Multidisciplinary System Design Optimization (MSDO)

Decomposition and Coupling

Lecture 4

Anas Alfaris

Today's Topics

- Information Flow and Coupling

- MDO frameworks
  - Single-Level (Distributed analysis)
  - Multi-Level (Distributed design)
    - Collaborative Optimization
    - Analytical Target Cascading
    - (Hierarchical Decomposition & Multi-Domain Formulation)

Standard Optimization Problem

Given

$$x \in \Re^n$$

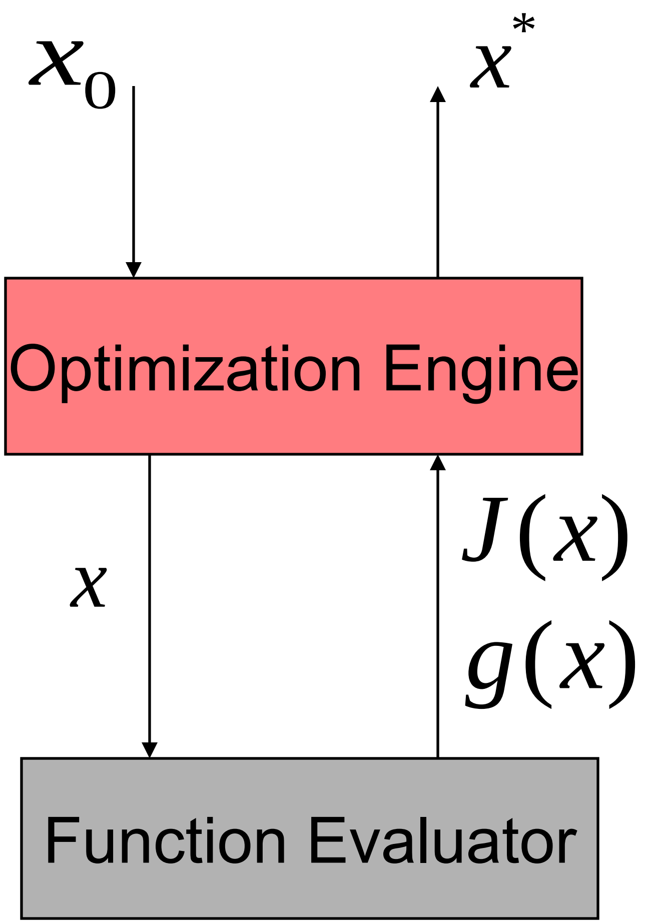
$$J : \Re^n \rightarrow \Re^z$$

$$g : \Re^n \rightarrow \Re^m$$

Solve the problem

$$\min J(x)$$

$$\text{s.t.} \quad g(x) \geq 0$$



That is, find x^* s.t. $J(x^*) \leq f(x), \ \forall x \in \text{dom}(J) \cap \text{dom}(g)$

Information Flow

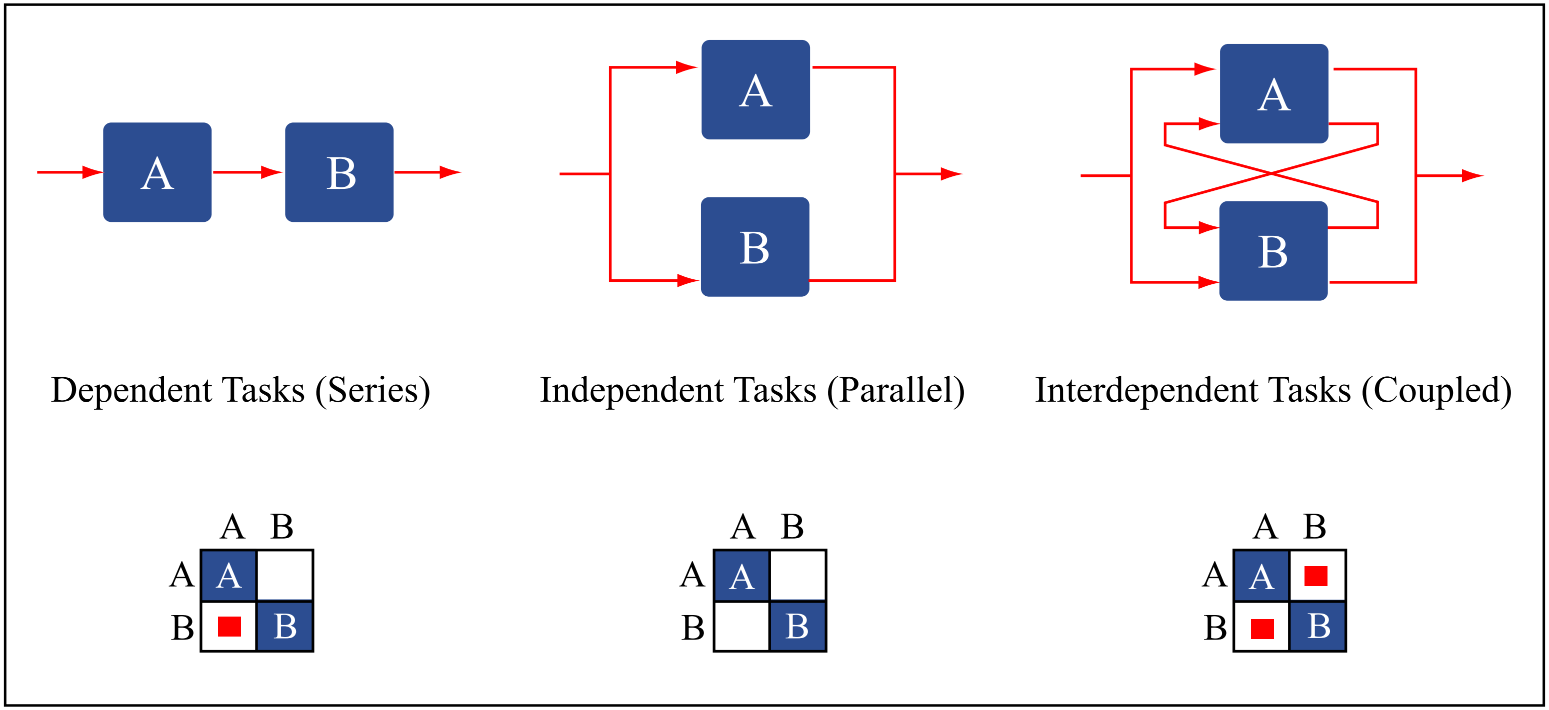


Image by MIT OpenCourseWare.

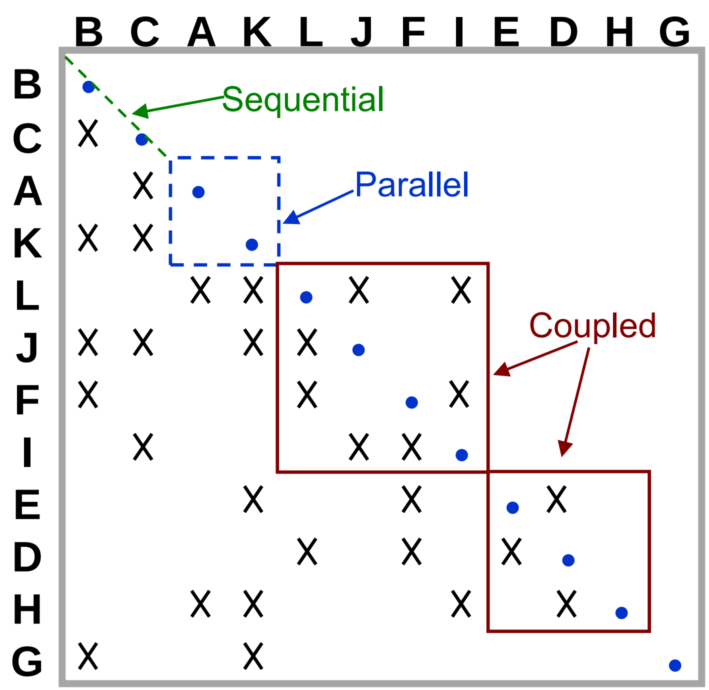
Information Flow

Feed Forward

	B	C	A	K	L	J	F	I	E	D	H	G
B	•											
C	X	•										
A		X	•									
K	X	X		•								
L			X	X	•	X		X				
J	X	X		X	X	•						
F	X				X		•	X				
I		X				X	X	•				
E				X			X		•	X		
D					X		X		X	•		
H			X	X				X		X	•	
G	X			X								•

Feed Backward

Information Flow



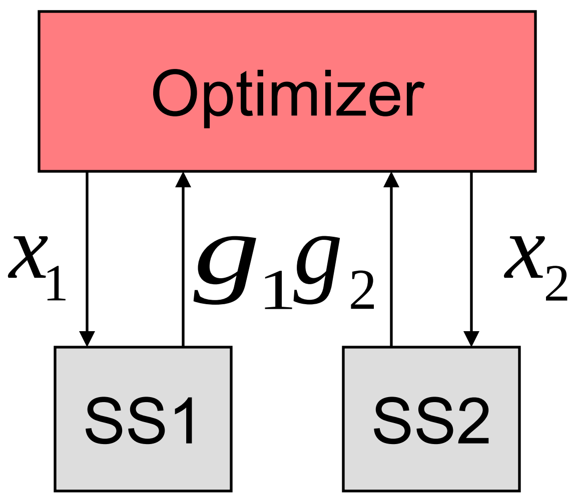
Advantages of Decoupling

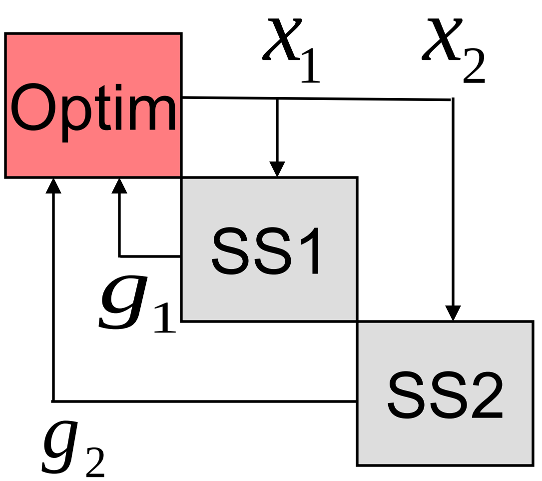
Computation of *g(x)* can be very time consuming, want to divide the work and compute in parallel.

