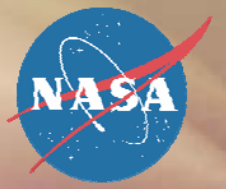
National Aeronautics and Space Administration



Genetic Algorithm Optimization of Inlet Geometry for a Hypersonic Jet Engine with Mode Transition

Ashley Micks
NASA Academy, NASA Glenn Research Center
Massachusetts Institute of Technology

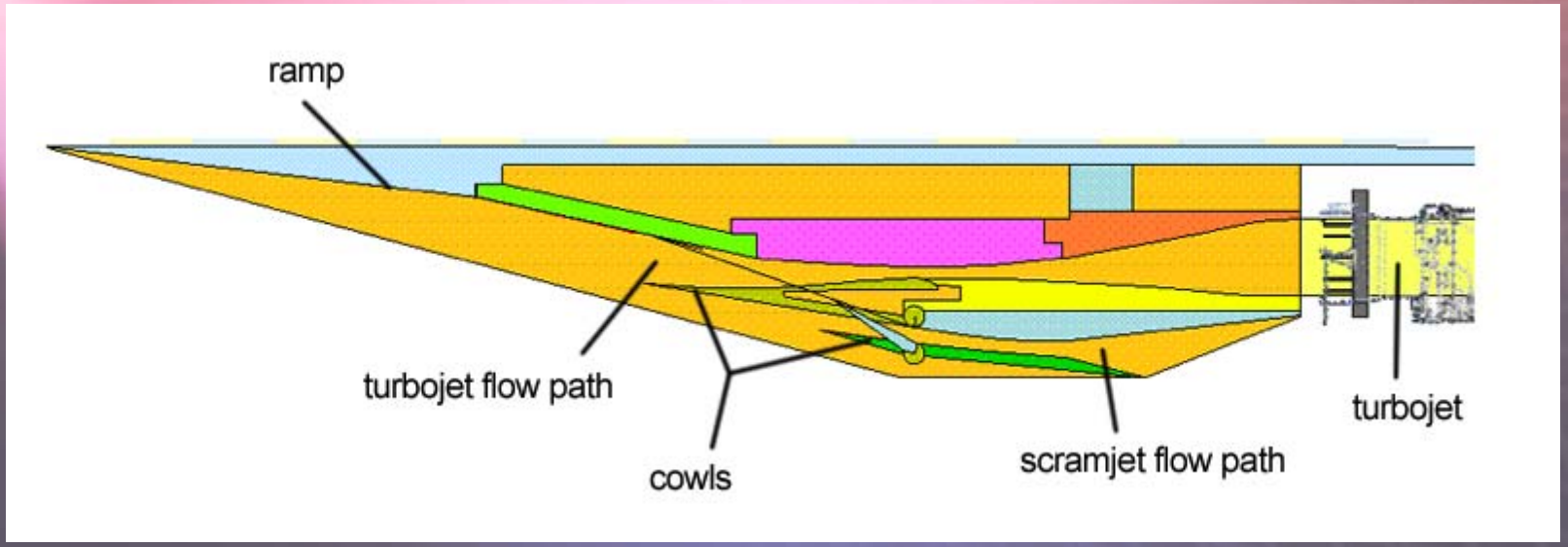
August 10, 2007

Introduction

- **The role of computational fluid dynamics (CFD) and optimization through CFD**
  - **Simulation data helps to focus experiments.**
  - **Reduce wind tunnel expenses**
- **The promise of genetic algorithm optimization**
  - **“natural selection” in a “population” of designs**
  - **Selection takes care of design process → little knowledge required**
  - **Multiple designs in a population → parallel computing**

Introduction

- **Baseline design for this GA optimization**
  - **The Large-scale Inlet Mode Transition (L/IMX) inlet**
  - **The need for transitional inlets on high-Mach aircraft**

