Challenges in simulation-based optimization

Natalia Alexandrov
Multidisciplinary Optimization Branch
NASA Langley Research Center

Robert Michael Lewis
Dept. of Mathematics
College of William & Mary

SIAM Conference on Computational Science & Engineering

Multidisciplinary optimization (MDO)

MDO = systematic approaches to the design of complex, coupled systems.

Multidisciplinary optimization (MDO)

MDO = systematic approaches to the design of complex, coupled systems.

Multidisciplinary refers to different aspects of the design problem (e.g., aerodynamics, structures, control, &c., for aircraft).

Multidisciplinary optimization (MDO)

MDO $=$ systematic approaches to the design of complex, coupled systems.

Multidisciplinary refers to different aspects of the design problem (e.g., aerodynamics, structures, control, &c., for aircraft).

Here: MDO is portion of the design process that can be meaningfully formulated as a nonlinear program (NLP).

Multidisciplinary optimization (MDO)

MDO = systematic approaches to the design of complex, coupled systems.

Multidisciplinary refers to different aspects of the design problem (e.g., aerodynamics, structures, control, &c., for aircraft).

Here: MDO is portion of the design process that can be meaningfully formulated as a nonlinear program (NLP).

This, of course, does not do justice to the real problem.

Efficiency in solving MDO problems

- Effects of problem formulation

Efficiency in solving MDO problems

- Effects of problem formulation
  - Computational efficiency

Efficiency in solving MDO problems

- Effects of problem formulation
  - Computational efficiency
  - Robustness and convergence to solutions

Efficiency in solving MDO problems

- Effects of problem formulation
  - Computational efficiency
  - Robustness and convergence to solutions
  - Disciplinary autonomy: Allowing disciplinary design occur as independently as possible

Efficiency in solving MDO problems

- Effects of problem formulation
  - Computational efficiency
  - Robustness and convergence to solutions
  - Disciplinary autonomy: Allowing disciplinary design occur as independently as possible
- Efficiency in problem integration.

Efficiency in solving MDO problems

- Effects of problem formulation
  - Computational efficiency
  - Robustness and convergence to solutions
  - Disciplinary autonomy: Allowing disciplinary design occur as independently as possible
- Efficiency in problem integration.
- Computational efficiency of disciplinary components.

Problem formulation

Analytical features of MDO problem formulation strongly influence the practical ability of optimization algorithms to solve the MDO problem reliably and efficiently.

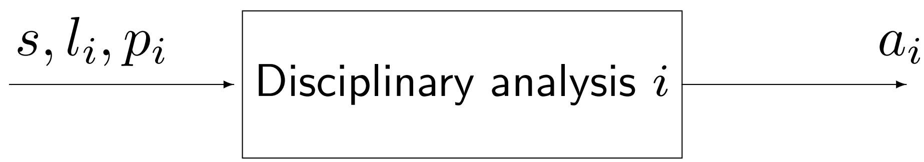
Problem formulation

Analytical features of MDO problem formulation strongly influence the practical ability of optimization algorithms to solve the MDO problem reliably and efficiently.

	1	2	3	4	5	6	7
FIO	610	220	610	81	3234	5024	8730
CO	15626	19872	1785	2102	837	40125	691058
DAO	9530	8796	382	-	544	932	-

Cost of optimization in terms of the number of analyses required,
3 different formulations, 7 test problems.
Alexandrov and Kodiyalam (1998)