For example, if $x = (x_1, x_2)$, where $x_1 \in \mathbb{R}^{n1}, x_2 \in \mathbb{R}^{n2}$

$$\text{and } g(x) = (g_1(x_1), g_2(x_2))$$

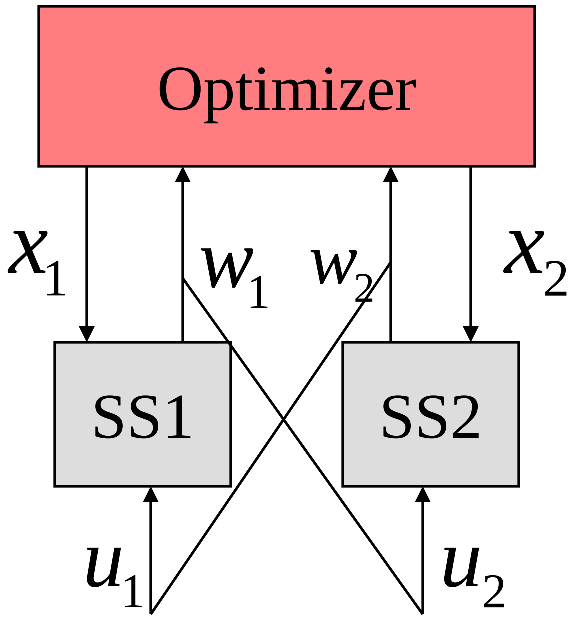
Then g_1 and g_2 can be computed in parallel. Graphically,

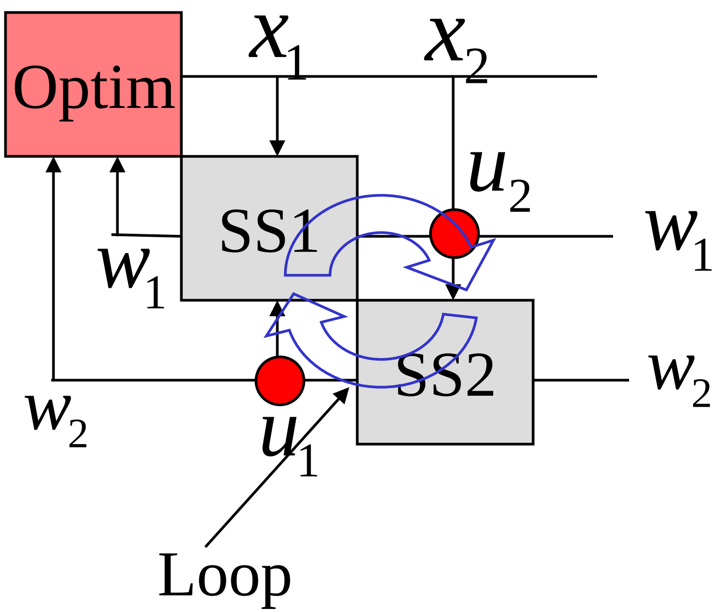




Coupling

The decoupled constraints assumption is not general. Subsystems can be coupled and loops can arise. For example,





x: decision variables
w: SS outputs (constraint, cost)
u: SS input (dependent)

vline: SS input
hline: SS output

Computation of w_1 and w_2 requires an iterative method.

Coupling

- An example where such a loop happens is as follows:

$$\min J(x_1, x_2)$$

$$\text{s.t.} \quad \begin{aligned} w_1 &= g_1(x_1, g_2(x_2, w_1)) \geq 0 \\ w_2 &= g_2(x_2, g_1(x_1, w_2)) \geq 0 \end{aligned}$$

where $x_1 \in \mathbb{R}^{n1}, x_2 \in \mathbb{R}^{n2}, g_i : x_i \times u_i \mapsto w_i, i = 1, 2$

- w_1 and w_2 satisfy coupled relations at each optimization iteration. At each constraint evaluation, nonlinear equations must be solved (e.g. by Newton's method) in order to obtain w_1 and w_2, which can be time consuming.

Want a way to return to the situation of decoupled constraints.

Surrogate Variables ("Tearing")

Information loop can be broken by introducing surrogate variables.

$$\min J(x_1, x_2)$$

s.t.
$$w_1 = g_1(x_1, g_2(x_2, w_1)) \geq 0$$
$$w_2 = g_2(x_2, g_1(x_1, w_2)) \geq 0$$

⇒

$$\min J(x_1, x_2)$$

s.t.
$$g_1(x_1, u_1) \geq 0$$
$$g_2(x_2, u_2) \geq 0$$
$$u_2 - g_1(x_1, u_1) = 0$$
$$u_1 - g_2(x_2, u_2) = 0$$

- u_1 and u_2 are decision variables acting as the inputs to g1(SS1) and g2 (SS2). Introducing surrogate variables breaks information loop but increases the number of decision variables.

Numerical Example

$$\min J_1 + J_2$$
$$\text{s.t. } w_1 \geq 0$$
$$w_2 \geq 0$$
$$\text{where } J_1 = x_1^2 + x_2^2$$
$$J_2 = (x_3 - 3)^2 + (x_4 - 4)^2$$
$$w_1 = x_1^3 - x_2^3 + 2w_2$$
$$w_2 = x_3^3 - x_4^3 + 2w_1$$

decoupled

$$\min x_1^2 + x_2^2 + (x_3 - 3)^2 + (x_4 - 4)^2$$
$$\text{s.t. } w_1 = x_1^3 - x_2^3 + 2x_5 \geq 0$$
$$w_2 = x_3^3 - x_4^3 + 2x_6 \geq 0$$
$$x_1^3 - x_2^3 + 2x_5 - x_6 = 0$$
$$x_3^3 - x_4^3 + 2x_6 - x_5 = 0$$

coupled

$$\min x_1^2 + x_2^2 + (x_3 - 3)^2 + (x_4 - 4)^2$$
$$\text{s.t. } w_1 = g_1(x_1, x_2, x_3, x_4) \geq 0$$
$$w_2 = g_2(x_1, x_2, x_3, x_4) \geq 0$$

Solution:

$$x = (0, 0, 4, 3, 12\tfrac{1}{3}, 24\tfrac{2}{3})$$

MATLAB® 5.3

coupled: 356,423 FLOPS 4.844s
uncoupled: 281,379 FLOPS 0.453s

Single-level and Multi-Level Frameworks

Single-level
(Distributed Analysis)
-disciplinary models provide analysis
-all optimization done at system level

→ non-hierarchical decomposition

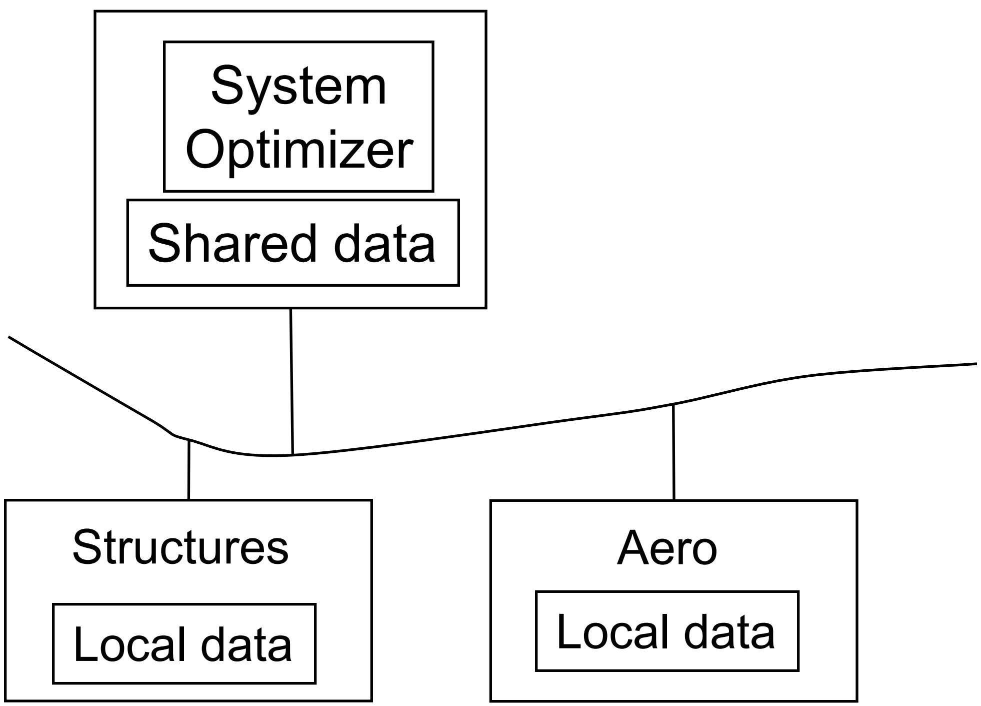
Multi-level
(Distributed Design)
-provide disciplinary models with design tasks
-optimization at subsystem and system levels

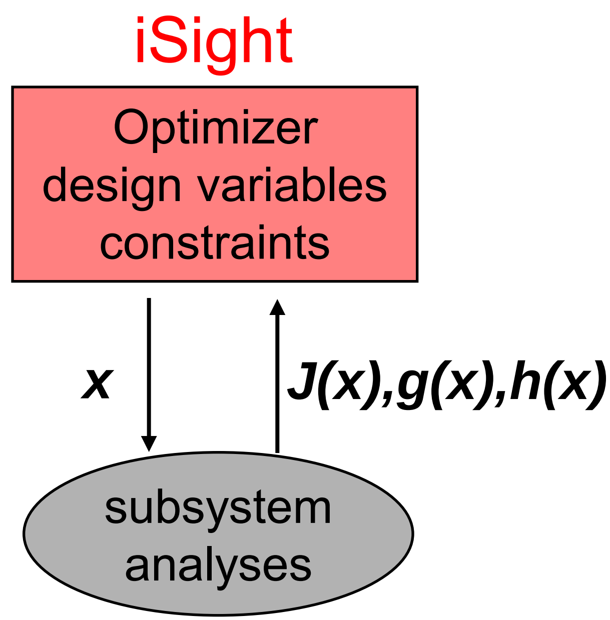
→ hierarchical decomposition

Single-level (Distributed Analysis)