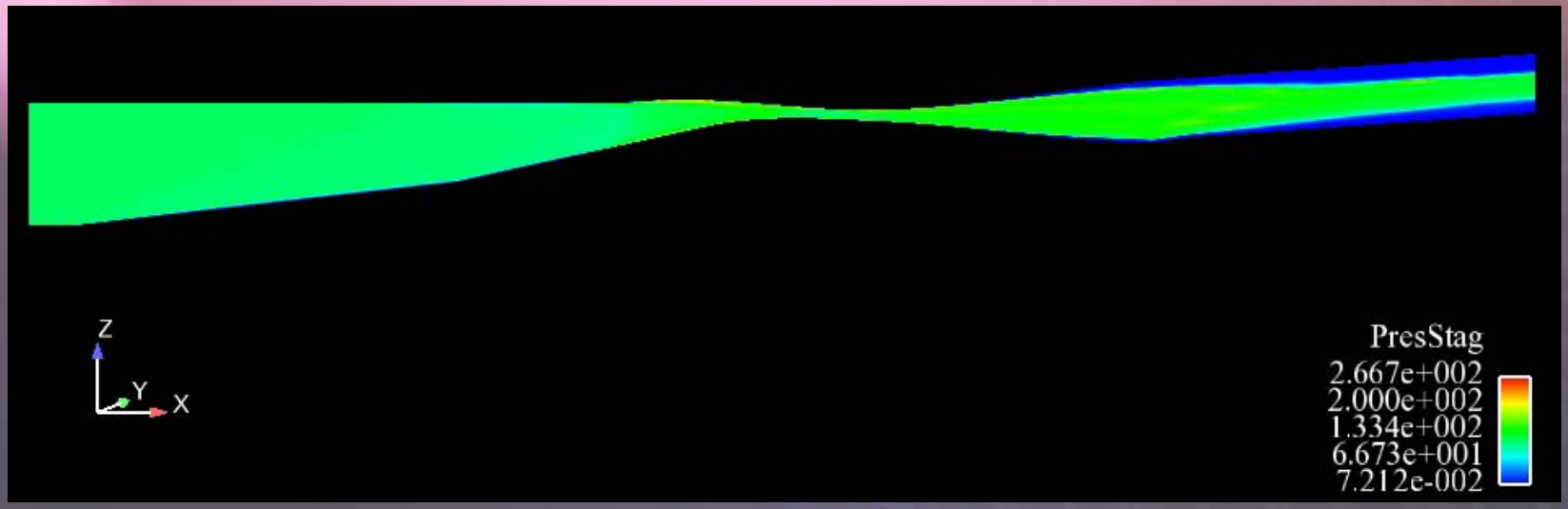


Problem Definition

- **Optimize inlet's contribution to the efficiency of the engine**
- **Significance of total pressure for efficiency**
  - **a.k.a stagnation pressure**
  - **Decreases with entropy generation**
  - **Reflects flow's ability to do work**
  - **Total pressure recovery = $p_{t,\ exit}$ / $p_{t,\ entrance}$**
- **Concerns:**
  - **The shock train → shock losses**
  - **Boundary layers → entropy, more shock loss**

Objective

- **“Implement a GA to design an optimum geometry for the turbojet flow path of a hypersonic jet engine inlet at Mach 4, using the L/IMX inlet as a baseline.”**
- **Mach 4 = cutoff for turbojet**
- **Optimum = highest possible total pressure recovery**
- **Non-uniform flow at exit → use average total pressure recovery over cross-section**

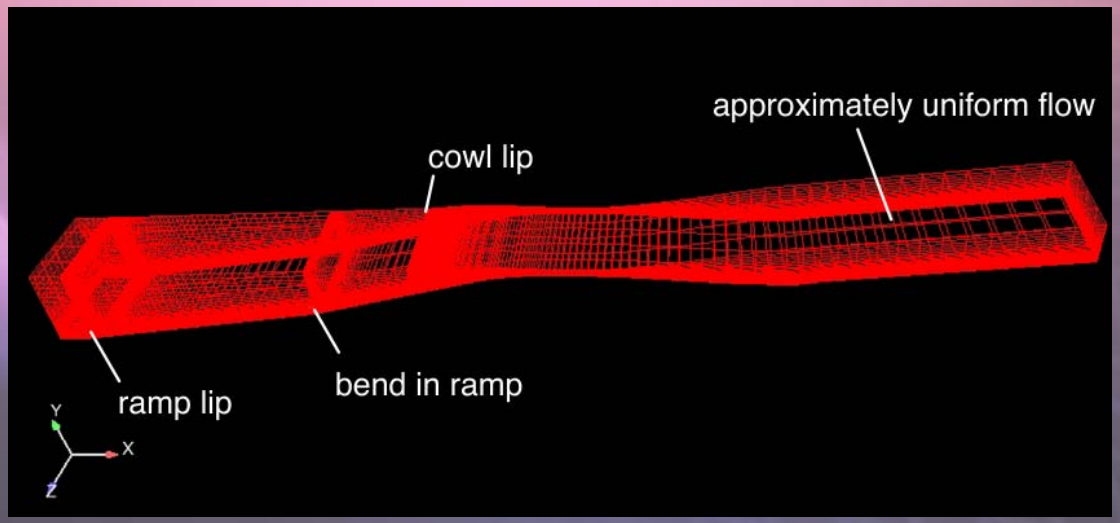


Approach

- **Define and implement key components of the GA**
  - **Alleles**
  - **Chromosomes**
  - **Fitness**
  - **Initialization**
  - **Parent selection**
  - **Crossover**
  - **Mutation**
  - **Elitism**
  - **Design constraints**
- **Validate the GA through tests**
- **Lay out how the GA interacts with auxiliary programs, data files, and scripts**

Approach: Alleles

- **Definition: design parameter values that define an individual**
- **25 total, 3 types:**
  - **Coordinates of key points along the inlet**
  - **Bias values: which way does the wall bend?**
  - **Tightness values: how sharply does it bend?**



Approach: Chromosomes

- **Definition: a list of an individual's alleles**
- **Alleles are listen in order of type:**
  - **X-coordinate of cowl lip**
  - **Y-coordinates of all key points (x fixed)**
  - **Bias values**
  - **Tightness values**

