5	0.0482
11	23.4	15.0	0.0055	0.0053	15.0	0.0644
12	17.8	5.50	0.0068	0.0090	79.0	0.0527
13	15.0	5.4	0.0054	0.0055	3.5	0.0522
14	20.0	20.3	0.0057	0.0106	71.0	0.0466
15	15.3	11.7	0.0037	0.0106	1.0	0.0190
16	13.0	3.0	0.0059	0.0040	25.0	0.0420
17	11.0	0.0	0.0076	0	0	0.0285
18	11.0	22.5	0.0075	0.0012	0	0.0285
19	16.6	5.0	0.0075	0.0005	16.0	0.0366
20	15.0	0.0	0.0042	0	0	0.0499
21	5.0	9.0	0.0037	0.0003	0	0.0124
22	16.0	0.0	0.0030	0	0	0.0222

E.g. Betweenness

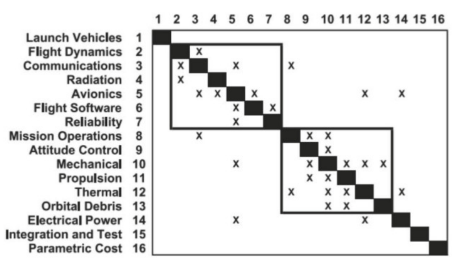


- High betweenness nodes act as critical intermediates.
- Have the potential to influence others near them, through both direct and indirect pathways.
- The nodes with high betweenness and low degree mean their few ties are crucial for network flow.

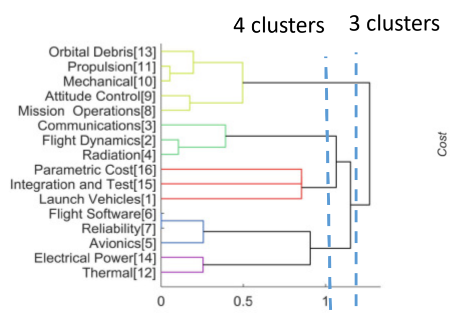
Network characteristics	Network interpretation	Resource recommendations	Example nodes
High in-degree	Prestigious/dependent, critical and vulnerable	Upstream information alignment and sharing service	9, 4, 5,10,8,11
High out-degree	Conductive, critical and vulnerable	Downstream information alignment and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

Team-based DSM clustering

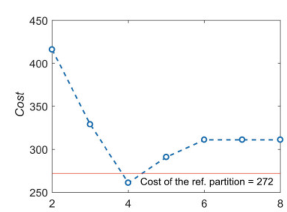
Team-based satellite design problem

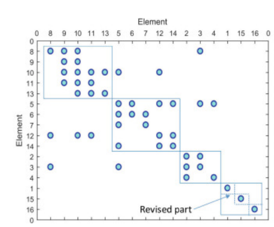


Binary Team-based satellite design problem.
16 subsystems/disciplines, marked as DSM16
X presents the interaction among members
Transfer it to a binary matrix.
New-Girvan algorithm is adopted to obtain



Dendrogram identify the related elements, such as Thermal[12] and Electrical power[14] .



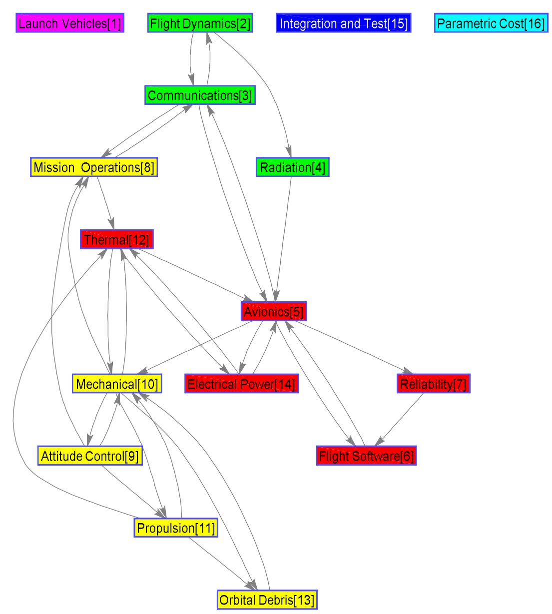


P_{ref} $\{\{1\},\{2,3,4,5,6,7\},\{8,9,10,11,12,13\},\{14\},\{15\},\{16\}\}$

Optimal partition with lower cost value of 264

Avnet, M.S., Weigel, A.L.: An application of the design structure matrix to integrated concurrent engineering. Acta Astronautica **66**(5-6), 937–949 (2010)

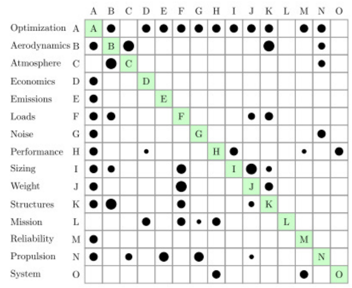
Team-based satellite design problem(cont.)

