Advances in Parameter Estimation Techniques Applied to Flexible Structures

$P.95$

Egbert Maben
David C. Zimmerman
University of Florida
231 Aerospace Building
## Gainesville, Florida 32611

OVERVIEW

- **Distributed Parameter Modeling – PDEMOD**
  - **Wittrick–Williams Root Solving**
  - **Time Simulation Capability**

- **Parameter Estimation Formulation**
  - **Modal Parameter**
  - **Time–Domain Data**

- **Parameter Estimation Techniques**
  - **Lee and Hossain Approach**
  - **Modified Steepest Descent**
  - **Genetic Algorithm Approach**

- **Examples – NASA Mini–Mast**

- **Concluding Remarks**

Dynamic Systems and Control Laboratory
University of Florida

In this work, various parameter estimation techniques are investigated in the context of structural system identification utilizing distributed parameter models and "measured" time-domain data. Distributed parameter models are formulated using the PDEMOD software developed by Taylor [1]. Enhancements made to PDEMOD for this work include (i) a Wittrick–Williams based root solving algorithm [2], (ii) a time simulation capability, and (iii) various parameter estimation algorithms. The parameter estimations schemes will be contrasted using the NASA Mini–Mast as the focus structure.

PDEMOD - Partial Differential
Equation MODeling

- Generic Configuration
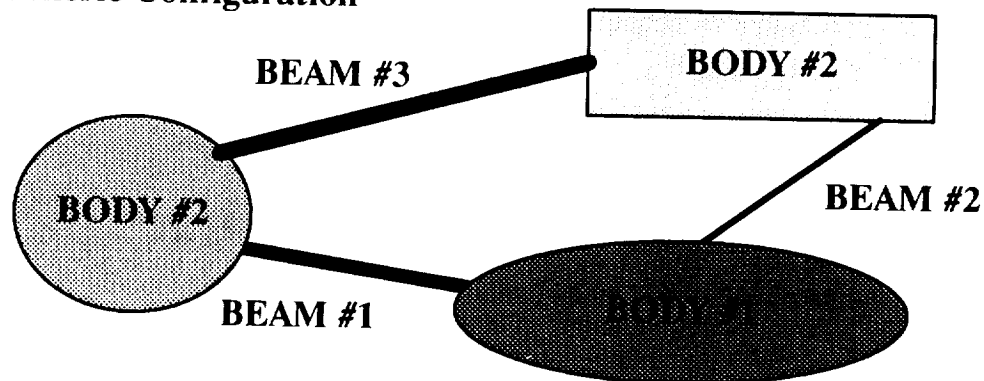
- 3 - Dimensional Spacecraft
- Rigid Bodies (Full Inertia Matrices)
- Flexible Beams
    - Bending - Euler-Bernoulli Equations
    - Elongation, Torsion - String Equation

- Force, Moment and Constraint Equations Yield
Characteristic Equation

$$\left| A(\omega) \right| = 0$$

- Dimension of A = 12*Number of Beams

Partial Differential Equation Modelling (PDEMOD-Release 1) is capable of modeling complex flexible spacecraft which consist of a three-dimensional network of flexible beams and rigid bodies. Each beam has bending (Euler-Bernoulli) in two directions, torsion, and elongation degrees of freedom. The rigid bodies can be attached to the beam ends at any angle or body location. The eigenvalues are determined by numerically solving for the values of frequencies which cause the determinant of a frequency-dependent matrix to become zero. Eigenfunctions can then be calculated in closed-form at a finite number of specified points.

EXTENSIONS TO PDEMOD

◉ **Wittrick-Williams Root Solving Capability**

   ◉ **Determines Number of E-values in Given Frequency Range**

   ◉ **Can Indicate Presence of Repeated Roots**

   ◉ **Reduces Computational Burden When Used in Conjunction with Root-Solving Technique**

◉ **Time-Domain Simulation Capabilities**

   ◉ **Outputs at Discrete Points Calculated From Modal Model**

   ◉ **"Modal Initial Conditions" Determined From Initial Conditions Using Finely Discretized Eigenvectors**

   ◉ **Closed-Form Modal Time Solutions Utilized**

In addition to the time-domain based parameter estimation techniques, two enhancements to PDEMOD-1 have been made. The first enhancement is a Wittrick-Williams based root-solving enhancement to the bisection root-solving algorithm. Provided lower and upper frequency limits, the Wittrick-Williams algorithm provides the number of frequencies that exist between the two limits. This information, when used properly, can greatly reduce the computational burden of solving for the eigenvalues of the structure. The second enhancement is the addition of time-simulation capabilities. Sensors can be placed at arbitrary points on the structure. A finite-dimensional (user specified) modal model is then created. Physical initial conditions are transformed to modal initial conditions using the eigenfunctions and mass distribution evaluated at a number of discrete points. It should be noted that for accurate results, the number of discrete points must be chosen relatively "large". The modal time responses are then calculated in closed-form. The resulting physical time response at the sensor locations can then be calculated.