Approach: Initialization

- **Definition: the generation of an initial population**
- **Initial population generator (IPG) separate from GA**
- **Individuals distributed randomly in design space**
  - **Uniform probability for all possibilities, each allele**
- **L/IMX design always included**

Approach: Fitness

- **Definition: the cross-sectional average total pressure recovery**
- **CFD output (density, velocity) → total pressure**
- **CFD output is nondimensional**
- **Accommodating non-uniform exit flow**
  - **CFD results only at discrete grid points**
  - **Riemann sum, divide by area → $p_{t,\ exit}$, $p_{t,\ entrance}$**
  - **Total pressure recovery = $p_{t,\ exit}$ / $p_{t,\ entrance}$**

Approach: Parent Selection

- **Definition: selection of individuals to pass on their alleles to the next generation**
- **Check: 2 unique parents for crossover**
- **Higher fitness → more likely to be chosen**
- **Roulette Wheel**
  - **Individual's "wedge on the wheel" proportional to fitness**
  - **"roulette ball" stops at a random point around the wheel**

Approach: Crossover

- **Definition: 2 parent chromosomes swap or blend alleles to form 2 children**
- **Occurs with 70% probability**
- **If crossover, then for each allele, numbered 1-25:**
  - **1/3 chance alleles remain unchanged**
  - **1/3 chance alleles swapped between parents**
  - **1/3 chance blended crossover**
- **Blended crossover**
  - **Children's alleles = randomly weighted averages of parents**

$$a_{child1} = \gamma a_{parent1} + (1-\gamma)a_{parent2}$$

$$a_{child2} = \gamma a_{parent2} + (1-\gamma)a_{parent1}$$

Approach: Mutation

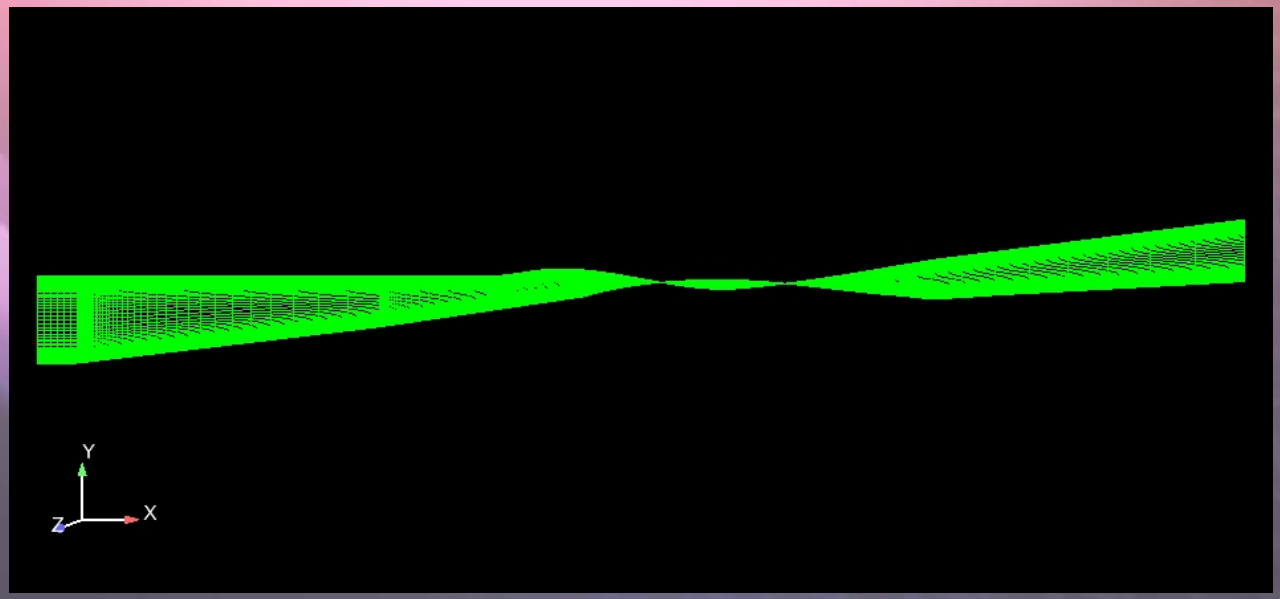
- **Definition: a random allele changes its value**
- **Occurs for every new individual except the one copied through elitism**
- **One allele chosen at random per individual**
- **Allele value changes to any value in design space**
- **Uniform probability of all possibilities**

Approach: Elitism

- **Definition: automatically copy the individual with the highest fitness is into the nest generation**
- **The only exception to crossover and mutation**
- **The best fitness cannot decrease over generations.**
- **Speeds convergence**
  - **Favors high fitness**
  - **May converge too soon for small populations**

Approach: Design Constraints

- **The inlet walls cannot intersect**
- **The cowl lip must be upstream of the shock off the ramp**
- **Retry generating the individual if it violates a constraint**

