Optimum Actuator Selection with a Genetic Algorithm for Aircraft Control

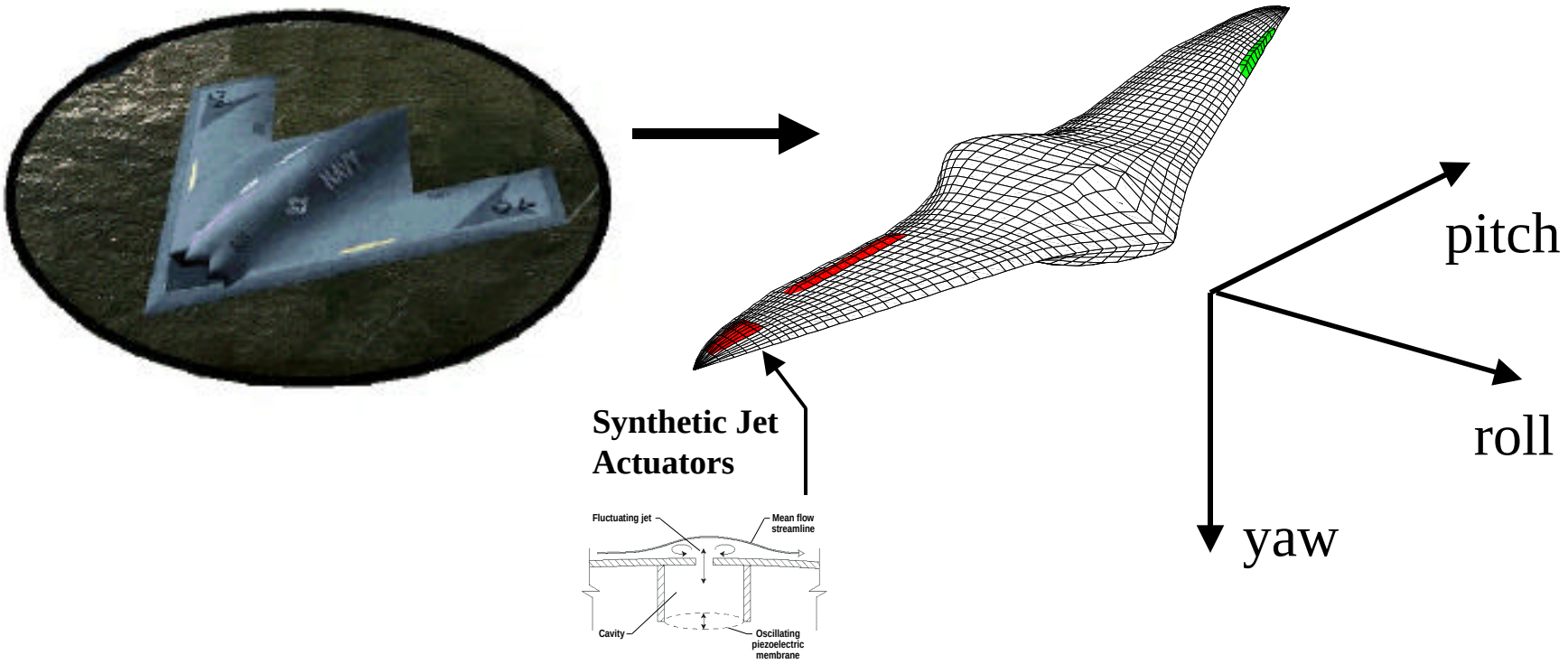
James L. Rogers
NASA Langley Research Center