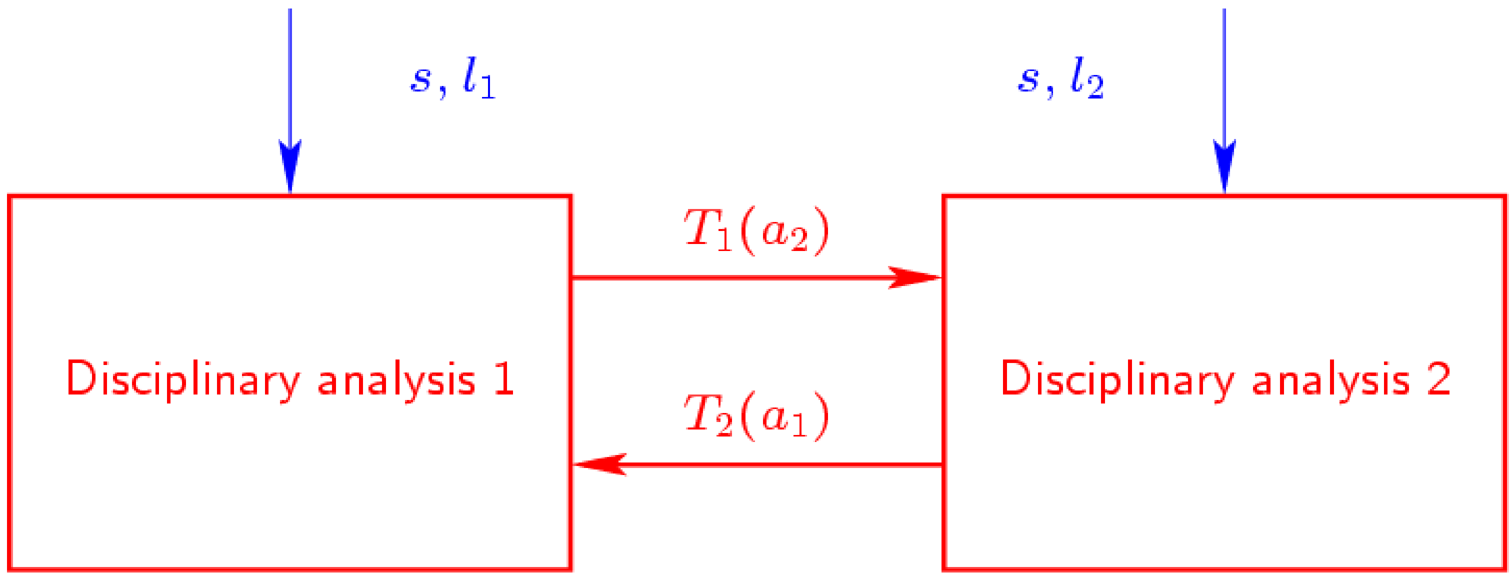
Disciplinary analysis



Disciplinary analysis:

$$A_i(a_i; p_i, s, l_i) \quad = \quad 0.$$

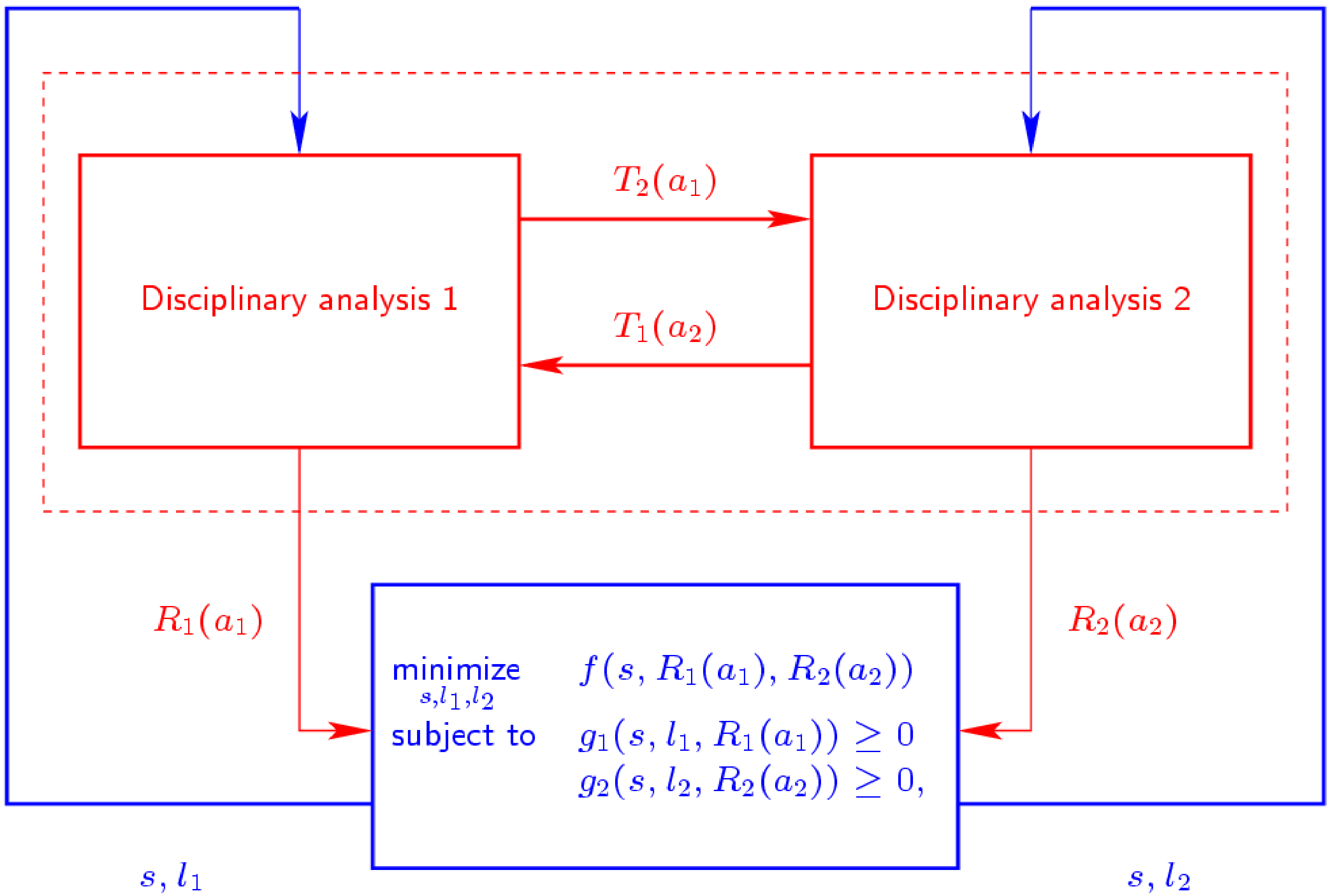
Multidisciplinary analysis: Two discipline example



Multidisciplinary analysis (MDA):

$$\begin{aligned} A_1(a_1, T_1(a_2); s, l_1) &= 0 \\ A_2(a_2, T_2(a_1); s, l_2) &= 0. \end{aligned}$$

Fully integrated optimization (FIO)



Fully integrated optimization (FIO)

Advantages:

- Uses extant MDA capability.

Fully integrated optimization (FIO)

Advantages:

- Uses extant MDA capability.
- Intermediate designs are physically sensible.

Fully integrated optimization (FIO)

Advantages:

- Uses extant MDA capability.
- Intermediate designs are physically sensible.

Disadvantages:

- Requires MDA capability.

Fully integrated optimization (FIO)

Advantages:

- Uses extant MDA capability.
- Intermediate designs are physically sensible.

Disadvantages:

- Requires MDA capability.
- Limited disciplinary autonomy.

Fully integrated optimization (FIO)

The NLP:

$$\begin{array}{ll} \underset{s,l_1,l_2}{\text{minimize}} & f(s, R_1(a_1(s,l_1,l_2)), R_2(a_2(s,l_1,l_2))) \\ \text{subject to} & g_1(s, l_1, R_1(a_1(s,l_1,l_2))) \geq 0 \\ & g_2(s, l_2, R_2(a_2(s,l_1,l_2))) \geq 0, \end{array}$$

where $a_1(s,l_1,l_2)$, $a_2(s,l_1,l_2)$ solve the MDA:

$$\begin{array}{rcl} A_1(a_1, T_1(a_2); s, l_1) & = & 0 \\ A_2(a_2, T_2(a_1); s, l_2) & = & 0. \end{array}$$

Simultaneous analysis and design (SAND)

Relax all coupling in the problem; all variables are independent:

$$
\begin{array}{ll}
\underset{s,l_1,l_2,a_1,a_2}{\text{minimize}} & f(s, R_1(a_1), R_2(a_2)) \\
\text{subject to} & g_1(s, l_1, R_1(a_1)) \geq 0 \\
& g_2(s, l_2, R_2(a_2)) \geq 0 \\
& A_1(a_1, T_1(a_2), s, l_1) = 0 \\
& A_2(a_2, T_2(a_1), s, l_2) = 0.
\end{array}
$$

Simultaneous analysis and design

Advantages:

- Reduced nonlinearity in the NLP.