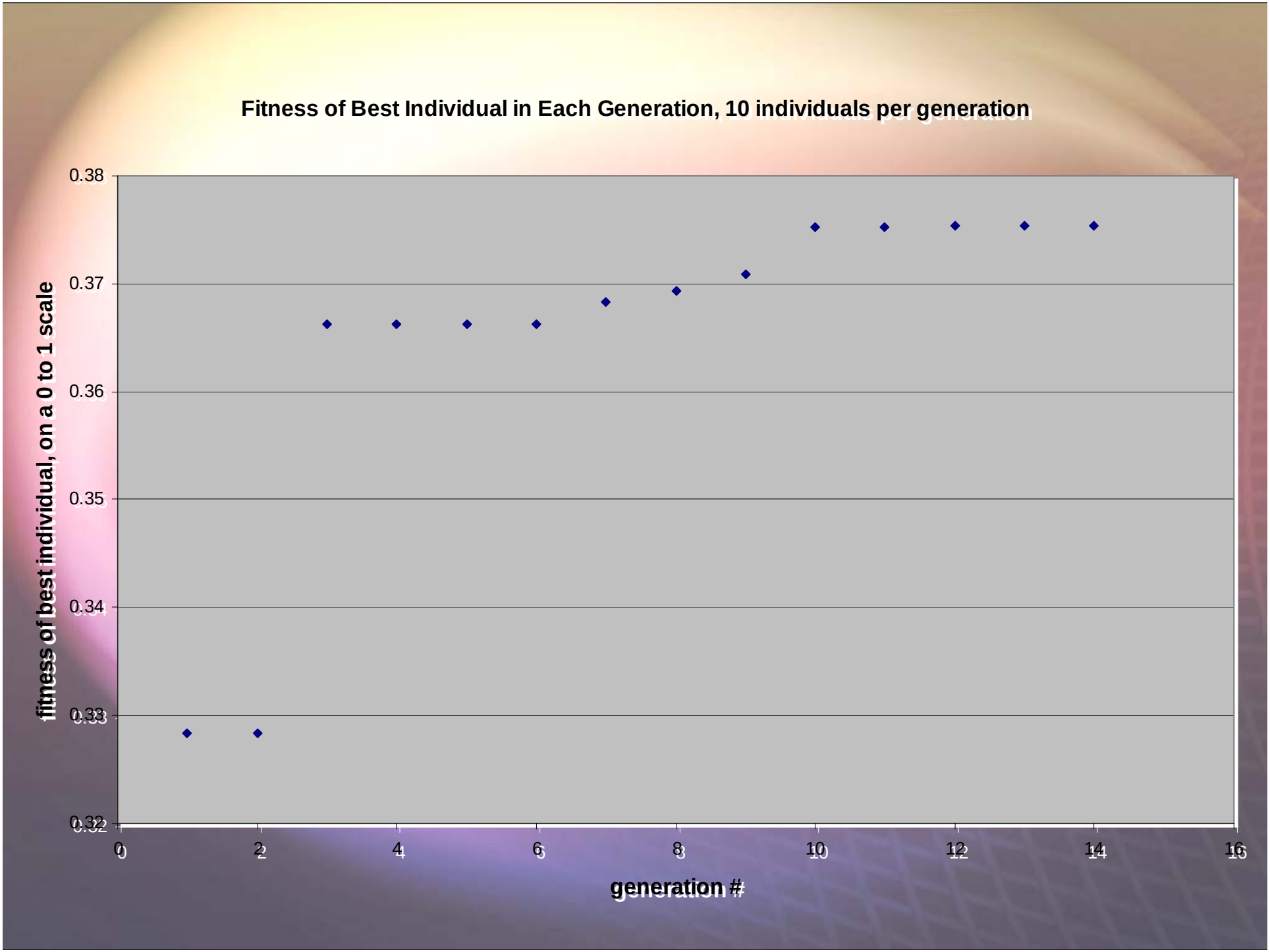


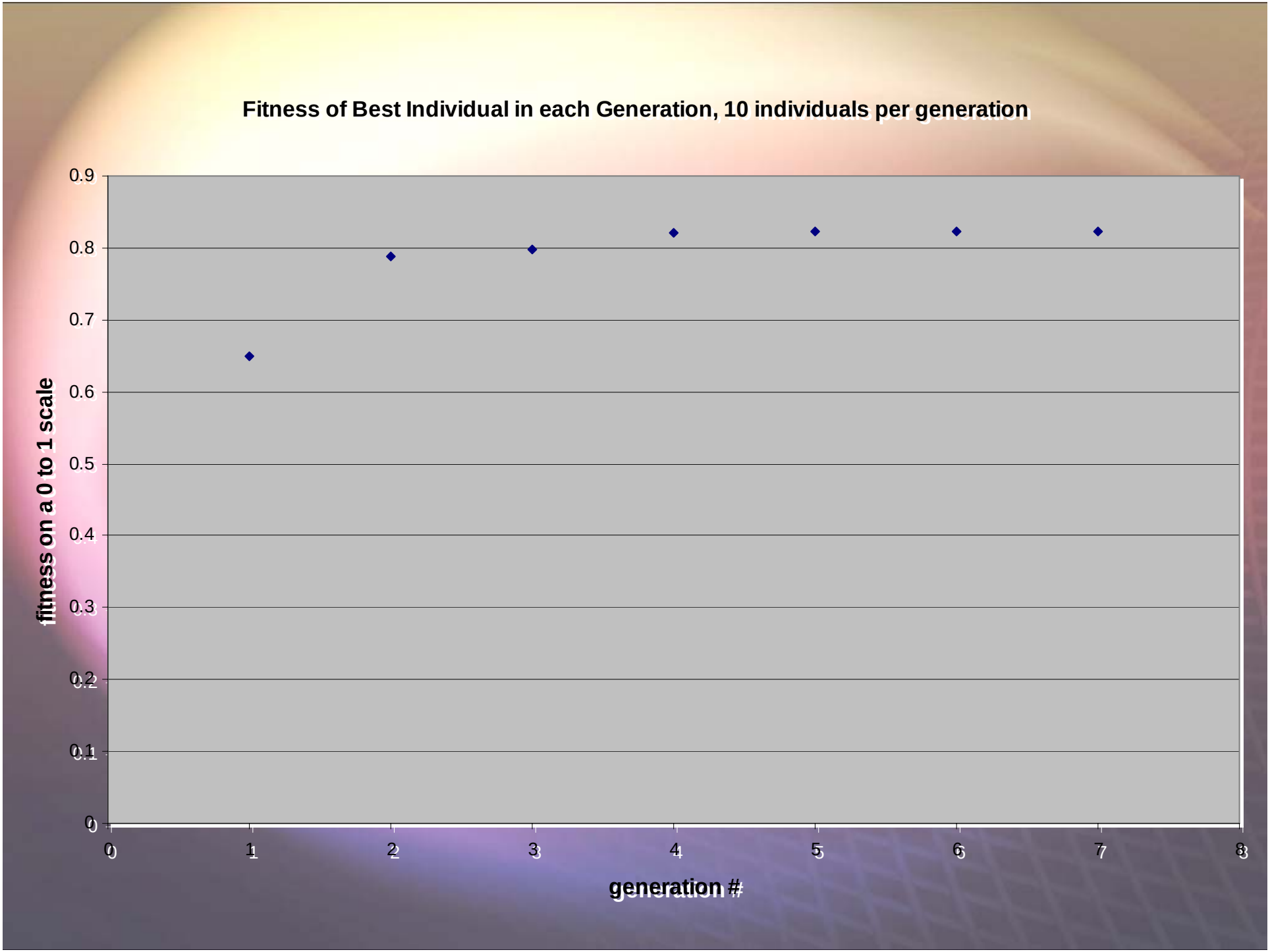
Approach: Validation

- **Use dummy fitness function to test GA performance**
- **Dummy fitness function:**

$$f(x) = \left| \cos((x-400)/25) * e^{-|(x-400)/400|} \right|$$

- **x = sum of allele values**
- **Resulting changes for improvement:**
  - **Don’t have all designs on one side of optimum**
  - **Give initial population uniform probability across design space**
  - **Give mutation uniform probability across design space**
  - **Increase mutation to 100% probability of occurrence**


Fitness of Best Individual in Each Generation, 10 individuals per generation
fitness of best individual, on a 0 to 1 scale
0.38
0.37
0.36
0.35
0.34
0.33
0.32
0
2
4
6
8
10
12
14
16
generation #