ANNIE 99

St. Louis, MO

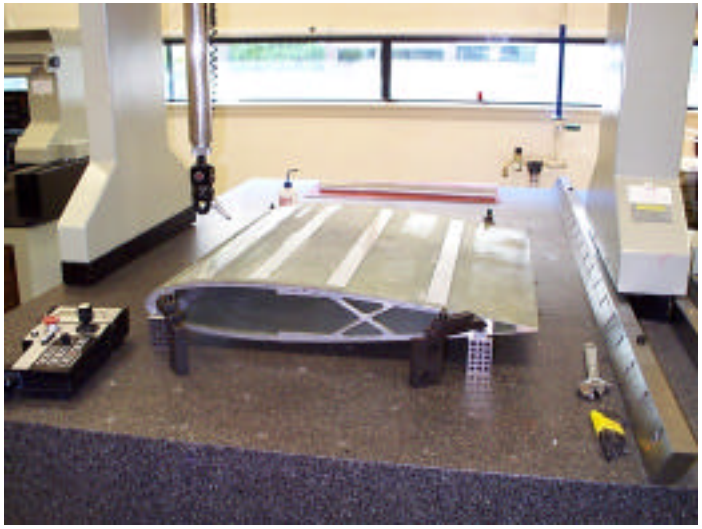
November 7-10, 1999

Why Do This Research?

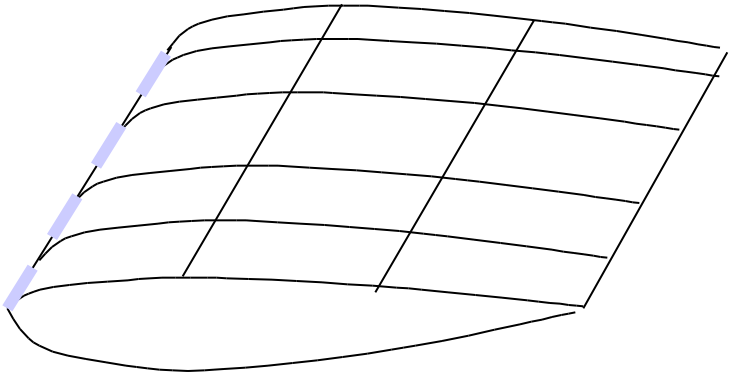


Minimize the number of actuators needed to maximize the uncoupled moments about the pitch, roll, and yaw axes.

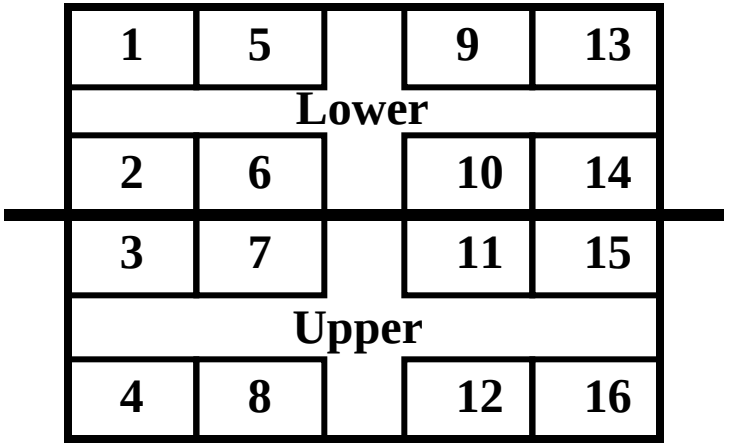
Wind Tunnel Model



Analysis model



Unwrapped model



Leading Edge

Genetic Algorithm Approach

- Rapidly examine a large number of candidate actuator placements.

- Select the optimum placement based on the minimum number of actuators as well as the moment and coupling data.

- The fitness of a population member is determined by calling a low-order, potential-flow program.

- Penalize fitness if constraints are violated.

Single Objective Application

(Maximize Pitch Down)

Given 16 actuator locations, find the best placement for 4 actuators to maximize pitch down with very little or no yaw or roll moments.

Penalize the fitness function if:

- |roll| > .001
- |yaw| > .001
- number actuators not equal to 4

(a) Fitness = 0.041

1	5	9	13
2	6	10	14
3	7	11	15
4	8	12	16

(b) Fitness = 0.074

1	5	9	13
2	6	10	14
3	7	11	15
4	8	12	16

(c) Optimum fitness = 0.078

1	5	9	13
2	6	10	14
3	7	11	15
4	8	12	16

Multiple Objective Application

(One Objective for Each of Pitch, Roll, and Yaw Subproblems)

Given 16 actuator locations, find the minimum number of actuators and their placement to provide uncoupled pitch, roll, and yaw moments.