218

PARAMETER ESTIMATION FORMULATION

- **Frequency Domain Formulation**

$$\min J = \sum_{i=0}^{n_{meas}} W_i \underbrace{\left(\omega_{exp}^i - \omega_{ana}^i\right)^2}_{\substack{\text{Frequency} \\ \text{Satisfaction}}} + H_i \underbrace{\| \underline{u}_{exp}^i - \underline{u}_{ana}^i \|_f}_{\substack{\text{Eigenfunction} \\ \text{Satisfaction}}}$$

  wrt physical design parameters

- **Many Sensitivity Formulations Omit Eigenfunction Satisfaction**

- **Weights W and H Allow For Engineering Judgement on Confidence of Measurements**

- **Time–Domain Formulation – This Study**

$$\min J = \sum_{i=0}^{n_{sensor}} W_i \int_{t_0}^{t_f} \left(y_{exp}^i(t) - y_{ana}^i(t)\right)^2 dt$$

  wrt physical design parameters

Dynamic Systems and Control Laboratory
University of Florida

Many sensitivity–based (and other) parameter estimation techniques are driven by the mismatch in analytical and "measured" modal properties. The parameter estimation problem is then to adjust the physical parameters of the system such that there is an improved match between measured and analytical modal properties, often times subject to various constraints. An alternate formulation is to work directly with time–history measurements and analytical predictions. This is the approach investigated in this paper.

PARAMETER ESTIMATION TECHNIQUES

- **Lee & Hossain – Time Domain**

$$q^{k+1} = q^k - S^k \frac{\partial J}{\partial q}$$

q^k – Vector of Design Parameters

S^k – Step Size Matrix (???)

$\frac{\partial J}{\partial q}$ – Gradient

- **Experience – Difficulty In Selecting Step Size Matrix**
  (Lee & Hossain Provide No Insight Into Selection)

- **Motivated by CHORDS (SDRC), Vary Only One Variable With Corresponding Highest Gradient Value**

  - **Uses 1–D Line Search on Single Physical Variable**

  - **Disadvantage: Loses Significant Gradient Information**

- **Genetic Algorithms**

Three parameter estimation techniques are investigated in this paper. The first is that proposed by Lee & Hossain [3]. In this work, the parameter of physical properties, q, are modified based on gradient information. There was no discussion in Ref. [3] on how to select the step–size matrix, S^k. Improper choice of S^k was found to lead to divergence of the solution (S^k to large), or in minimal improvement (S^k to small). Motivated from an optimization technique utilized in the CHORDS software program, a simplified one–dimension line search was investigated. In this approach, the variable with the highest sensitivity is chosen to be varied, with all others held constant. The optimal step–size of the one–dimensional search was calculated using a quadratic approximation. This approximation required an additional function evaluation. Finally, a Genetic Algorithm [4,5] approach was investigated.
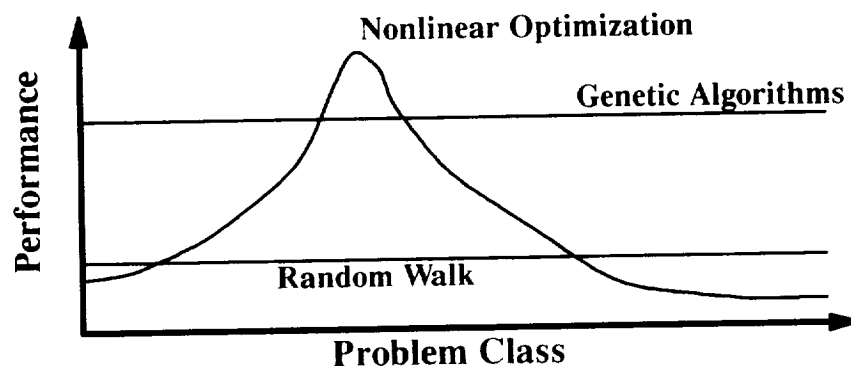
GENETIC ALGORITHMS