Fitness of Best Individual in each Generation, 10 individuals per generation
fitness on a 0 to 1 scale
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0
0
1
2
3
4
5
6
7
8
generation #

Approach: Validation

- **Use L/IMX inlet to test actual fitness calculations**
- **After debugging, total pressure recovery = 0.4377**
- **Resulting changes for improvement:**
  - **gridline distribution function from grid generator -> average by area**
  - **Read from binary result file, not ASCII**

Approach: Analysis Tools

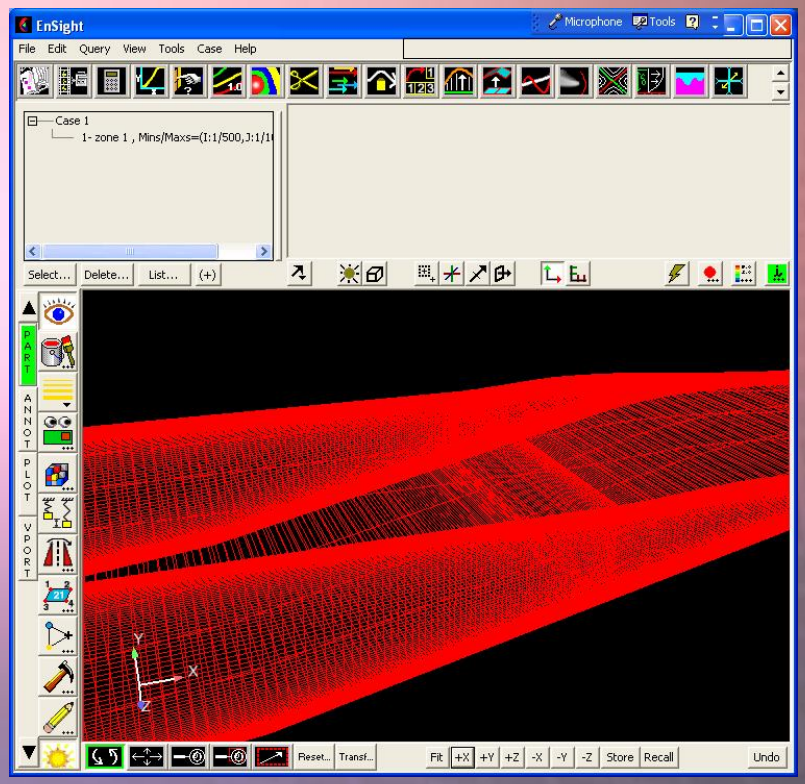
- **Overflow**
  - **CFD software, calculates flow behavior**
  - **Input: Plot3D grid, boundary conditions, “restart” options**
  - **Output: density, velocity vectors at gridline intersections**
  - **Boundary conditions: inviscid approaching inlet, viscid inlet walls**