Penalize the fitness function for the **pitch subproblem** if:

- |roll| > .001
- |yaw| > .001
- number actuators less than some minimum number (initially 4)
- |**pitch**| < .001

Similar penalties for the roll and yaw subproblems.

Multiple Objective Application

GA Information

- Population size = 100 (different populations for each subproblem)
- Crossover - single point crossover
- Mutation rate = .01
- Selection - tournament approach
- Absolute values used for moments
- Wing model is symmetric so information can be used to determine a composite model for all six uncoupled moments

Multiple Objective Application

Computing the Composite Fitness

The string 0 1 1 0 0 0 1 0 0 0 0 1 0 0 1 0
indicates there are actuators in locations 2 3 7 12 and 15

Composite fitness computed using an OR function

Location	1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6
Pitch (4)	0 0 0 1 0 1 0 0 0 1 0 0 0 0 0 1
Roll (4)	0 0 0 0 0 0 1 1 1 1 0 0 0 0 0 0
Yaw (4)	0 0 0 0 0 0 0 0 0 1 1 0 1 0 0 1

Composite (9)	0 0 0 1 0 1 1 1 1 1 1 0 1 0 0 1
Location	1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6

Multiple Objective Application

Problems Encountered and Resolved

- Accepted as valid, strings with pitch, roll, and yaw all zero
  Corrected by adding a penalty

- Crossovers kept producing the same strings
  Corrected by only crossing different strings

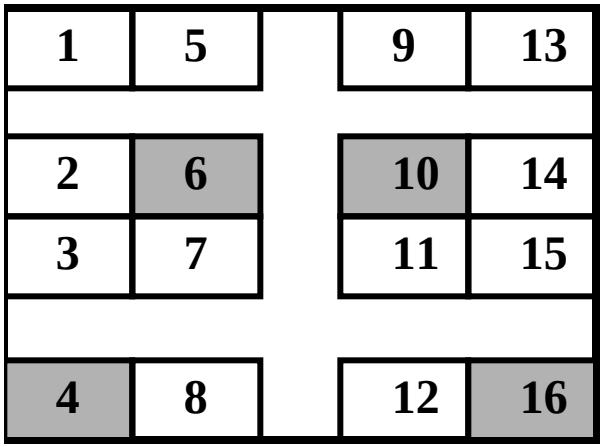
- Originally looked at composite strings inefficiently by computing the composite a member at a time, for example:

Member 5 pitch = 4 roll = 10 yaw = 4 composite = 13
Member 10 pitch = 10 roll = 4 yaw = 10 composite = 10

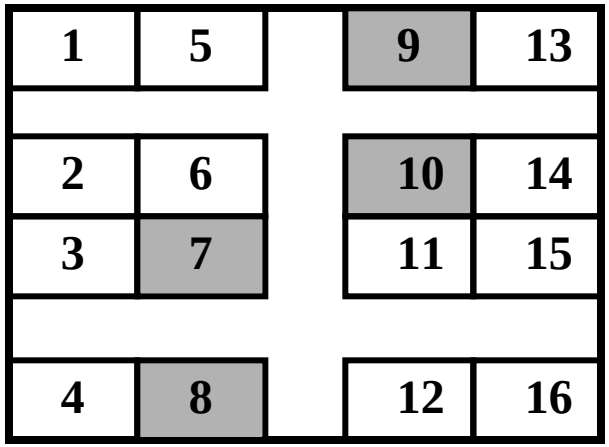
Corrected by saving all valid strings and comparing
pitch = 4 roll = 4 yaw = 4 composite = 9

Four Actuator Placement with Uncoupled Moments (18 Generations)