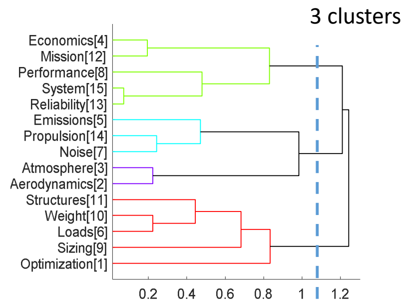


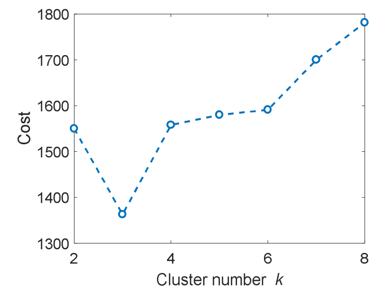
- The optimal number of cluster $k^* = 4$.
- Launch Vehicle (1), Integration and Test (15) and Parametric Cost (16) are included though there are not any dependencies to other disciplines. Set the design constraints here that (1), (15) and (16) must be located in independent groups. Then we revise the 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





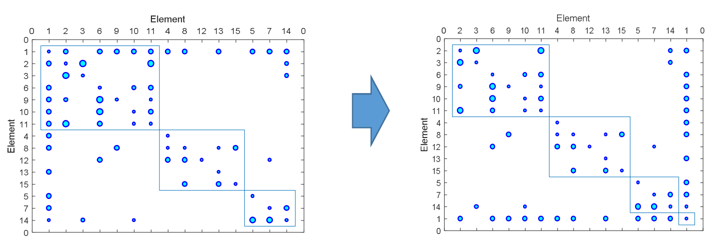


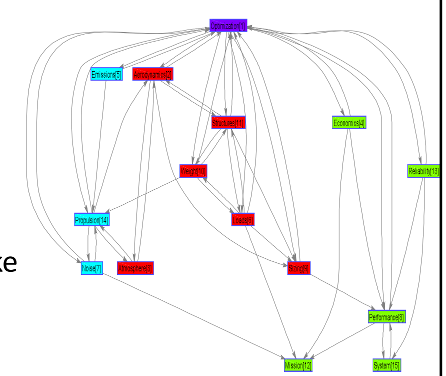
Plot of cost vs.k suggests the optimal k^{*}=3.

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.

Lambe, A.B., Martins, J.R.R.A.: Extensions to the design structure matrix for the description of multidisciplinary design, analysis, and optimization processes. Struct. Multidisciplinary Optim. **46**(2), 273–284 (2012)

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





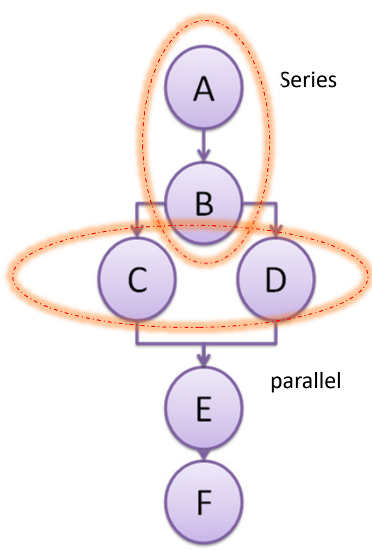
The activities with the same colour should work more closely together.

- $P_{k=3}$ 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.