Approach: Analysis Tools

- **Grid generation**
  - **GA outputs 2D grid file**
  - **“makegrid” script converts to nominally 3D file for Overflow**
  - **Grid concentration near walls, ramp corners, cowl lip**

Approach: Analysis Tools

- **EnSight**
  - **Visualizes grid files, Overflow result files**
  - **Color codes according to flow properties (temperature, Mach, pressure, etc.)**



Approach: Program Interaction

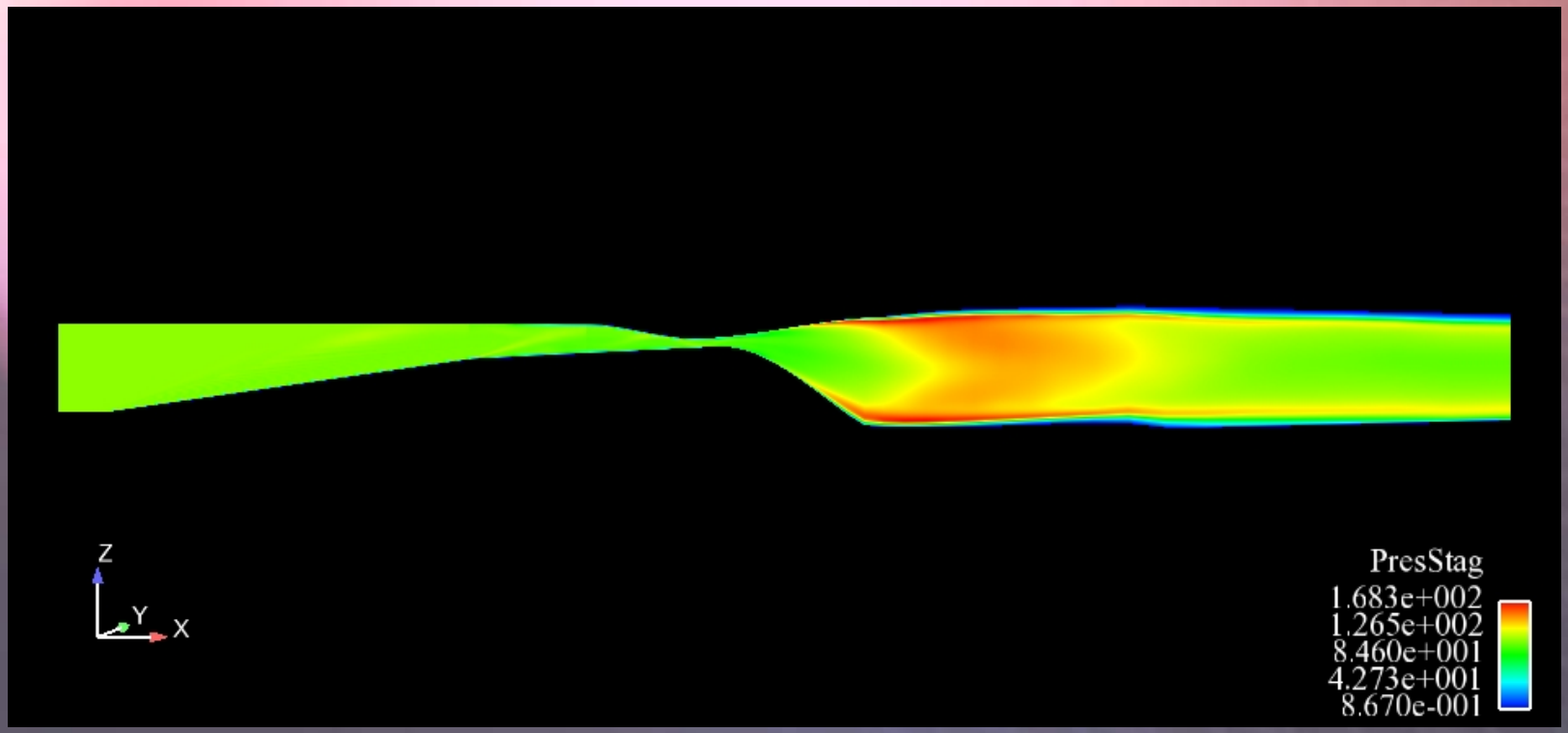
- **Programs and files across 2 machines**
  - **Columbia, NASA supercomputer at Ames**
  - **Lou, NASA mass storage machine**
- **Batch script calls:**
  - **IPG to begin**
  - **GA for each generation**
  - **moves files as needed.**
- **Columbia stores executables**
  - **GA**
  - **IPG**
  - **Overflow**
  - **Scripts**
- **Lou stores archives**
  - **grid files**
  - **Chromosomes**
  - **Fitnesses**
  - **best scores**