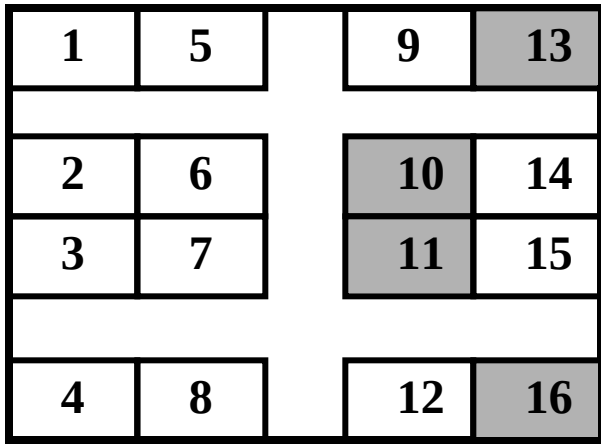
(a) Pitch = 0.05



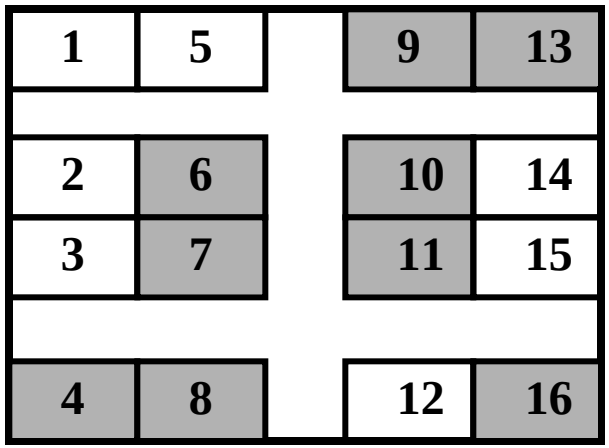
(b) Roll = 0.12



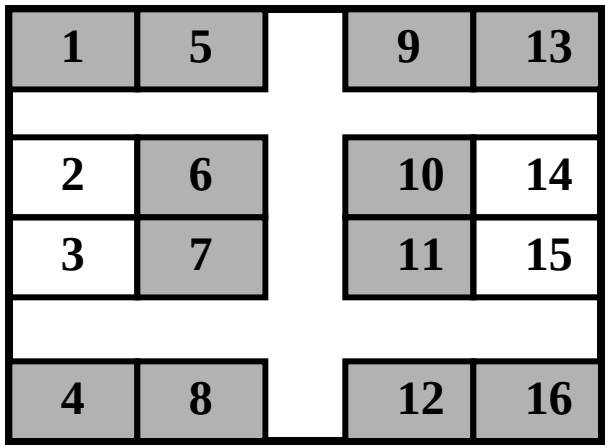
(c) Yaw = 0.02



Composite

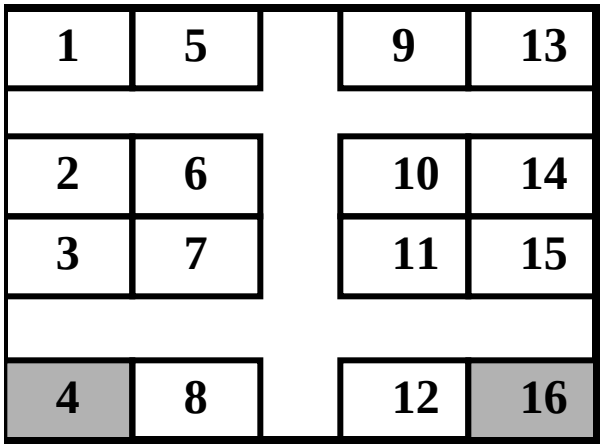


Composite from symmetry

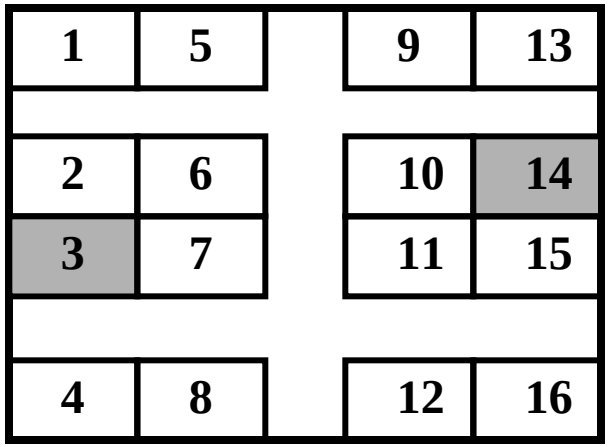


Two Actuator Placement with Uncoupled Moments (13 Generations)