Simultaneous analysis and design

Advantages:

- Reduced nonlinearity in the NLP.
- No expensive analyses when far from an optimal solution.

Simultaneous analysis and design

Advantages:

- Reduced nonlinearity in the NLP.
- No expensive analyses when far from an optimal solution.

Disadvantages:

- Potential for a humongous number of variables in the NLP.

Simultaneous analysis and design

Advantages:

- Reduced nonlinearity in the NLP.
- No expensive analyses when far from an optimal solution.

Disadvantages:

- Potential for a humongous number of variables in the NLP.
- Solution techniques from disciplinary analyses must be integrated with the optimization algorithm.

Simultaneous analysis and design

Advantages:

- Reduced nonlinearity in the NLP.
- No expensive analyses when far from an optimal solution.

Disadvantages:

- Potential for a humongous number of variables in the NLP.
- Solution techniques from disciplinary analyses must be integrated with the optimization algorithm.
- Intermediate designs may not be physically sensible.

Simultaneous analysis and design

Advantages:

- Reduced nonlinearity in the NLP.
- No expensive analyses when far from an optimal solution.

Disadvantages:

- Potential for a humongous number of variables in the NLP.
- Solution techniques from disciplinary analyses must be integrated with the optimization algorithm.
- Intermediate designs may not be physically sensible.
- Role of disciplinary autonomy not clear.

Distributed analysis optimization (DAO)

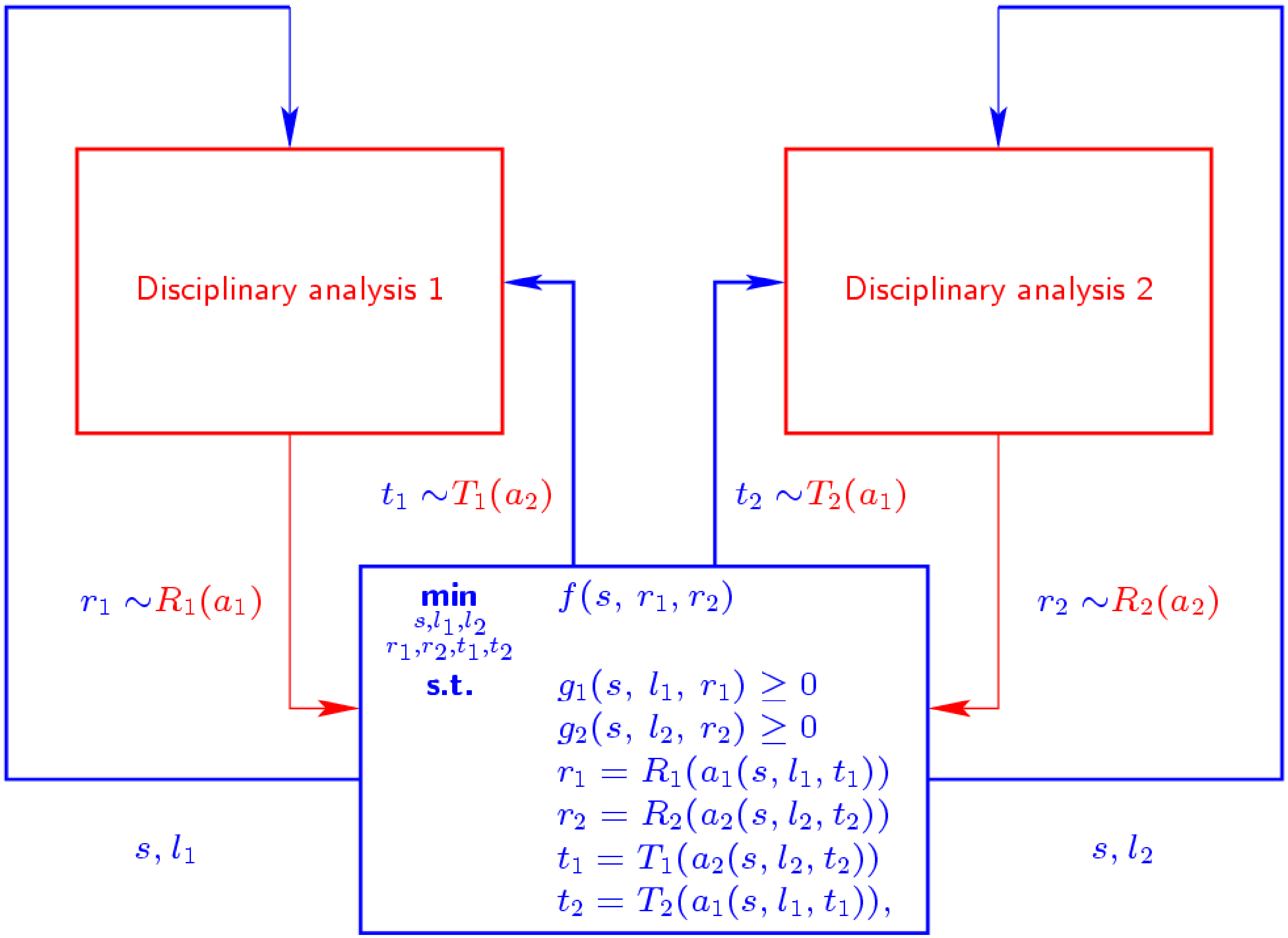
Relax the coupling in the MDA, but preserve the disciplinary analyses:

$$
\begin{array}{cl}
\underset{s,l_1,l_2,r_1,r_2,t_1,t_2}{\textbf{min}} & f(s,\, r_1, r_2) \\
\textbf{s.t.} & g_1(s,\, l_1,\, r_1) \geq 0 \\
& g_2(s,\, l_2,\, r_2) \geq 0 \\
& r_1 = R_1(a_1(s, l_1, t_1)) \quad t_1 = T_1(a_2(s, l_2, t_2)) \\
& r_2 = R_2(a_2(s, l_2, t_2)) \quad t_2 = T_2(a_1(s, l_1, t_1)),
\end{array}
$$

where $a_1 = a_1(s, l_1, t_1)$, $a_2 = a_2(s, l_2, t_2)$ satisfy the disciplinary analyses:

$$
\begin{aligned}
A_1(a_1;\, t_1, s, l_1) &= 0 \\
A_2(a_2;\, t_2, s, l_2) &= 0.
\end{aligned}
$$

Distributed analysis optimization (DAO)



Distributed analysis optimization (DAO)

Advantages:

- Conventional one-level NLP.

Distributed analysis optimization (DAO)

Advantages:

- Conventional one-level NLP.
- More disciplinary autonomy than in FIO.

Distributed analysis optimization (DAO)

Advantages:

- Conventional one-level NLP.
- More disciplinary autonomy than in FIO.
- Fewer independent variables than in SAND.

Distributed analysis optimization (DAO)

Advantages:

- Conventional one-level NLP.
- More disciplinary autonomy than in FIO.
- Fewer independent variables than in SAND.

Disadvantages:

- Intermediate designs may not be physically sensible (though perhaps "less infeasible" than in SAND).

Distributed analysis optimization (DAO)

Advantages:

- Conventional one-level NLP.
- More disciplinary autonomy than in FIO.
- Fewer independent variables than in SAND.

Disadvantages:

- Intermediate designs may not be physically sensible (though perhaps "less infeasible" than in SAND).
- Disciplinary autonomy still limited.

Distributed analysis optimization (DAO)

Advantages:

- Conventional one-level NLP.
- More disciplinary autonomy than in FIO.
- Fewer independent variables than in SAND.

Disadvantages:

- Intermediate designs may not be physically sensible (though perhaps "less infeasible" than in SAND).
- Disciplinary autonomy still limited.

DAO is similar to the Individual Discipline Feasible (IDF) formulation.

Hierarchy of formulations

Various problem formulations can be viewed as being derived from SAND through the elimination of variables.

SAND—all variables are independent (s, l_1, l_2, a_1, a_2).