Results

- **Functional Initial Population Generator (IPG)**
  - **Well distributed across design space**
  - **Avoids constraint violations**
- **Functional fitness calculation**
- **Functional random number generator**
  - **Does not repeat between runs of the program**
  - **Does not repeat when called repeatedly in program**
- **Functional GA**
  - **Fitness improves from baseline and never backslides**
  - **Performance improves with larger populations, but computing time limited for this project**
  - **Avoids constraint violations**
  - **Well-tested grid generation**

Results

- **Improved design from small-scale run**
  - **Total pressure recovery of baseline = 0.4377**
  - **Improved total pressure recovery = 0.4450**



Future Work

- **Improved compatibility with Columbia**
- **Run large-population, many-generation cases**
  - **Possibly multiple days each to run on Columbia**
  - **Generations of >100 individuals**
  - **Thoroughly explore design space**
- **Tweak GA to improve efficiency**
  - **Alternate parent selection methods**
  - **Crossover, mutation probabilities**
  - **Alternate crossover, mutation setup**
- **Tweak constraints, coordinating with L/IMX**

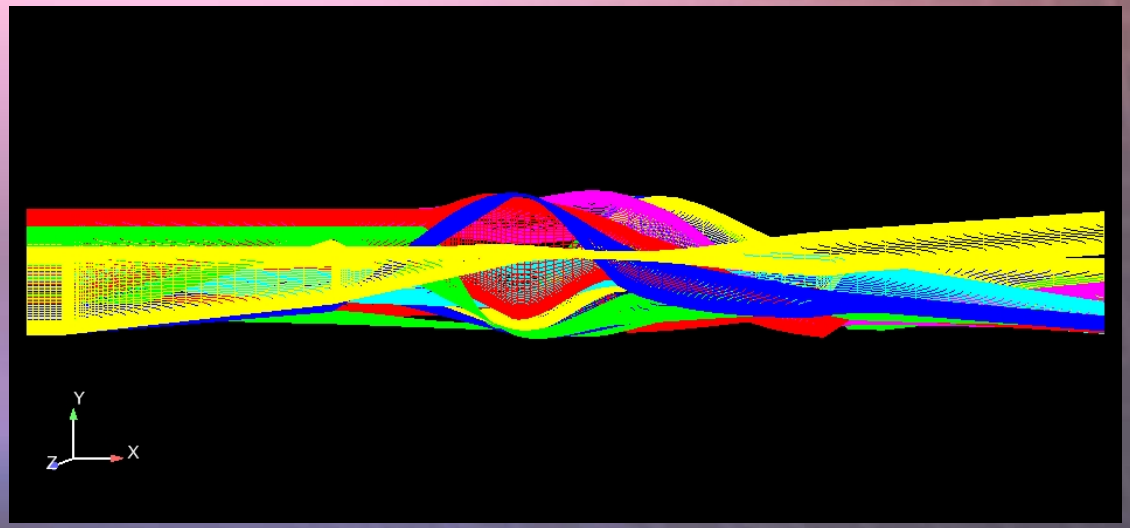
Conclusions

Pros of GA Approach

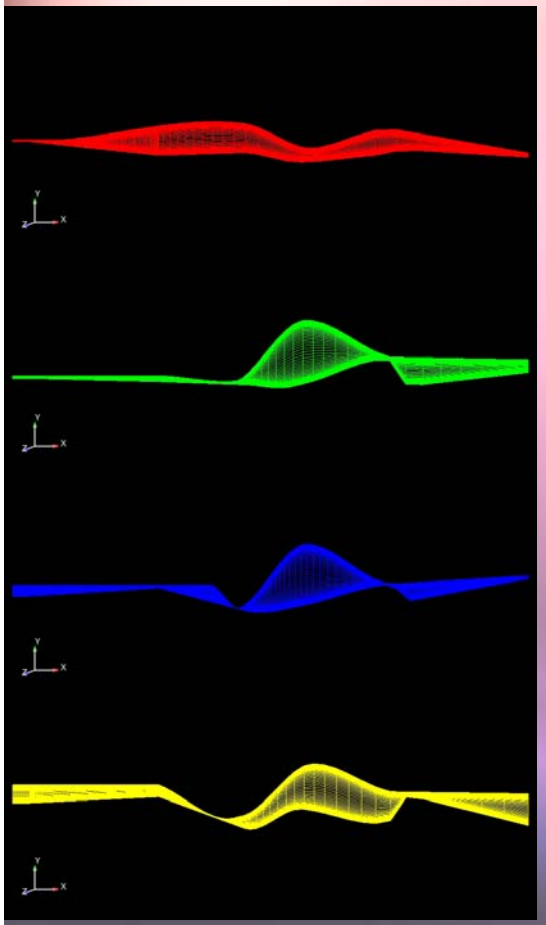
- GAs explore parts of design space far from baseline
- Multiple designs to test → Parallel (efficient) computing
- adaptable