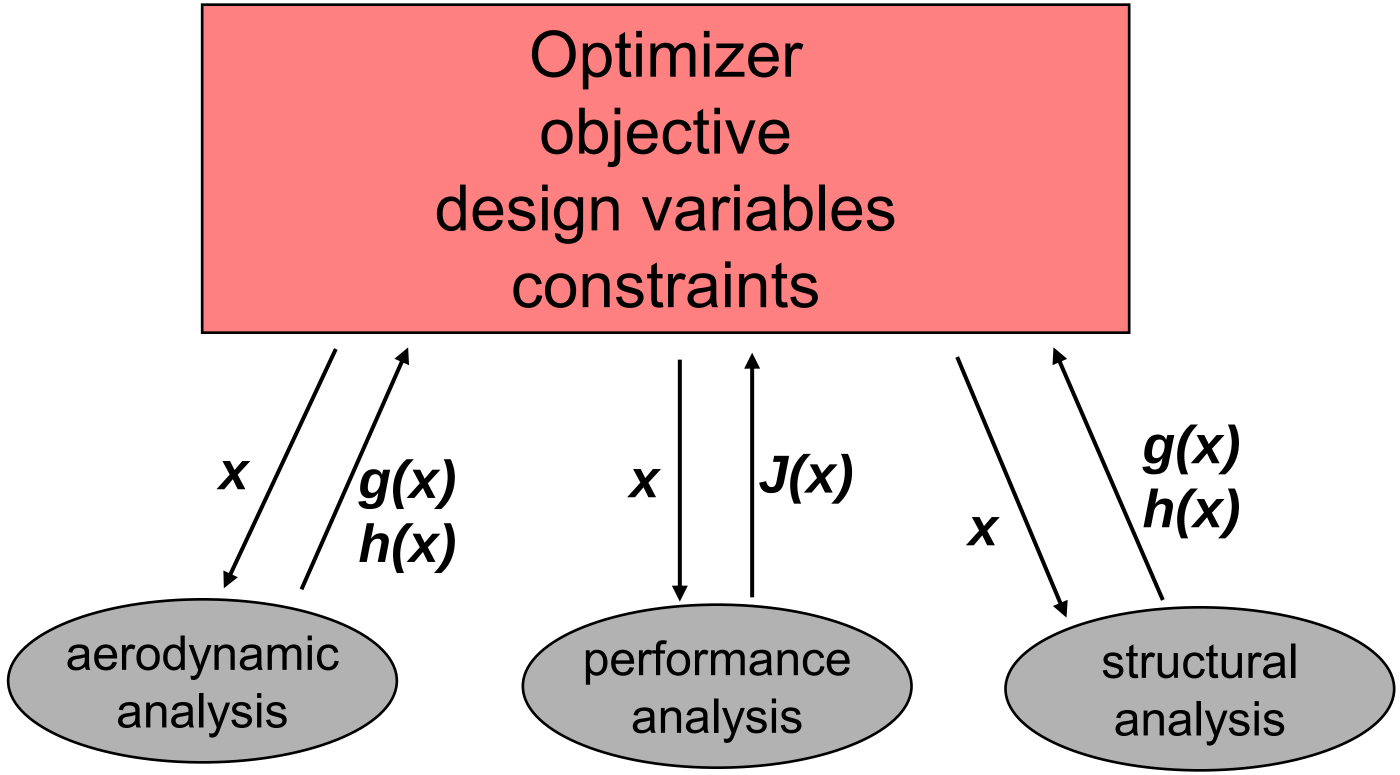
- Disciplinary models provide analysis
- Optimization is controlled by some overseeing code or database

  e.g. ISight (Optimizer)





Single-level (Distributed Analysis)



- During the optimization, the overseeing code keeps track of the values of the design variables and objective
- The values of the design variables are changed according to the optimization algorithm
- Disciplinary models are asked to evaluate constraints/objective

Multi-level (Distributed Design)

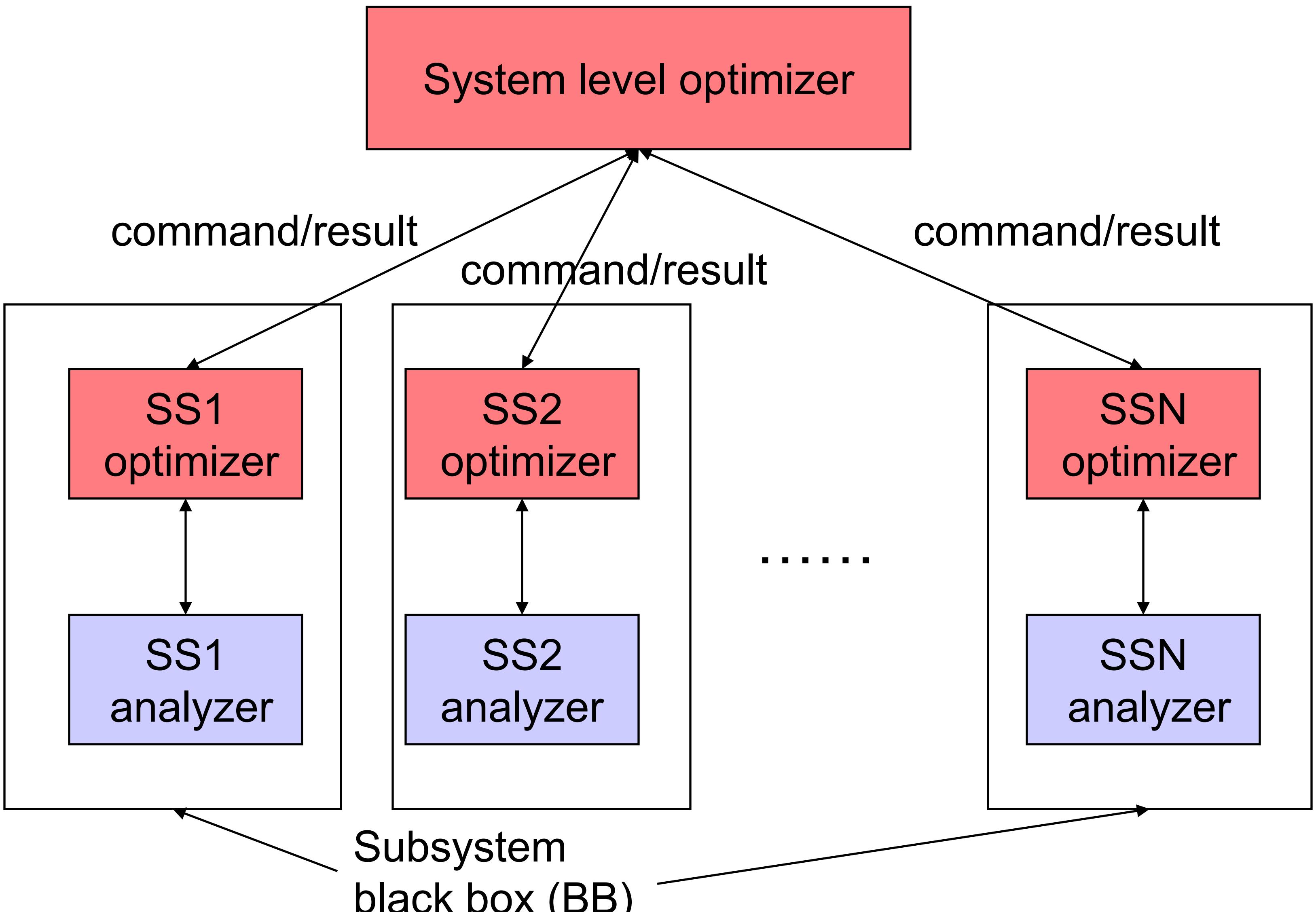
- Multi-level Optimization methods distribute decision making throughout the system

- Subsystem level models are provided with design tasks

- Optimization is performed at a subsystem level in addition to the system level

- Provide some autonomy to design groups and reduces communication requirements.

Multi-level (Distributed Design)