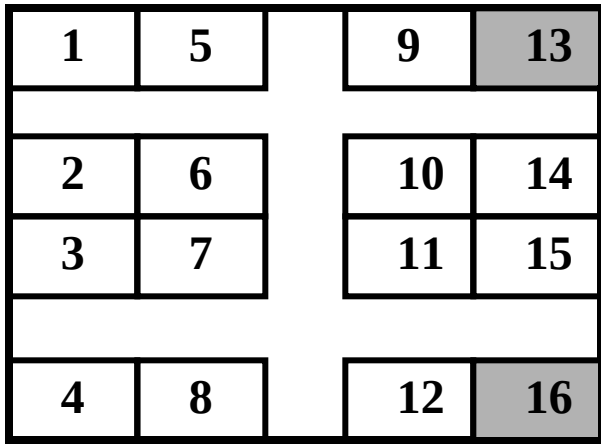
(a) Pitch = 0.01



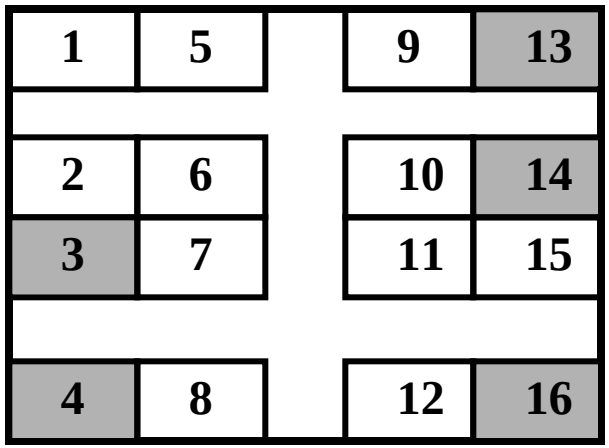
(b) Roll = -.016



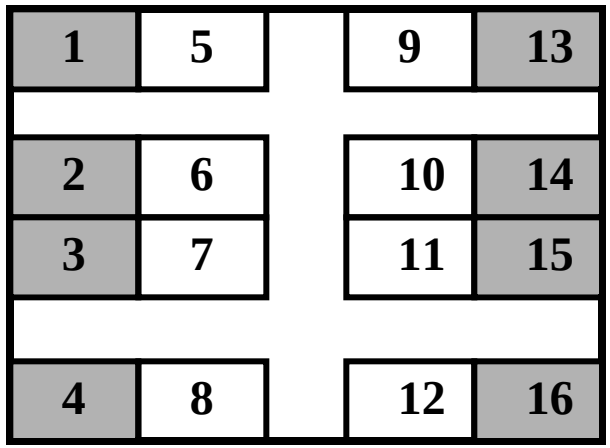
(c) Yaw = 0.02



Composite



Composite with symmetry



Timing Results

Each analysis takes one minute.
GA has 300 analyses per generation with 13 generations.

Actuators	Combinations	Analysis Time	GA Time
16	65,536	~1,100 hours	65 hours
34	1,717,986,918	~286M hours	
100	Do not even think about it!		

Concluding Remarks

- A problem that appeared to be unsolvable using traditional methods, now looks promising with the application of a GA

- GA approach saves computation time for a small number of actuators

- GA approach is portable to larger problems and can handle a large number of discrete variables with few changes to the code

- GA approach is portable to parallel computers

- GA approach can take advantage of problem symmetry