Hierarchy of formulations

Various problem formulations can be viewed as being derived from SAND through the elimination of variables.

SAND—all variables are independent (s, l_1, l_2, a_1, a_2).

Eliminate (a_1, a_2) via multidisciplinary analysis $\Rightarrow$ FIO.

Hierarchy of formulations

Various problem formulations can be viewed as being derived from SAND through the elimination of variables.

SAND—all variables are independent (s, l_1, l_2, a_1, a_2).

Eliminate (a_1, a_2) via multidisciplinary analysis $\Rightarrow$ FIO.

Introduce coupling variables (a_{12}, a_{21})
$+$ eliminate (a_1, a_2) via disciplinary analyses $\Rightarrow$ DAO

Hierarchy of formulations

Various problem formulations can be viewed as being derived from SAND through the elimination of variables.

SAND—all variables are independent (s, l_1, l_2, a_1, a_2).

Eliminate (a_1, a_2) via multidisciplinary analysis $\Rightarrow$ FIO.

Introduce coupling variables (a_{12}, a_{21})
$+$ eliminate (a_1, a_2) via disciplinary analyses $\Rightarrow$ DAO

Introduce coupling variables (a_{12}, a_{21}),
$+$ eliminate (a_1, a_2) via disciplinary analyses
$+$ eliminate (l_1, l_2) via disciplinary design constraints $\Rightarrow$???

Elimination of local design variables

One idea is to eliminate l_1, l_2 by solving disciplinary optimization problems involving these variables.

Elimination of local design variables

One idea is to eliminate l_1, l_2 by solving disciplinary optimization problems involving these variables. This leads to a bilevel program.

Elimination of local design variables

One idea is to eliminate l_1, l_2 by solving disciplinary optimization problems involving these variables. This leads to a bilevel program.

An illustration is an approach sometimes called collaborative optimization.

System-level problem:

$$\begin{aligned} \min_{s,r_1,r_2,t_1,t_2} \quad & f(s,r_1,r_2) \\ \text{s.t.} \quad & c(s,r_1,r_2,t_1,t_2) = 0, \end{aligned}$$

System-level problem:

$$\begin{array}{ll} \min\limits_{s,r_1,r_2,t_1,t_2} & f(s,r_1,r_2) \\ \textbf{s.t.} & c(s,r_1,r_2,t_1,t_2)=0, \end{array}$$

Disciplinary problems:

$$(1) \qquad \begin{array}{ll} \min\limits_{l_1} & \| \, r_1 - R_1(a_1(s,l_1,t_1)) \, \|^2 + \| \, t_2 - T_2(a_1(s,l_1,t_1)) \, \|^2 \\ \textbf{s.t.} & g_1(s,l_1,a_1(s,l_1,t_1)) \geq 0 \\ \textbf{where} & a_1 = A_1(s,l_1,t_1). \end{array}$$

System-level problem:

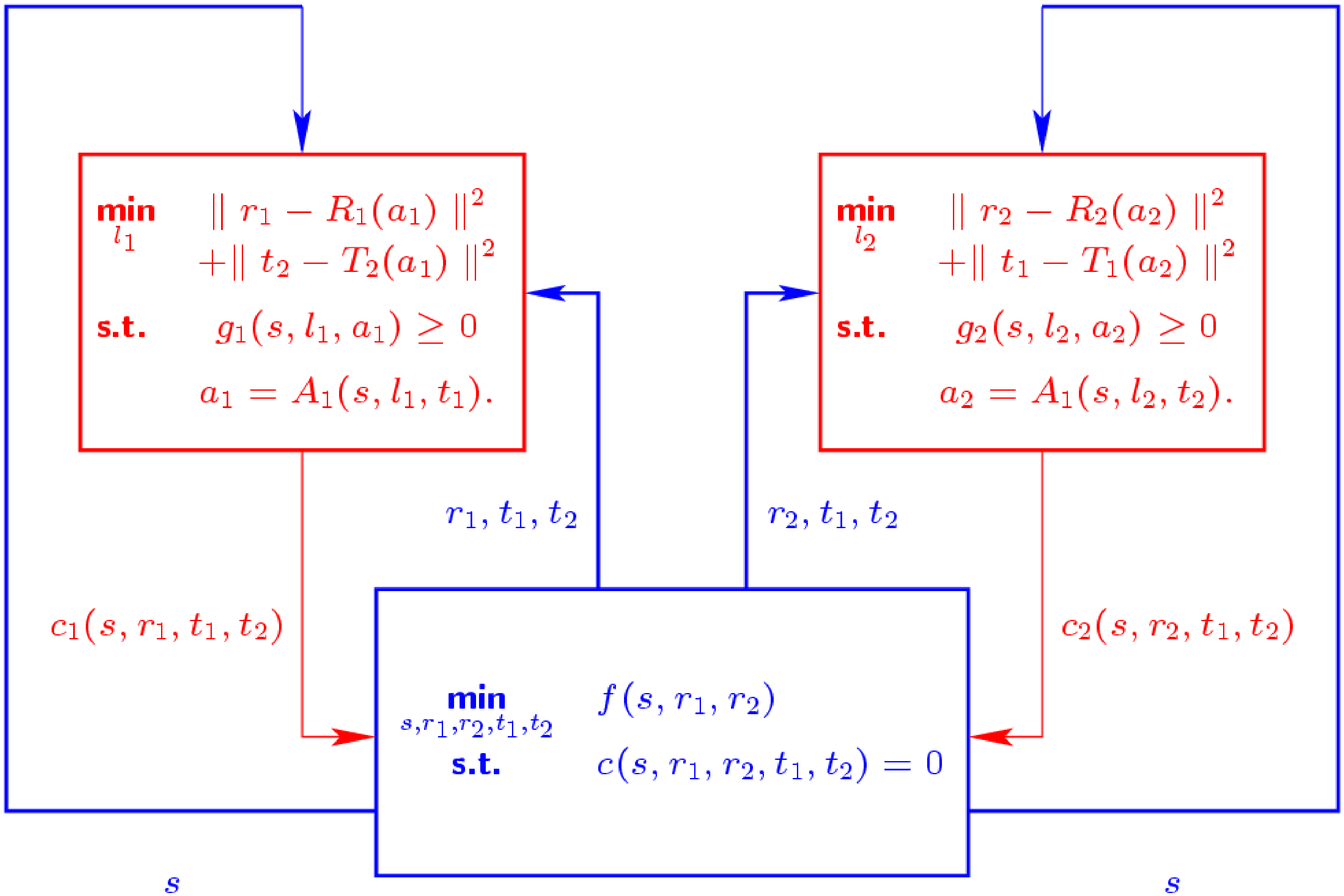
$$\begin{array}{cl} \min\limits_{s,r_1,r_2,t_1,t_2} & f(s,r_1,r_2) \\ \text{s.t.} & c(s,r_1,r_2,t_1,t_2)=0, \end{array}$$

Disciplinary problems:

$$(1) \quad \begin{array}{lc} \min\limits_{l_1} & \| \, r_1 - R_1(a_1(s,l_1,t_1)) \, \|^2 + \| \, t_2 - T_2(a_1(s,l_1,t_1)) \, \|^2 \\ \text{s.t.} & g_1(s,l_1,a_1(s,l_1,t_1)) \geq 0 \\ \text{where} & a_1 = A_1(s,l_1,t_1). \end{array}$$

$$(2) \quad \begin{array}{lc} \min\limits_{l_2} & \| \, r_2 - R_2(a_2(s,l_2,t_2)) \, \|^2 + \| \, t_1 - T_1(a_2(s,l_2,t_2)) \, \|^2 \\ \text{s.t.} & g_2(s,l_2,a_2(s,l_2,t_2)) \geq 0 \\ \text{where} & a_2 = A_2(s,l_2,t_2). \end{array}$$

Elimination of local design variables via CO



System-level constraints

Let $\bar{l}_1(s, l_1, t_2)$ and $\bar{l}_2(s, l_2, t_1)$ be the solutions of the disciplinary optimization problems.

System-level constraints

Let $\bar{l}_1(s, l_1, t_2)$ and $\bar{l}_2(s, l_2, t_1)$ be the solutions of the disciplinary optimization problems.