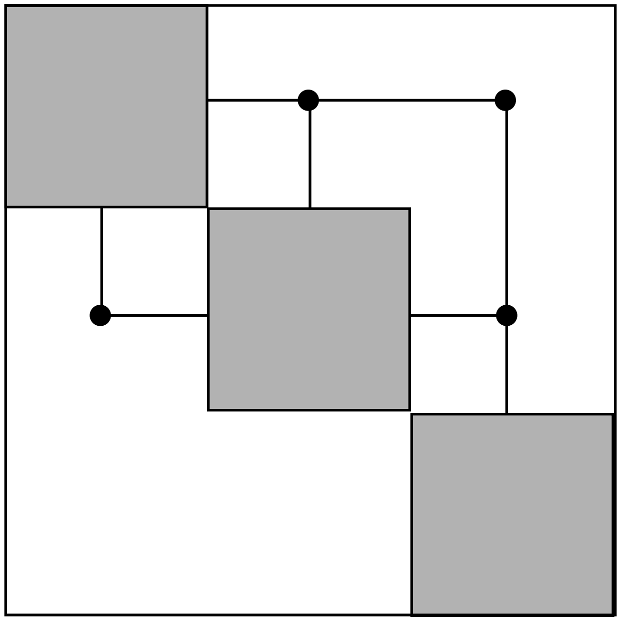


Collaborative Optimization

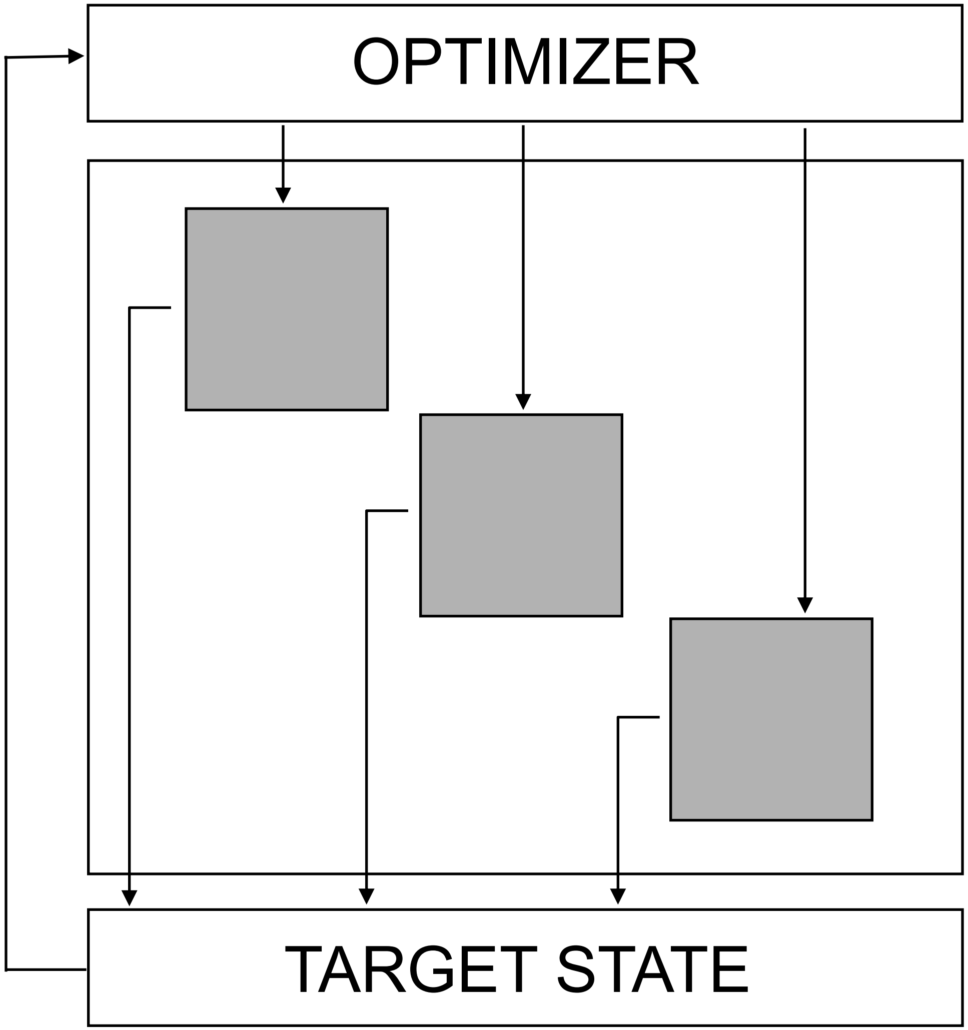
Collaborative Optimization (CO)

- disciplinary teams satisfy local constraints while trying to match target values specified by a system coordinator
- preserves disciplinary-level design freedom.
- CO is used typically to solve discipline-based decomposed system optimization problems.

Collaborative Optimization



Coupled



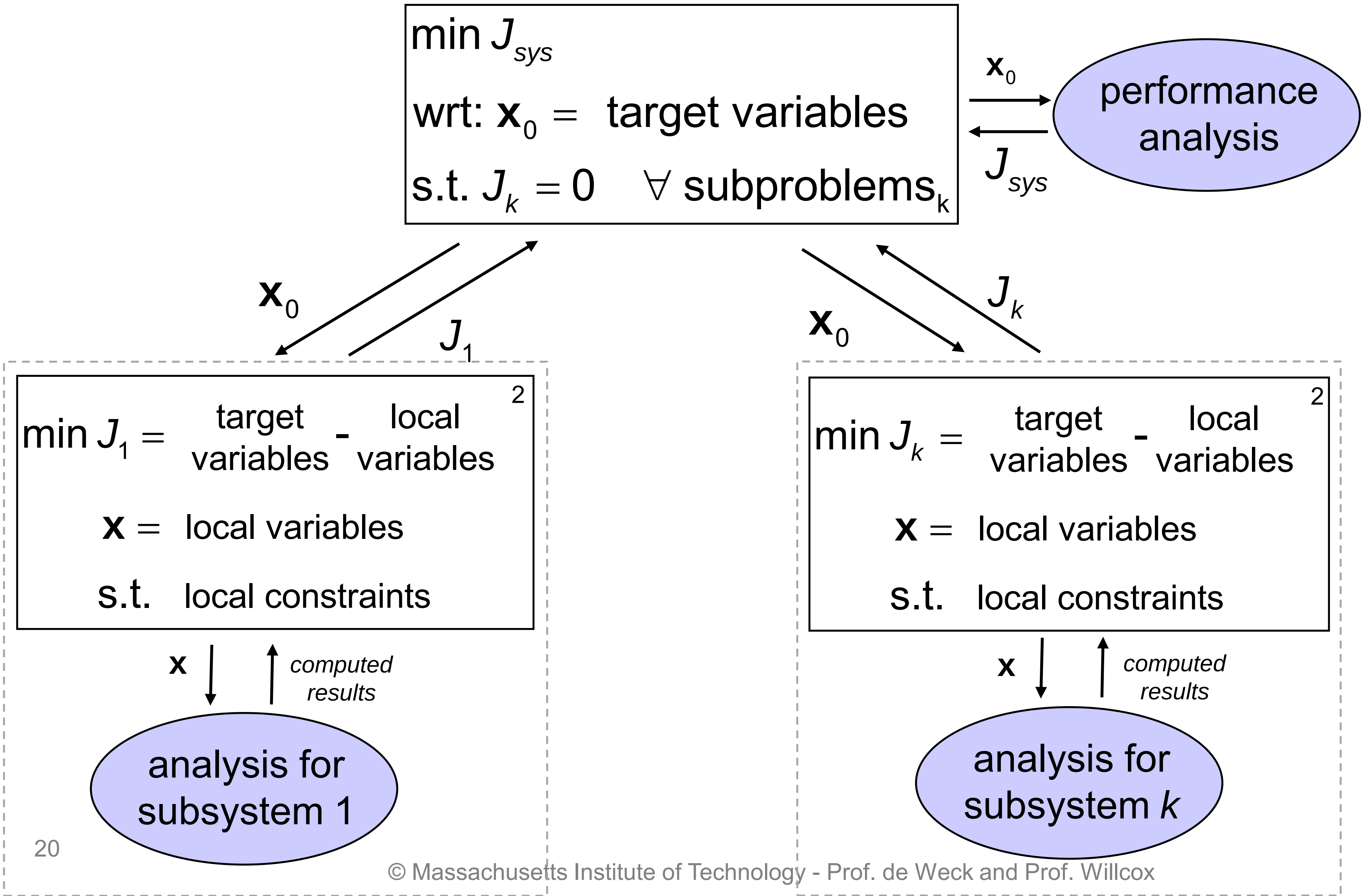
Uncoupled

Collaborative Optimization

Two levels of optimization:

- A system-level optimizer provides a set of targets.
  - These targets are chosen to optimize the system-level objective function
- A subsystem optimizer finds a design that minimizes the difference between current states and the targets.
  - Subject to local constraints

Collaborative Optimization



CO – Subsystem Level

$$\min J_1 = \left\| \begin{matrix}\text{target} \\ \text{variables}\end{matrix} - \begin{matrix}\text{local} \\ \text{variables}\end{matrix} \right\|^2$$

$$\mathbf{x} = \text{local variables}$$

$$\text{s.t.} \quad \text{local constraints}$$

- The subsystem optimizer modifies local variables to achieve the best design for which the set of local variables and computed results most nearly matches the system targets
- The local constraints must also be satisfied