- **Based on Darwin's "Survival of the Fittest" Theories**

- **Shows Great Potential for**
  - **Multi–Modal Objective Functions**
  - **Discrete and/or Continuous Design Variables**
  - **Discontinuous Design Space**

- **Works With a Coding of the Design Variables,**
  Not the Design Variables Themselves

- **Searches From a Population of Designs,**
  Not a Single Design Point

- **Uses Payoff (Objective Function) Information,**
  Not Gradient Information

- **Uses Probabilistic Transition Rules,**
  Not Deterministic Rules

Genetic algorithms (GA's), as introduced by Holland [4], are one form of directed random search. The form of direction is based on Darwin's "survival of the fittest" theories. GA's are radically different from the more traditional design optimization techniques. GA's work with a coding of the design variables, as opposed to working with the design variables directly. The search is conducted from a population of designs (i.e., from a large number of points in the design space), unlike the traditional algorithms which search from a single design point. The GA requires only objective function information, as opposed to gradient or other auxiliary information. Finally, the GA is based on probabilistic transition rules, as opposed to deterministic rules. These features allow the GA to attack problems with local–global minima, discontinuous design spaces and mixed variable problems, all in a single, consistent framework.

221

GA's - ALGORITHM OVERVIEW

- Initial Population of Designs Created – Random or Heuristic

- Initial Population Allowed to *"Evolve"* Over Generations

- Conjecture – Evolution is the Best Compromise Between Determinism and Chance

- Motivation – GA's are Robust Over a Wide Range of Problems

Performance vs *Problem Class* — curves labeled Nonlinear Optimization, Genetic Algorithms, Random Walk

In GA's, a finite number of candidate solutions or designs are randomly or heuristically generated to create an initial population of designs. This initial population is then allowed to evolve over generations to produce new, and hopefully better designs. The basic conjecture behind GA's is that evolution is the best compromise between determinism and chance. The basic motivation behind the development of GA's is that they are robust problem solvers for a wide class of problems. However, it should be noted that they are not as efficient as nonlinear optimization techniques over the class of problems which are ideally suited for nonlinear optimization; namely continuous design variables with a continuous differentiable unimodal design space.
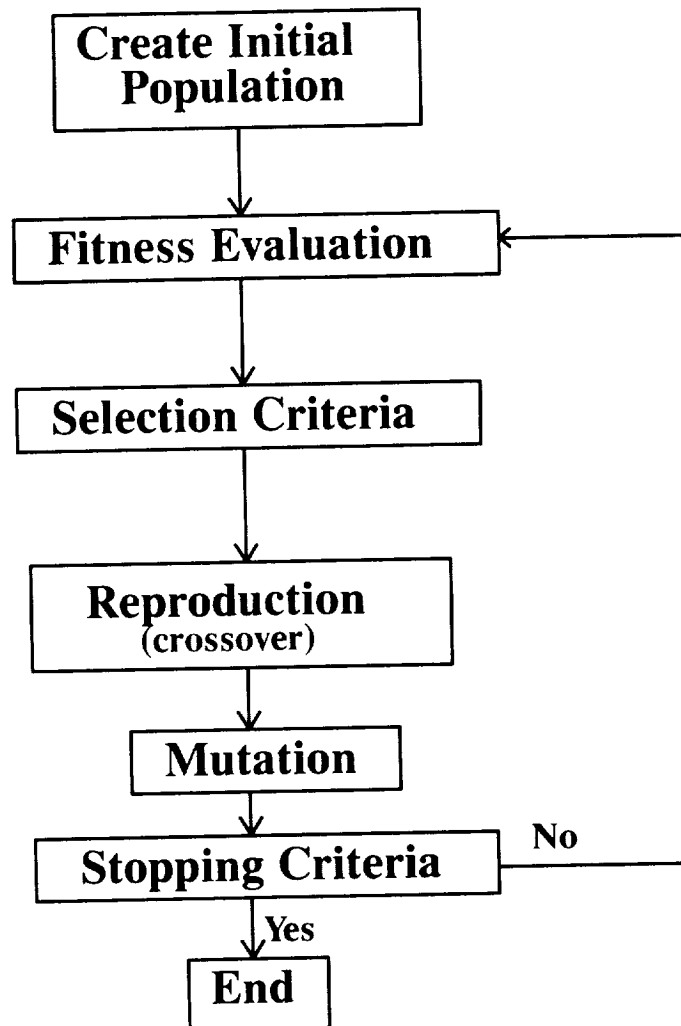
222