One choice of system-level consistency constraints is:

$$\begin{aligned} c_1(s, r_1, r_2, t_1, t_2) &= \| r_1 - R_1(a_1(s, \bar{l}_1, t_1)) \|^2 + \| t_2 - T_2(a_1(s, \bar{l}_1, t_1)) \|^2 \\ c_2(s, r_1, r_2, t_1, t_2) &= \| r_2 - R_2(a_2(s, \bar{l}_2, t_2)) \|^2 + \| t_1 - T_1(a_2(s, \bar{l}_2, t_2)) \|^2 \end{aligned}$$

$c_i(s, r_1, r_2, t_1, t_2) = 0 \Leftrightarrow$ Discipline i can find local design variables $\bar{l}_i$ such that the analysis output is consistent with the targets r_i and t_i.

System-level constraints

Let $\bar{l}_1(s, l_1, t_2)$ and $\bar{l}_2(s, l_2, t_1)$ be the solutions of the disciplinary optimization problems.

One choice of system-level consistency constraints is:

$$
\begin{aligned}
c_1(s, r_1, r_2, t_1, t_2) &= \| r_1 - R_1(a_1(s, \bar{l}_1, t_1)) \|^2 + \| t_2 - T_2(a_1(s, \bar{l}_1, t_1)) \|^2 \\
c_2(s, r_1, r_2, t_1, t_2) &= \| r_2 - R_2(a_2(s, \bar{l}_2, t_2)) \|^2 + \| t_1 - T_1(a_2(s, \bar{l}_2, t_2)) \|^2
\end{aligned}
$$

$c_i(s, r_1, r_2, t_1, t_2) = 0 \Leftrightarrow$ Discipline i can find local design variables $\bar{l}_i$ such that the analysis output is consistent with the targets r_i and t_i.

We call such values of s, r_i, t_i *realizable* designs for Discipline i.

Elimination of local design variables

Advantages:

- Marked degree of disciplinary autonomy.

Elimination of local design variables

Advantages:

- Marked degree of disciplinary autonomy.

Disadvantages:

- Disciplinary autonomy exercised in auxiliary disciplinary optimization problems.

Elimination of local design variables

Advantages:

- Marked degree of disciplinary autonomy.

Disadvantages:

- Disciplinary autonomy exercised in auxiliary disciplinary optimization problems.

- Bilevel program.

Elimination of local design variables

Advantages:

- Marked degree of disciplinary autonomy.

Disadvantages:

- Disciplinary autonomy exercised in auxiliary disciplinary optimization problems.
- Bilevel program.
- Badly behaved system-level problem.

Analytical difficulties in eliminating local design variables

Approaches that eliminate local design variables and enforce interdisciplinary consistency via equality constraints will, in general, suffer from

- Nonsmoothness of the system-level constraints,

Analytical difficulties in eliminating local design variables

Approaches that eliminate local design variables and enforce interdisciplinary consistency via equality constraints will, in general, suffer from

- Nonsmoothness of the system-level constraints, or
- Degeneracy of the system-level constraints

at designs of interest (realizable designs).

Analytical difficulties in eliminating local design variables

Approaches that eliminate local design variables and enforce interdisciplinary consistency via equality constraints will, in general, suffer from

- Nonsmoothness of the system-level constraints, or
- Degeneracy of the system-level constraints

at designs of interest (realizable designs).

Either can cause conventional NLP algorithms to be slow, or to fail.

Analytical difficulties in eliminating local design variables

Approaches that eliminate local design variables and enforce interdisciplinary consistency via equality constraints will, in general, suffer from

- Nonsmoothness of the system-level constraints, or

- Degeneracy of the system-level constraints

at designs of interest (realizable designs).

Either can cause conventional NLP algorithms to be slow, or to fail.

We believe this explains computational difficulties reported with bilevel problem formulations.

(NMA and RML; *AIAA Journal (2001)*,
Optimization and Engineering (to appear))

What is possible?

CONJECTURE. In general, an algorithm for MDO can possess at most two of the following three attributes:

- Computational autonomy.
- Computational efficiency.
- Computational robustness.