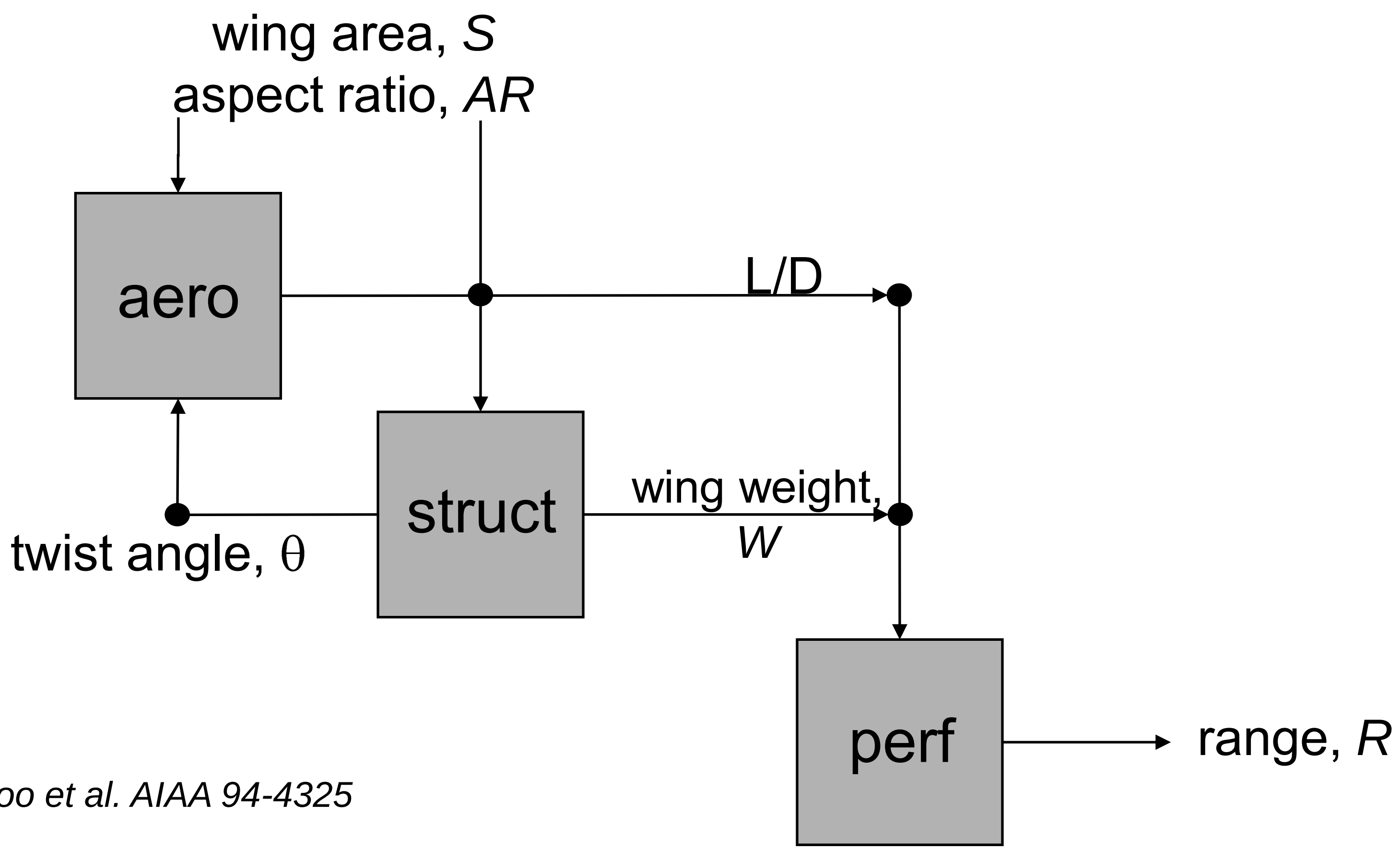
CO – System Level

$$\min J_{sys}$$
$$\text{wrt: } \mathbf{x}_0 = \text{ target variables}$$
$$\text{s.t. } J_k = 0 \quad \forall \text{ subproblems}_k$$

- System-level optimizer changes target variables to improve objective and reduce differences J_k
  - $J_k=0$ are called compatibility constraints
  - compatibility constraints are driven to zero, but may be violated during the optimization

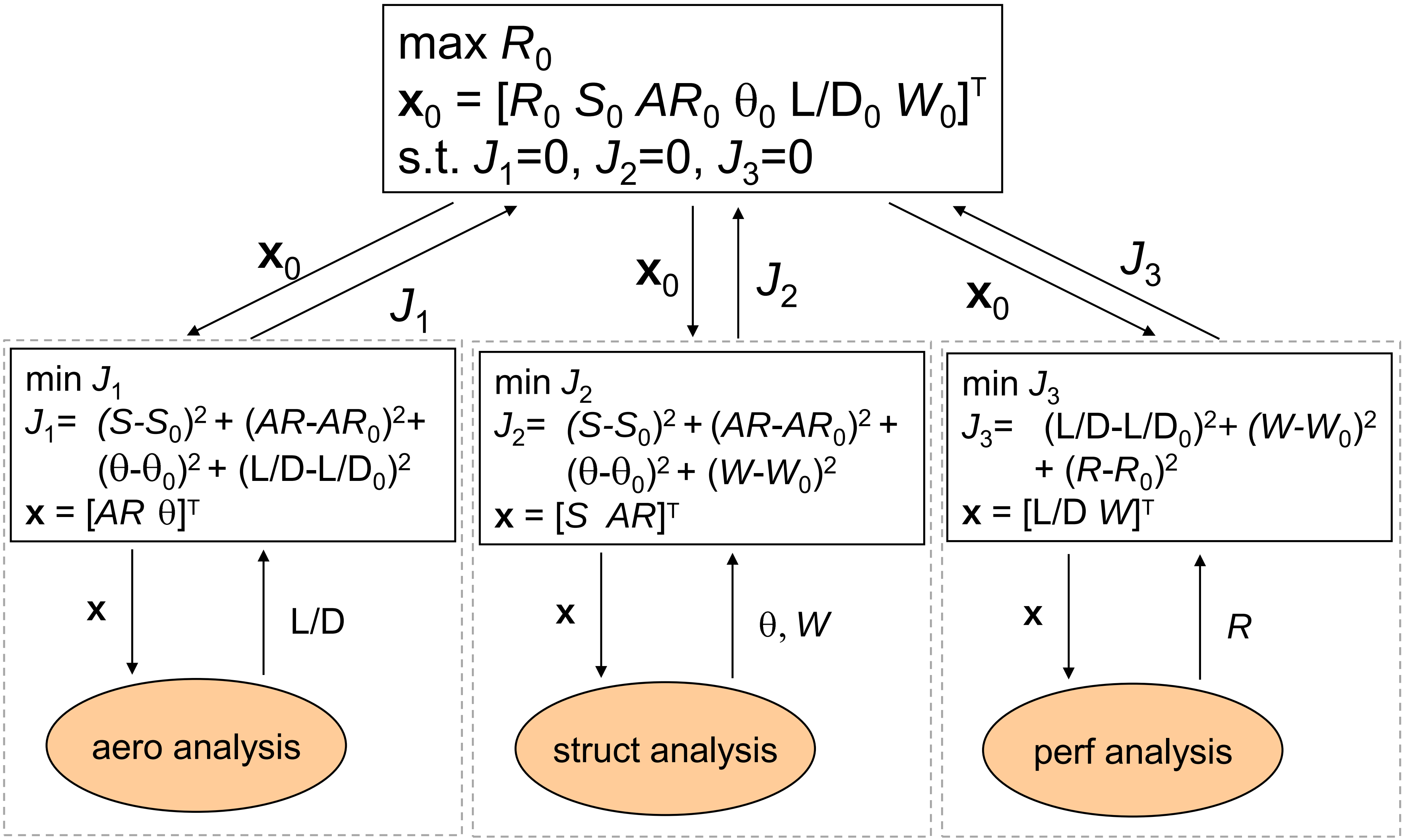
CO Example: Aircraft Design

Consider a simple aircraft design problem:
maximize range for a given take-off weight by choosing wing area, aspect ratio, twist angle, L/D, and wing weight.

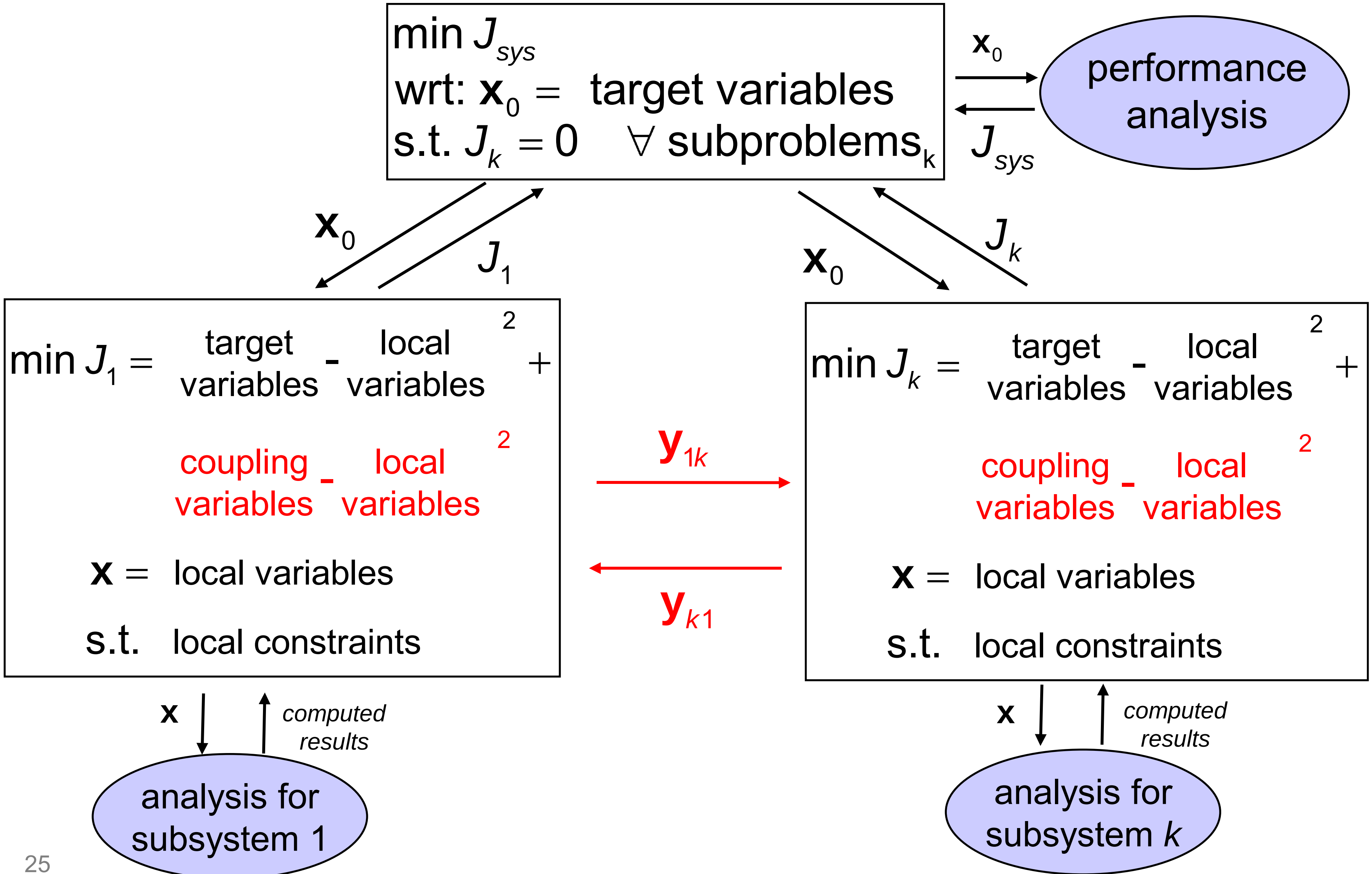


modified from Kroo et al. AIAA 94-4325

CO Example: Aircraft Design



Collaborative Optimization



Collaborative Optimization

$\mathbf{x}_0$ = system-level target variable values

$\mathbf{x}$ = subsystem local variables

$\mathbf{y}_{ij}$ = coupling functions

- $\mathbf{y}_{ij}$ =outputs of subsystem *j* which are needed as inputs to subsystem *i*.
- Coupling equations must also be satisfied, so coupling variables are included in subsystem objective.
- Used to reduce the number of system-level parameters.