GENETIC ALGORITHM MODULES

⊛ **Design Variables *Coded* as a q-Bit Binary Number**
  - ⊛ **Continuous Variables Like A/D Converter**
  - ⊛ **Discrete Variables Have Unique Binary Strings**
  - ⊛ **A Population Member is Just a String of Design Variables**

⊛ **GA *Evaluation* – Level of *Fitness* Assigned to Each Member**
  - ⊛ **Fitness Chosen to be Related to Objective Function**
  - ⊛ **GA's Maximize Fitness**

⊛ **GA *Selection* – Determination of Which Individuals in Current Population Chosen to be Parents**
  - ⊛ **Biased Towards More Fit Members**
  - ⊛ **Proportional Bias –** $p\binom{member}{i} = \dfrac{fitness_i}{\displaystyle\sum_{j=1}^{npop} fitness_j}$

⊛ **GA *Crossover* – Transfer of Design Information From Parents to Prodigy**

⊛ **GA *Mutation* – Low Probability Random Switch of Bits**
  - ⊛ **Retains Design Information Over Entire Design Space**
  - ⊛ **Aids Search For Global Optimal Solution**

Each design variable is *coded* as a q-bit binary number. A continuous design variable is approximated by 2^q discrete numbers between lower and upper bounds set for the design variable. Discrete variables would each be assigned a unique binary string. A population member is obtained by concatenating all design variables to obtain a single string of ones and zeros. *Evaluation* is the process of assigning a fitness measure to each member of the current population. Because GA's attempt to maximize the fitness of each member, an objective function which is to be minimized must be converted into an equivalent maximization problem. *Selection* is biased towards the most fit members of the population. Therefore, designs which are better as viewed from the fitness function, and therefore the objective function, are more likely to be chosen as parents. *Crossover* is the process in which design information is transferred to the prodigy from the parents. Many crossover operators (1-point, 2-point, uniform) have been investigated. *Mutation* is a low probability random operation which may perturb the design represented by the prodigy. The operator works on a bit-by-bit basis and is governed by the probability of mutation, p_m. At each bit, a biased coin toss is used to determine whether the bit should be logically "NOTed". The mutation operator is used to retain design information over the entire domain of the design space during the evolutionary process.

223

GENETIC ALGORITHMS - SUMMARY

Create Initial Population

↓

Fitness Evaluation

↓

Selection Criteria

↓

Reproduction (crossover)

↓

Mutation

↓

Stopping Criteria → No

↓ Yes

End

Dynamic Systems and Control Laboratory
University of Florida

In the implementation of the GA shown above, the prodigies are produced until the number of prodigies created is equal to n_{pop}, the population size. At that point, the current population of parents are discarded and the prodigies are in turn made parents which are capable of producing the next generation of prodigies. Thus, the production of n_{pop} prodigies can be viewed as the completion of one generation cycle in the evolutionary process. During this procedure, it is possible that both the fitness of the most fit member and the average population fitness can be temporarily reduced during the evolutionary process. To overcome this, the concept of a steady-state GA was implemented. In a steady-state GA (SSGA), the fitness of the children after they have been mutated is evaluated. These fitness values are then compared to the fitness of the two least fit parents in the current population. If the mutated child's fitness is higher than the least fit member in the population, the child will replace that member and will instantly become a candidate parent. To keep intact the concept of a generation, a generation is defined to be complete when the number of children produced, but not necessarily accepted into the population, is equal to n_{pop}.

EXAMPLE - NASA MINI-MAST

BAY #
18
16
14
12
10
8
6
4
2
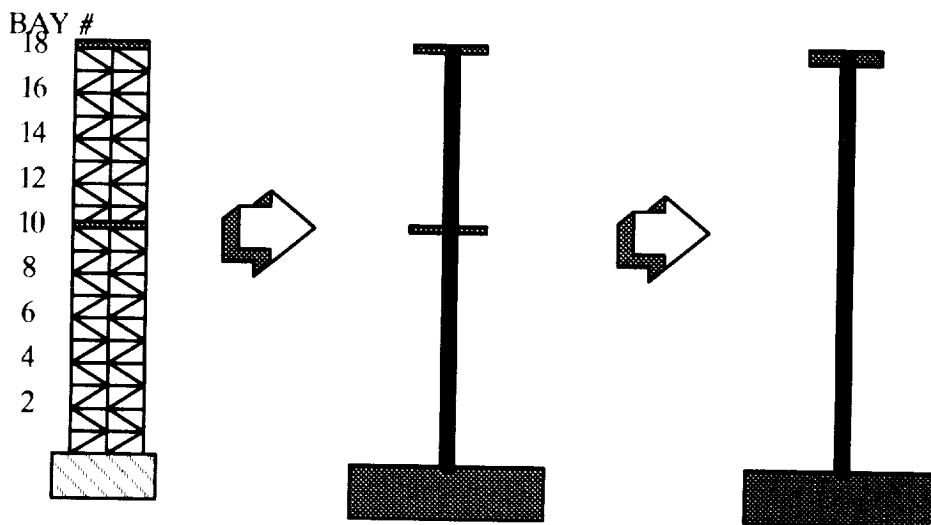
- Mini–Mast Modeled With One or Two Beam PDEMOD