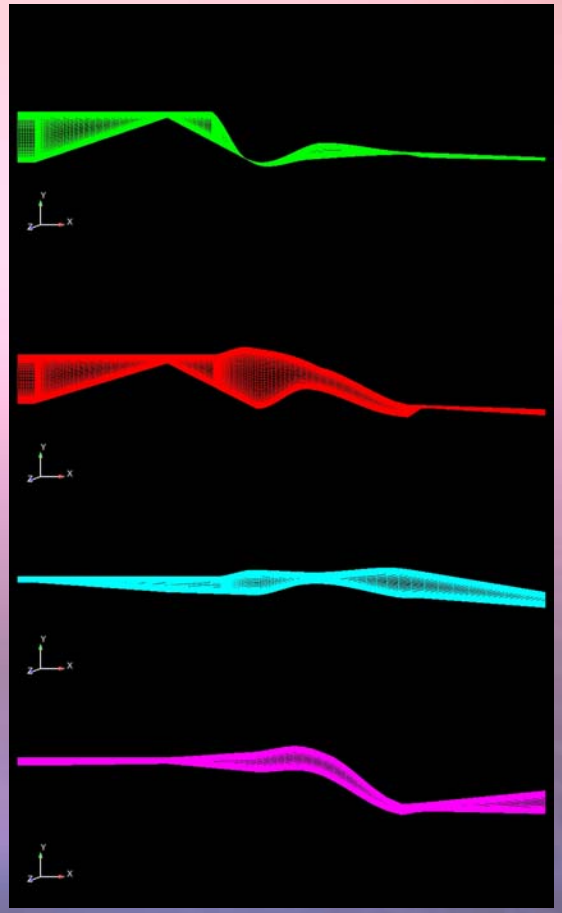
Cons of GA Approach

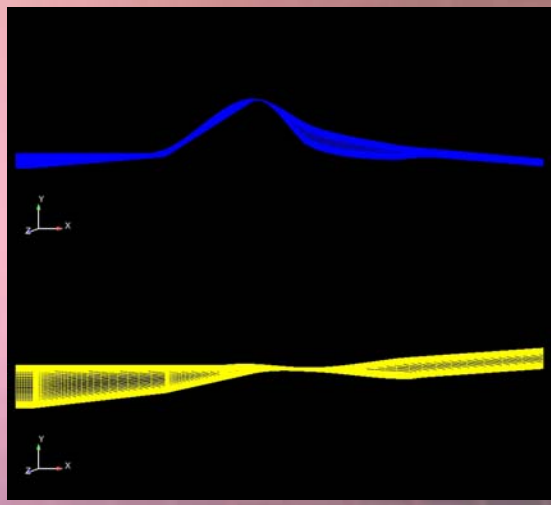
- Requires great (expensive) computational power
- Requires extensive computing time



Conclusions







Acknowledgements

Special thanks to Meng-Sing Liou, May-Fun Liou, Dave Saunders, MaryJo Long-Davis, Jay Horowitz, and Mary Vickerman for their mentorship and support. Thanks also to the Massachusetts Space Grant Consortium for making this work with NASA possible, to Dr. David Kankam for directing the NASA Glenn Academy, and to Kyle Gaiser for his friendship and collaboration.

References

[1]Bourke, Paul, "Interpolation Methods," URL: http://local.wasp.uwa.edu.au/~pbourke/other/interpolation/ [cited 16 July 2007].

[2]Davis, Lawrence, *Handbook of Genetic Algorithms*, Van Nostrand Reinhold, New York, 1991.

[3]Goldberg, David E., *Genetic Algorithms in Search, Optimization, and Machine Learning*, Addison-Wesley Publishing Company, Inc., Reading, Massachusetts, 1989.

[4]Oyama, Akira, and Liou, Meng-Sing, "Multiobjective Optimization of Rocket Engine Pumps Using Evolutionary Algorithm," AIAA 2001-2581, 2001.

[5]Walatka, Pamela P., Buning, Pieter G., Pierce, Larry, and Elson, Patricia A., *PLOT3D User's Manual*, NASA Technical Memorandum 101067, 1990.

[6]Whitlock, Sarah T., Boman, Erik, and Terry, Steven H., "Fortran Tutorial," URL: http://gershwin.ens.fr/vdaniel/Doc-Locale/Langages-Program-Scientific/Fortran/Tutorial/index.htm [cited 14 June 2007].