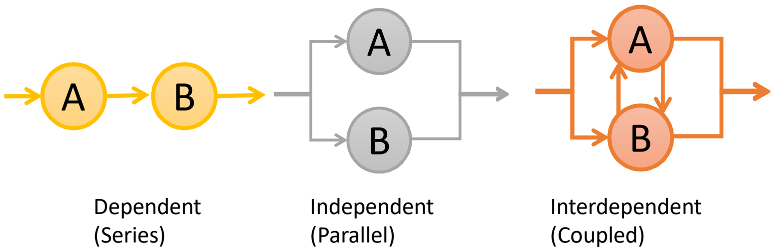
Parameter-based DSM sequencing

DSM sequencing

- Project management is hard…We needs 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



	A	B
A		
B	1	

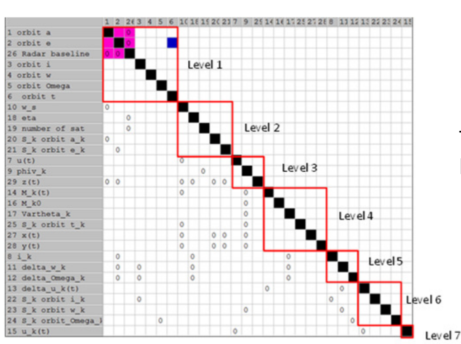
	A	B
A		
B		

	A	B
A		1
B	1	

- 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 DSM29
- Ssequencing 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.

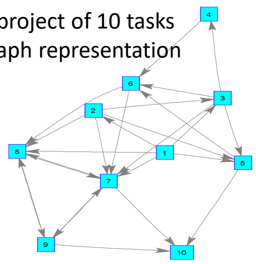
The coupled cycle highlight in pink

Parameter(s)	Meaning	Parameter(s)	Meaning
1 orbit a	Reference satellite orbit semimajor axis	16 M_k0	Add-on satellite initial mean anomaly
2 orbit e	Reference satellite orbit eccentricity	17 Vartheta_k(ϑ_k)	Cartwheel design parameter
3 orbit i	Reference satellite orbit inclination	18 eta (η)	Cartwheel design parameter
4 orbit w (ω)	Reference satellite orbit argument of perigee	19 number of sat	Number of add-on satellites
5 orbit Omega (Ω)	Reference satellite orbit right ascension of the ascending node	20 S_k orbit a_k (a_k)	Add-on satellite orbit semimajor axis
6 orbit t(τ)	Time of reference satellite perigee passing point	21 S_k orbit e_k (a_k)	Add-on satellite orbit eccentricity
7 u(t)	Reference satellite orbit Intersection angle	22 S_k orbit i_k (i_k)	Add-on satellite orbit inclination
8 delta i_k (Δi_k)	Inclination difference in formation	23 S_k orbit w_k (ω_k)	Add-on satellite argument of perigee
9 phiv_k (φ_k)	Cartwheel design parameter: Add-on satellite initial phase	24 S_k orbit_Omega_k(Ω_k)	Add-on satellite
10 w_s (ws)	Add-on satellite mean angular rate	25 S_k orbit t_k(τk)	Time of add-on satellite perigee passing point
11 delta_w_k (Δwk)	Change in argument of perigee	26 Radar baseline	High-level activity presents all the radar requirements
12 delta_Omega_k($\Delta\Omega$k)	Modal separation	27 x(t)	Radial difference between the two satellites
13 delta_u_k(t)	Add-on satellite change in intersection angle	28 y(t)	Along-track difference between the two satellites
14 M_k(t)	Add-on satellite mean anomaly	29 z(t)	Cross-track difference between the two satellites
15 u_k(t)	Add-on satellite intersection angle		

29 parameters in satellite formation flying problem

Sequencing algorithm derived from reachability (an ideal case)

A project of 10 tasks graph representation

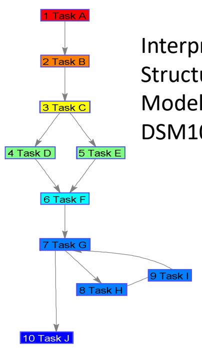


		1	2	3	4	5	6	7	8	9	10
1 Task A	1	0	0	0	0	0	0	0	0	0	0
2 Task B	2	1	0	0	0	0	0	0	0	0	0
3 Task C	3	1	1	0	0	0	0	0	0	0	0
4 Task D	4	0	0	1	0	0	0	0	0	0	0
5 Task E	5	1	1	1		0	0	0	0	0	0
6 Task F	6	0	0	1	1	1	0	0	0	0	0
7 Task G	7	1	1	1	0	0	1	0	1	1	0
8 Task H	8	1	1	0	0	0	1	1	0	1	0
9 Task I	9	0	0	0	0	0	0	1	1	0	0
10 Task J	10	0	0	0	0	1	0	1	0	1	0

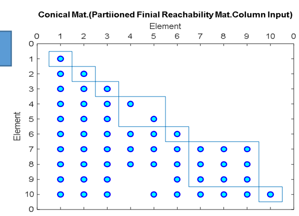
A DSM representation of a project with 10 tasks (column as input)