- Single Sensor at Tip (Bay 18)

- Initial Conditions:

$$y(x) = \frac{0.6}{66.24^3}x^3 - \frac{0.4}{66.24}x \quad ft$$

$$v(x) = 0 \ ft/sec$$

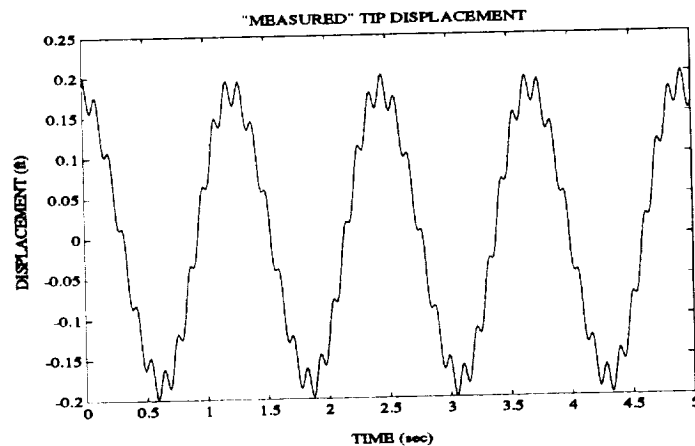Dynamic Systems and Control Laboratory
University of Florida

The NASA Langley Research Center Mini–Mast is an eighteen bay truss structure cantilevered at one end and free at the other. The bays are numbered one to eighteen starting with one at the cantilevered end. Discrete masses are located at bays ten and eighteen. Three different models of the Mini–Mast were created. The first model was a two beam PDEMOD resulting in a frequency matrix of dimension twenty–four. The second model, which was used in the parameter estimation algorithms, was a one beam PDEMOD whose tip mass was adjusted to produce "good" agreement with the two beam model. The reduction in the frequency matrix from twenty–four to twelve greatly reduces the computational burden. In addition, a 30 element FEM was created for comparison purposes. In all models, the single sensor output (position) was located at the tip (bay 18).

225

NASA MINIMAST - IC AND RESPONSE

● **Initial Condition**

INITIAL DISPLACEMENT PROFILE
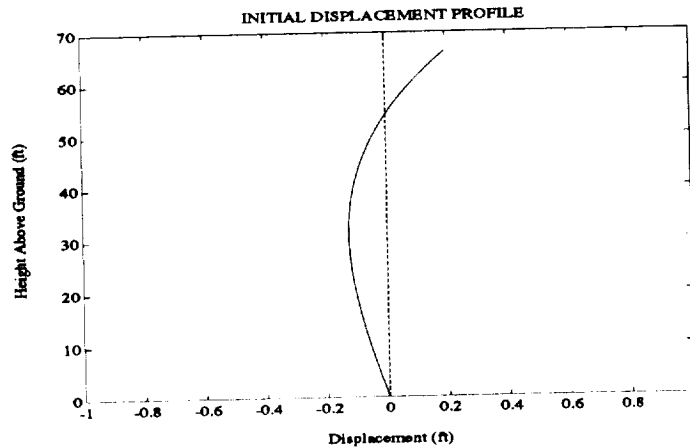
● **Tip Time Response (2 mode summation)**