Analytical Target Cascading

- ATC was initially developed as a product development tool to cascade system-level product targets through a hierarchy of design groups

- ATC is typically used to solve object-based decomposed system optimization problems

- The ATC paradigm is based on hierarchical organizational and analysis structures

- ATC approach is to take a high-level system analysis and use more detailed subsystem analyses at the lower levels.

Analytical Target Cascading

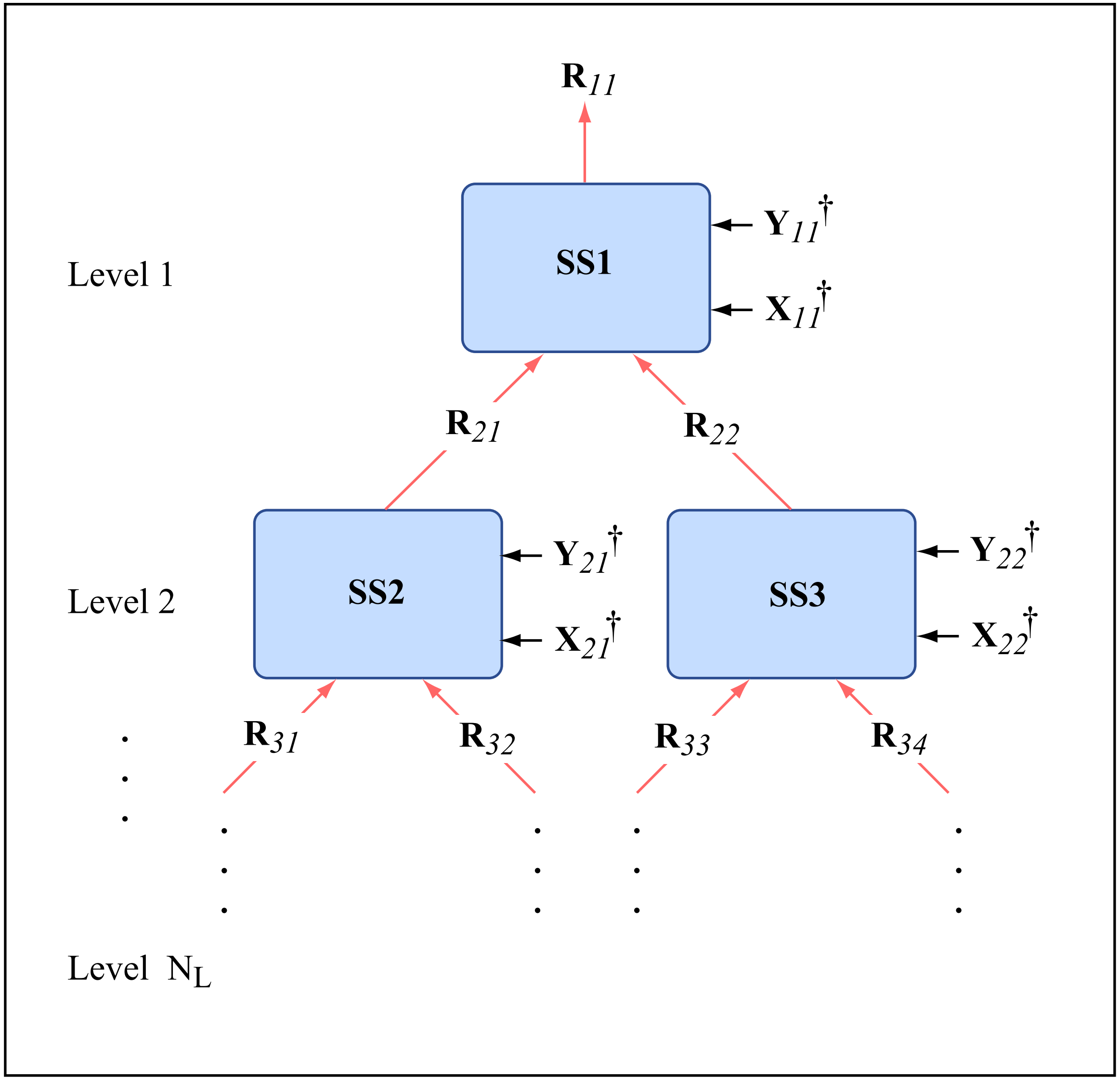


Image by MIT OpenCourseWare.

Analytical Target Cascading

- ATC's mathematical formulation is similar to CO although they were developed with different motivations.

- Bottom level problems have the most design freedom. Many possible solutions can exist that both match targets while satisfying local design constraints.

- At higher levels design freedom is progressively reduced, until it is a minimum at the top level.

Analytical Target Cascading

Top Level Problem

$\mathbf{P}_{sup}$

$$\min_{\mathbf{x}^{\dagger}_{sup}=\{\mathbf{x}_{sup},\mathbf{y}^{\dagger}_{s},\mathbf{R}_{s},\varepsilon_R,\varepsilon_y\}} \quad ||\mathbf{R}_{sup}-\mathbf{T}_{sup}||+\varepsilon_R+\varepsilon_y$$

$$\text{subject to} \quad \sum_{k\in C_{sup}} ||\mathbf{R}_{s,k}-\mathbf{R}^{L}_{s,k}|| \le \varepsilon_R$$

$$\sum_{k\in C_{sup}} ||\mathbf{y}^{\dagger}_{s,k}-\mathbf{y}^{\dagger L}_{s,k}|| \le \varepsilon_y$$

$$g_{sup}(\mathbf{x}_{sup},\ \mathbf{R}_s) \le 0$$

$$h_{sup}(\mathbf{x}_{sup},\ \mathbf{R}_s) \le 0$$

Intermediate Level Problem

$\mathbf{P}_{s,j}$

$$\min_{\mathbf{x}^{\dagger}_{s,j}=\{\mathbf{x}_{s,j},\mathbf{y}^{\dagger}_{s,j},\mathbf{y}^{\dagger}_{ss},\mathbf{R}_{ss},\varepsilon_R,\varepsilon_y\}} \quad ||\mathbf{R}_{s,j}-\mathbf{R}^{U}_{s,j}||+||\mathbf{y}_{s,j}-\mathbf{y}^{U}_{s,j}||+\varepsilon_R+\varepsilon_y$$

$$\text{subject to} \quad \sum_{k\in C_{s,j}} ||\mathbf{R}_{ss,k}-\mathbf{R}^{L}_{ss,k}|| \le \varepsilon_R$$

$$\sum_{k\in C_{s,j}} ||\mathbf{y}^{\dagger}_{ss,k}-\mathbf{y}^{\dagger L}_{ss,k}|| \le \varepsilon_y$$

$$g_{s,j}(\mathbf{x}_{s,j},\ \mathbf{y}^{\dagger}_{s,j},\ \mathbf{R}_{s,j}) \le 0$$

$$h_{s,j}(\mathbf{x}_{s,j},\ \mathbf{y}^{\dagger}_{s,j},\ \mathbf{R}_{s,j}) \le 0$$

Bottom Level Problem

$\mathbf{P}_{ss,j}$

$$\min_{\mathbf{x}^{\dagger}_{ss,j}=\{\mathbf{x}_{ss,j},\mathbf{y}^{\dagger}_{ss,j}\}} \quad ||\mathbf{R}_{ss,j}-\mathbf{R}^{U}_{ss,j}||+||\mathbf{y}_{ss,j}-\mathbf{y}^{U}_{ss,j}||$$

$$\text{subject to} \quad g_{s,j}(\mathbf{x}_{s,j},\ \mathbf{y}^{\dagger}_{s,j},\ \mathbf{R}_{s,j}) \le 0$$

$$h_{s,j}(\mathbf{x}_{s,j},\ \mathbf{y}^{\dagger}_{s,j},\ \mathbf{R}_{s,j}) \le 0$$

Analytical Target Cascading

- Linking variables *y*: Quantities that are input to more than one subspace. These could be either shared variables or coupling variables.

- Local decision variables *x*: Variables that a particular subspace determines the value of.

- Responses *R*: Values generated by subspaces required as inputs to respective parent subspaces.

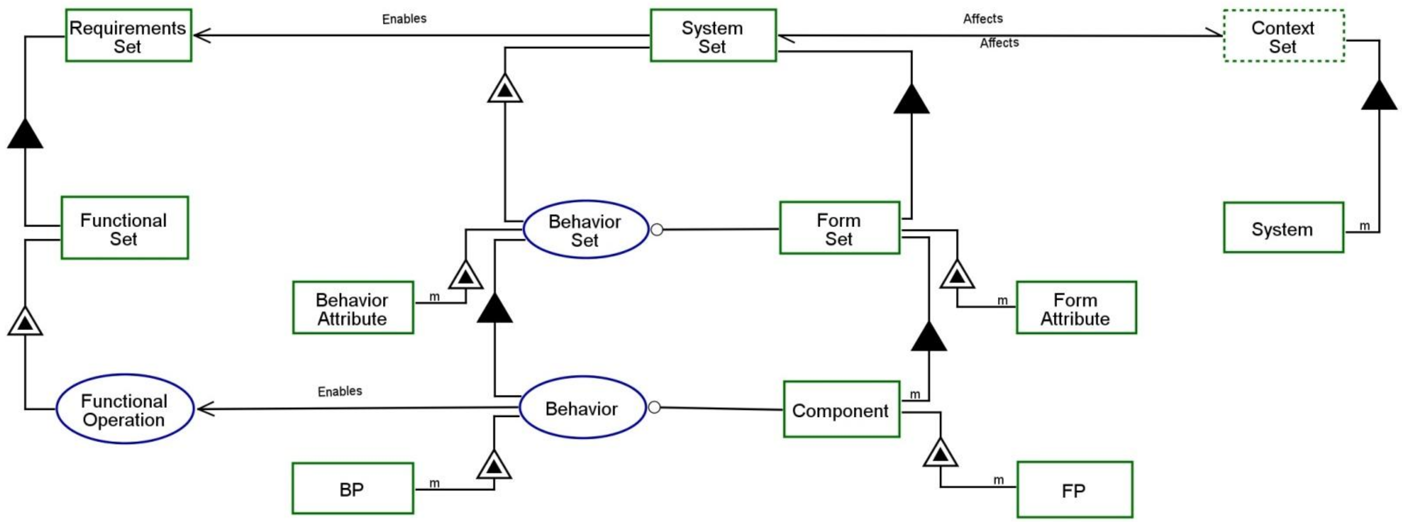
- Targets *T*: Values set by parent subspaces to be matched by the corresponding quantities from child subspaces.