1	1	1	0	1	0	1	1	0	0	6	Driver power
0	1	1	0	1	0	1	1	0	0	5	
0	0	1	1	1	1	1	0	0	0	5	
0	0	0	1	0	1	0	0	0	0	2	
0	0	0	0	1	1	0	0	0	1	3	
0	0	0	0	0	1	1	1	0	0	3	
0	0	0	0	0	0	1	1	1	1	4	
0	0	0	0	0	0	1	1	1	0	3	
0	0	0	0	0	0	1	1	1	1	4	
0	0	0	0	0	0	0	0	0	1	1	
1	2	3	2	4	4	7	6	3	4	36 / 36	
Dependence											

Finial Reachability matrix



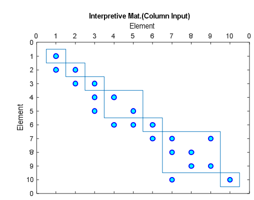
Interpretive Structural Model of DSM10



. Conical matrix

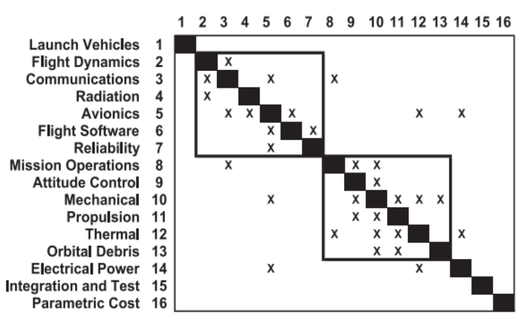
Level 1	10	0	0	0
Level 2	7	8	9	0
Level 3	6	0	0	0
Level 4	4	5	0	0
Level 5	3	0	0	0
Level 6	2	0	0	0
Level 7	1	0	0	0

Level matrix



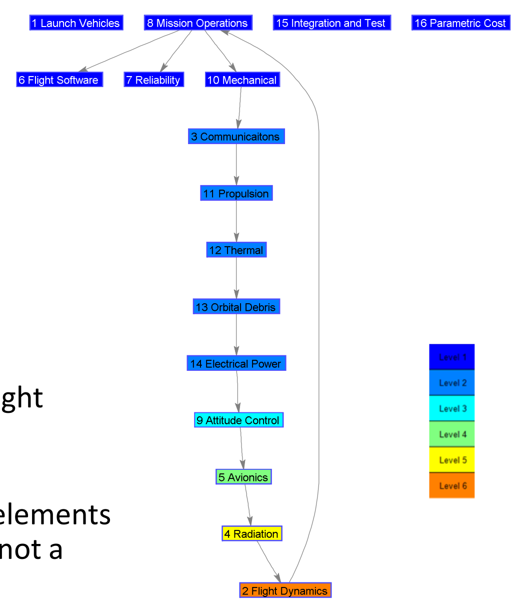
Interpretive matrix

Satellite design tasks problem

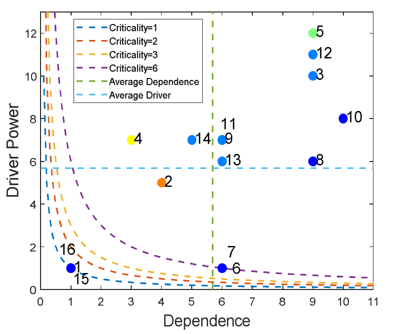


The temperature(warm/cool) of the colour shows the precedence relationship between the elements. "2 Flight dynamics" has the warmest colour, which suggests its importance as the start work.

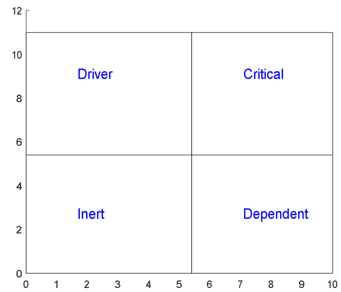
The rest 11 elements forms a big feedback loop. The elements in this loop has different colours, which means this is not a simple loop.



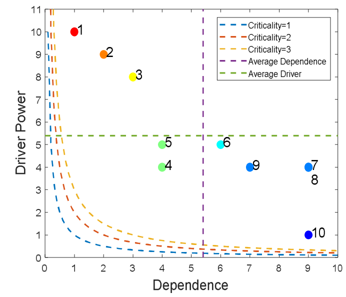
Satellite design tasks problem(cont.)



Influence diagram
(Drive vs. Dependence plot)



Classification of elements into four sectors



Influence diagram of ideal case

- The driver sector elements influences 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…*