"MEASURED" TIP DISPLACEMENT

Dynamic Systems and Control Laboratory
University of Florida

The above figures provide the initial displacement and resulting time–history used in the parameter estimation scheme. The initial condition was selected such that multi–modal response was present.
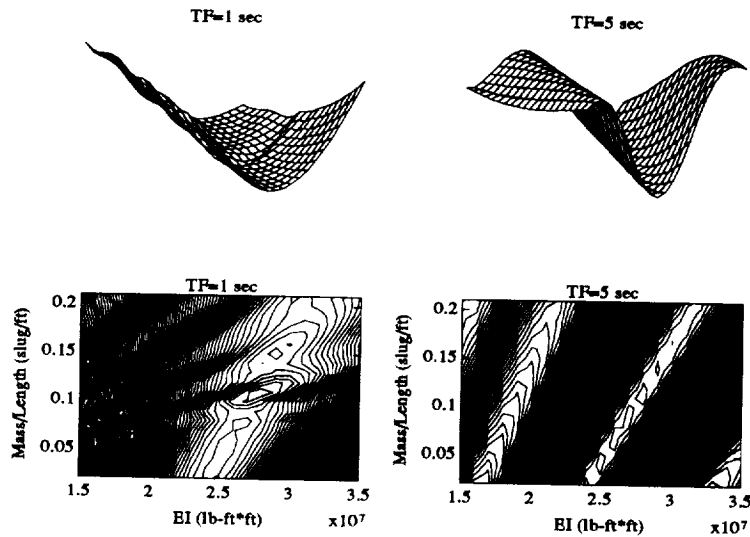
PARAMETER ESTIMATION

⊛ **Determine EI and ϱ Using Measured vs Predicted Time Responses**

$$\min J \ = \ \sum_{i=0}^{n_{sensor}} W_i \int_{t_o}^{t_f} \left(y_{exp}^i(t) - y_{ana}^i(t)\right)^2 dt$$

⊛ **Function Space Characteristics (tf = 1sec, tf = 5sec)**

$$1.5e^7 \leq EI \leq 3.5e^7 \qquad (EI^* = 2.76e^7 \ lb-ft^2)$$

$$0.01 \leq \varrho \leq 0.21 \qquad (\varrho^* = 0.1075 \ slug/ft)$$

TF=1 sec   TF=5 sec

TF=1 sec   TF=5 sec

The parameter estimation problem investigated is to minimize J with respect to EI and ρ. In the above Figure, surface profiles of J are presented for the cases of $t_f = 1$ and $t_f = 5$ seconds. The upper figures are mesh plots of J. The lower subfigures are contour plots of J vs the design parameters. From all figures, it is apparent that the function exhibits local minima and maxima. In addition, from the contour plots, it is evident that the "valley" is rippled, in that there are local minima in the valley.

PARAMETER ESTIMATION PROBLEM

- **Contour Plot Expanded (tf = 1sec)**

CONTOURS: INTEGRAL SQUARED ERROR

Mass/Length (slug/ft) vs *EI (lb-ft*ft)* ×10⁷
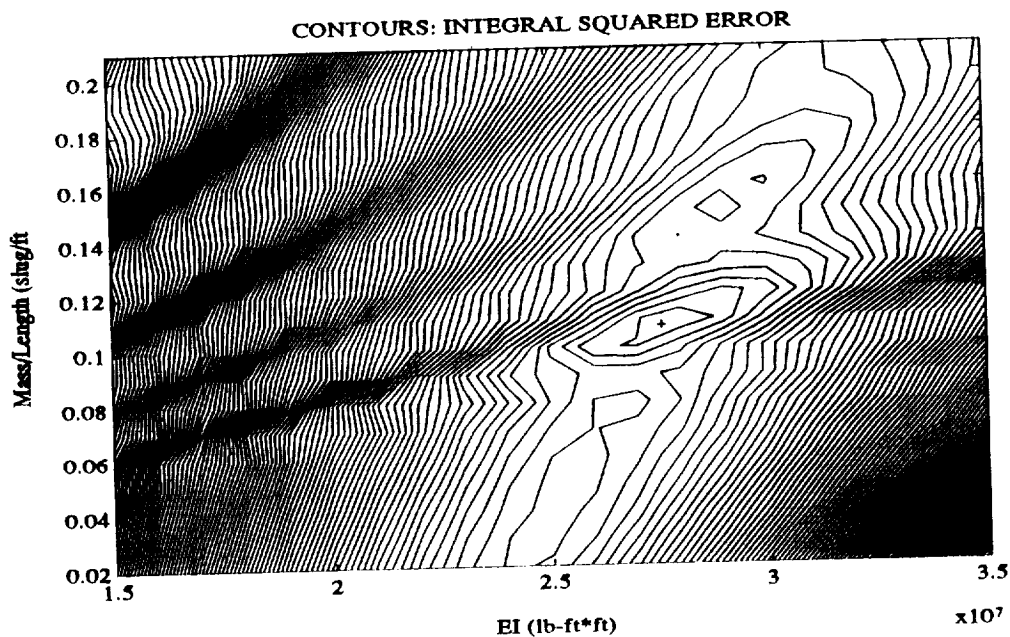
- **Steep/Shallow Walled Problem – similar to classic "Banana Valley Problem**