- εR and εy: allowable compatibility tolerance.

Hierarchical Decomposition & Multi-Domain Formulation

16.888 ESD.77

Decomposition

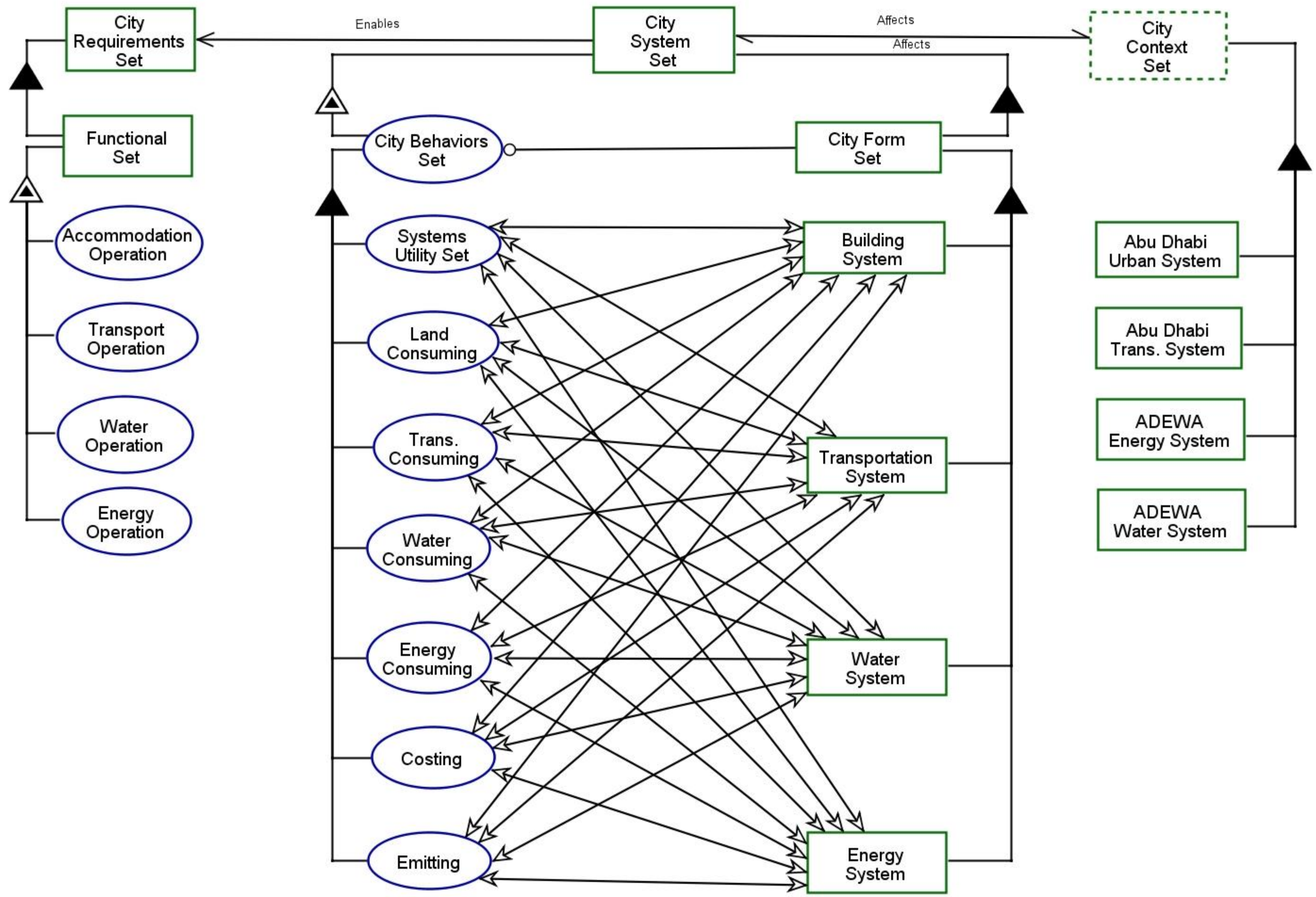


Courtesy of Anas Alfaris. Used with permission.

Hierarchical Decomposition & Multi-Domain Formulation

Decomposition

L1

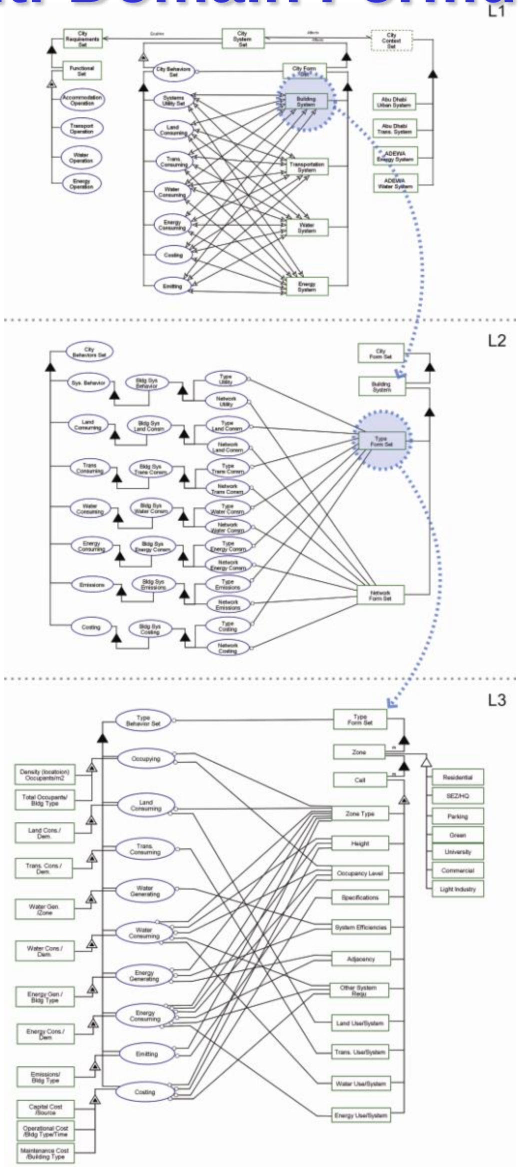


Courtesy of Anas Alfaris.Used with permission.

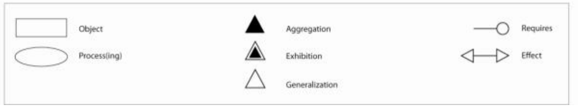
Hierarchical Decomposition & Multi-Domain Formulation

Decomposition

Building System



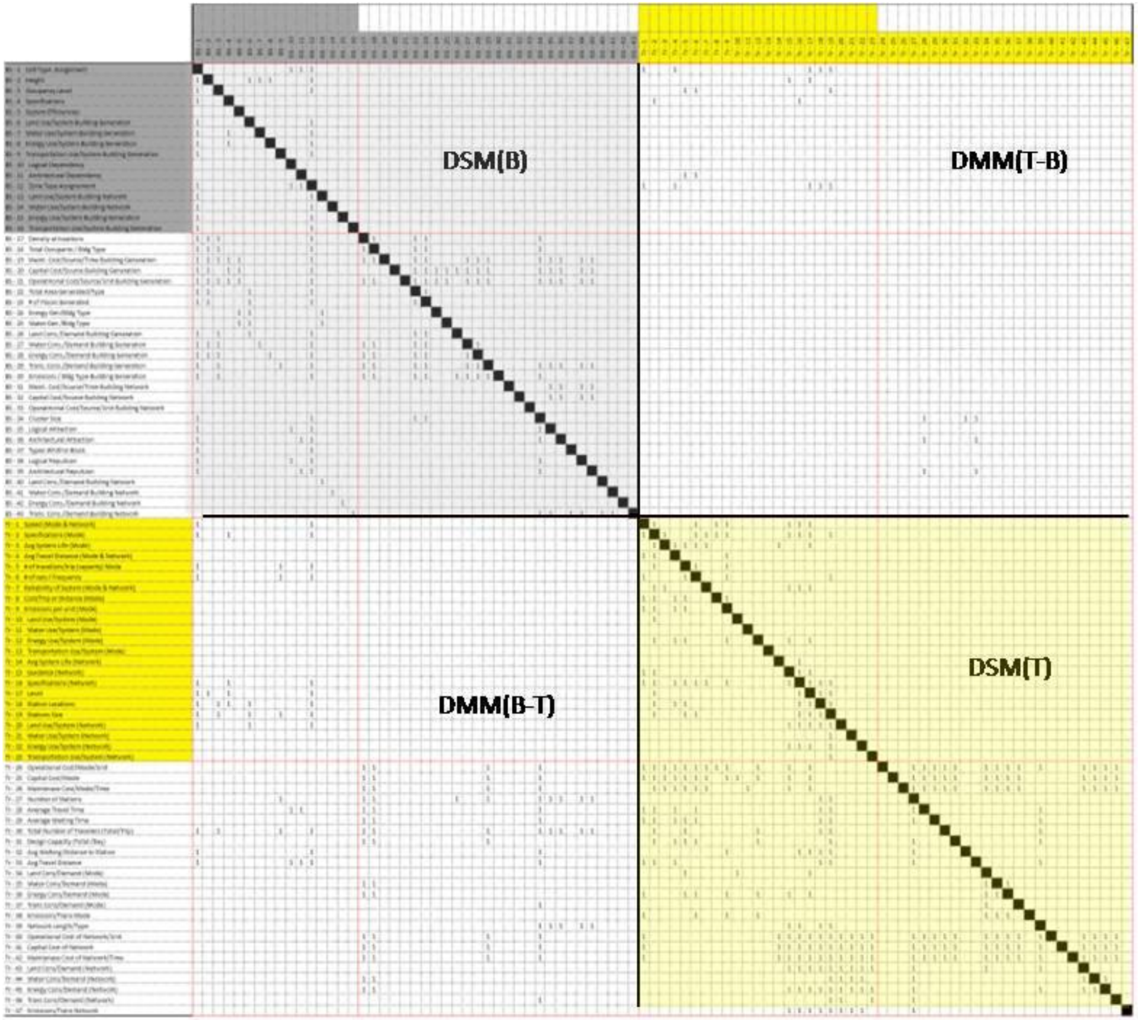
OPD Level 3	System	Type	# of FPs	# of BPs
	Building	Mode	11	14
		Network	12	8
	Water	Mode	19	10
		Network	17	13
	Energy	Mode	21	10
		Network	14	13
	Transportation	Mode	18	16
		Network	21	11



Courtesy of Anas Alfaris. Used with permission.