- **The "Valley" is Rippled**

- **Multi–Modal Function Space**

- **tf = 5sec Case More Difficult Than 1sec Case**

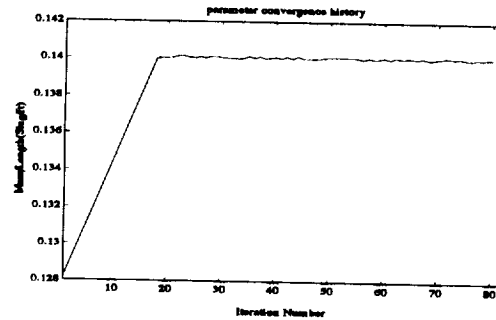Dynamic Systems and Control Laboratory
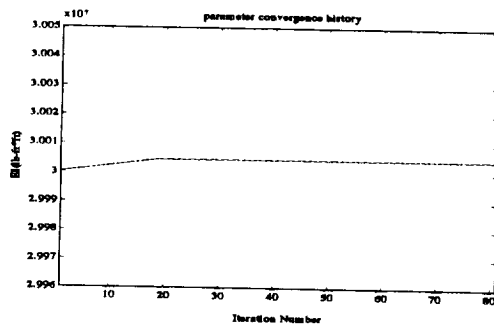University of Florida

The figure above is just an expanded view of the lower left figure of the previous slide. The optimal solution is marked by the " + " symbol. From this, and the previous figure, it is seen that the function has characteristics similar to Rosenbrocks "Banana Valley" problem. The problem at hand has the characteristic steep walled/gentle gradient valley of the "Banana Valley" Problem. The $t_f = 5sec$ case represents the more difficult problem in that the walls are steeper.

228

LEE & HOSSAIN APPROACH

- Parameter Divergence (S^k = diag(1e5, 175): Large)
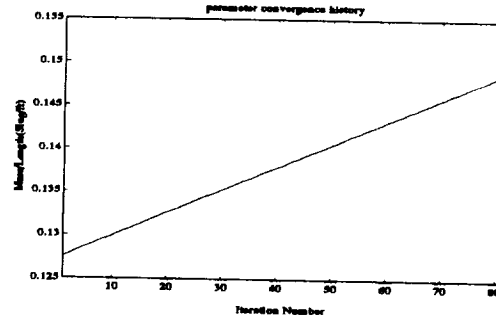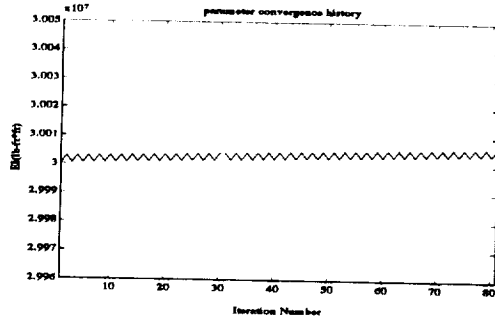
- Parameter Convergence To Local Minimum (S^k = diag(1e4, 95))

The above figures show parameter value vs iteration number for the Lee & Hossain approach. The upper figure corresponds to the case where the step-size matrix, S^k, has been chosen to be to large. It is apparent that the parameter values are diverging and the actual path followed by the design variables is uphill. The lower figure corresponds to the case that there is convergence to a local minima. In comparing the two mass/length plots, it is apparent that the case of choosing S^k to large has caused the algorithm to miss the local minima.

229

LEE & HOSSAIN APPROACH (cont'd)

- **Slower Convergence ($S^k = \text{diag}(1e2, 55)$)**
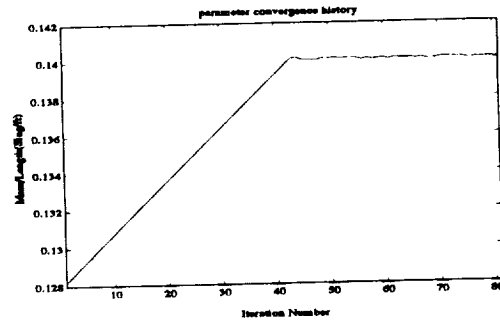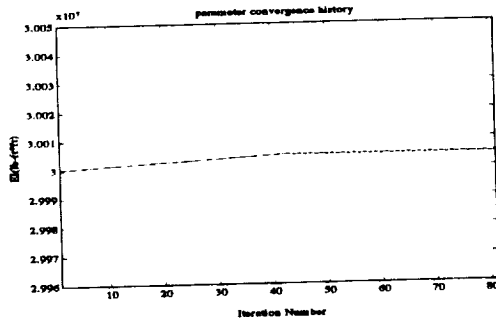
- **Algorithm Performance Sensitive To Selection of S^k**