Hierarchical Decomposition & Multi-Domain Formulation

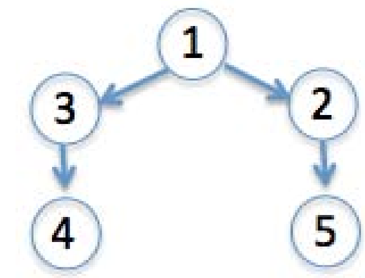
Formulation



Courtesy of Anas Alfaris. Used with permission.

Hierarchical Decomposition & Multi-Domain Formulation

Formulation



$$A = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \end{bmatrix} \quad A^2 = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 \end{bmatrix} \quad A^3 = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

$$V = \sum_{n=1}^{4} A^n = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 \end{bmatrix}$$

$$\pi_j = \frac{1}{N} \sum_{i=1}^{N} a_{ij}$$

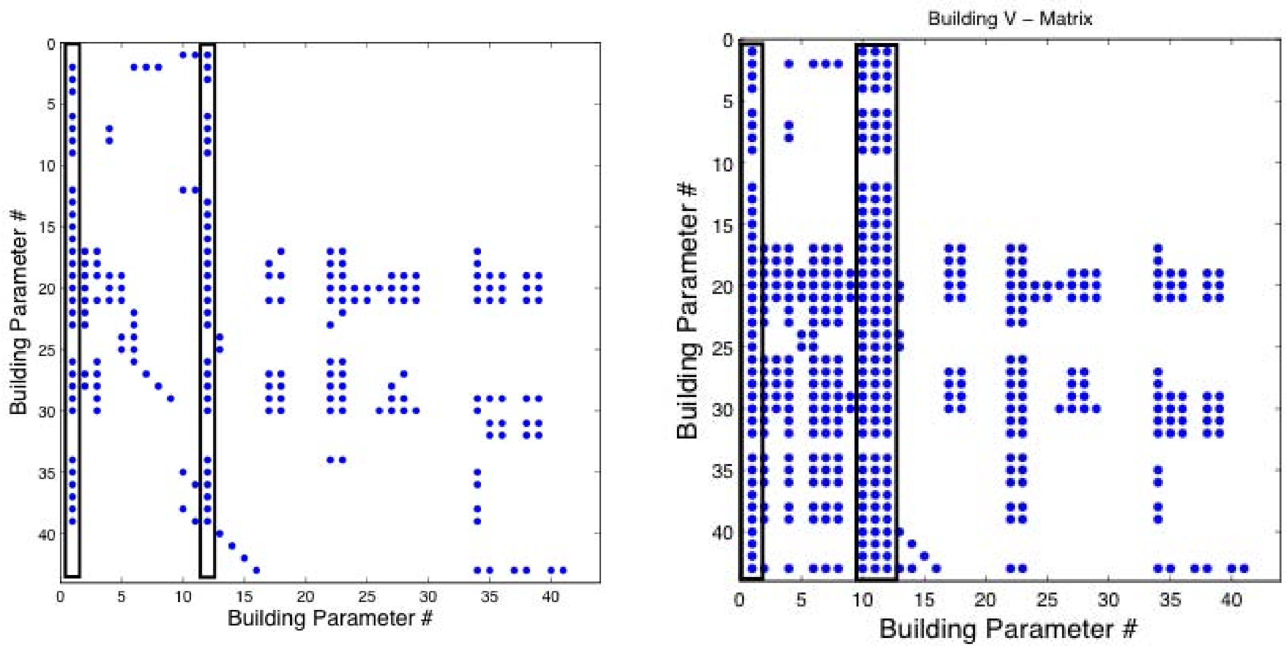
$$\delta_i = \frac{1}{N} \sum_{j=1}^{N} a_{ij}$$

Where:
π_j is an indicator of the fraction of total elements to which element *j* provides input,
δ_i is the fraction of total elements on which element *i* depends, and
$\mathbf{a_{ij}}$ is an element of a matrix that can be the DSM, a power of the DSM, or the V matrix.

Hierarchical Decomposition & Multi-Domain Formulation

16.888
ESD.77

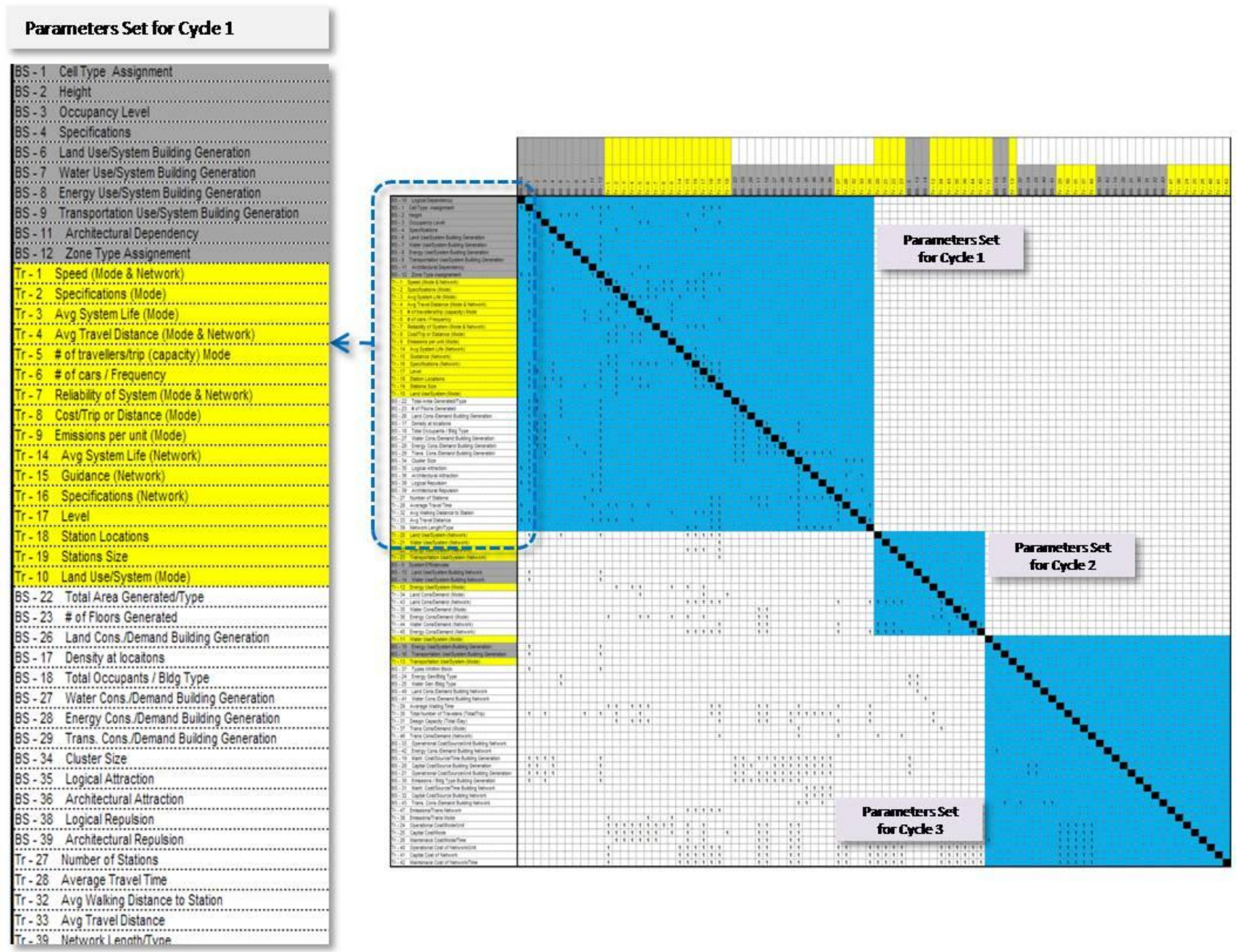
Formulation



Courtesy of Anas Alfaris. Used with permission.

Hierarchical Decomposition & Multi-Domain Formulation

Formulation



Courtesy of Anas Alfaris. Used with permission.

References

I.P. Sobieski and I.M. Kroo. *Collaborative Optimization Using Response Surface Estimation*. AIAA Journal Vol. 38 No. 10. Oct 2000.

Erin J. Cramer et al. *Problem Formulation for Multidisciplinary Optimization*. SIAM Journal of Optimization. Vol. 4, No. 4 pp. 754-776, Nov 1994.

Kim, H.M., Michelena, N.F., Papalambros, P.Y., and Jiang, T., "Target Cascading in Optimal System Design," *Transaction of ASME: Journal of Mechanical Design*, Vol. 125, pp. 481-489, 2003