- **Convergence To Local Minima**

Dynamic Systems and Control Laboratory
University of Florida

In this figure, the value of the step-size matrix was chosen to be lower than the previous case of convergence. The algorithm converges to the same local minima, but requires a greater number of iterations. These slides indicate the sensitivity of the algorithm to step-size selection. Lee & Hossain provide no indication of how to select S^k. Thus, this remains an unresolved research issue for this algorithm.

230

SIMPLIFIED 1-D SEARCH

⊕ Convergence Criteria Set to $\dfrac{\partial J}{\partial q_i} \leq 1e^{-12}$

⊕ Convergence Histories For Four Different Starting Points

Convergence Path:Multiple Starting Points
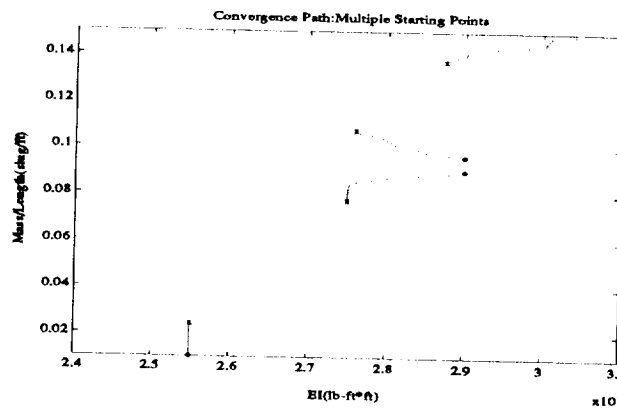
⊕ Starting Points – "o"; Converged Solution – "x"

⊕ One Solution Converges to Global Optimal

⊕ Starting/Converged Points

(2.55e7, 0.01) ⟹ (2.55e7, 0.024)

(2.9e7, 0.096) ⟹ (2.76e7, 0.1070) NOTE: OPTIMAL

(2.9e7, 0.090) ⟹ (2.75e7, 0.077) NOTE:Close to Previous Start

(3.1e7, 0.150) ⟹ (2.875e7, 0.1370)

In the simplified 1–D search strategy, the variable with the corresponding highest gradient is varied. A quadratic approximation technique is used to determine the optimal step–size. The above plot shows the result of the algorithm for four different starting initial conditions. The starting points are indicated by the "o's" and the ending points by "x's". Note that all starting points were in the valley. The stopping criteria used to halt the iterations was when the maximum gradient was lower than approximate machine precision. One of the four starting points ended up near the global minima. However, another nearby starting point actually converged to a farther away local minima.

GENETIC ALGORITHM SOLUTION - Case I

- **Utilized Linear Ranking Scheme to Map Minimization of Integral Into Maximization Problem**

- **Random Initial Population Utilized for Case I**
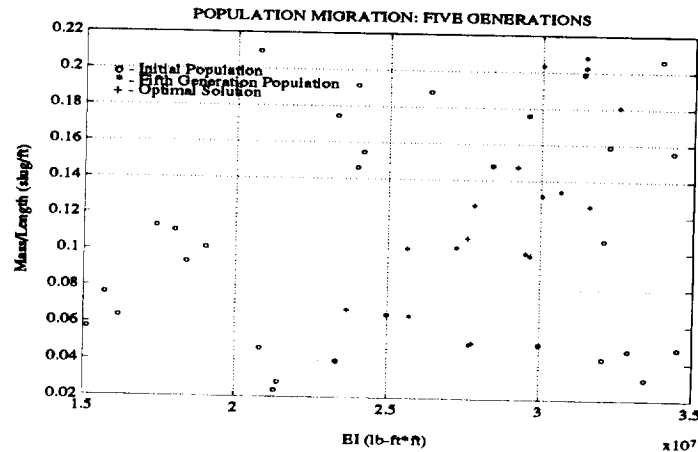
- **Convergence History**

CONVERGENCE HISTORY

Solid - Most Fit Design
Dashed - Average Fitness
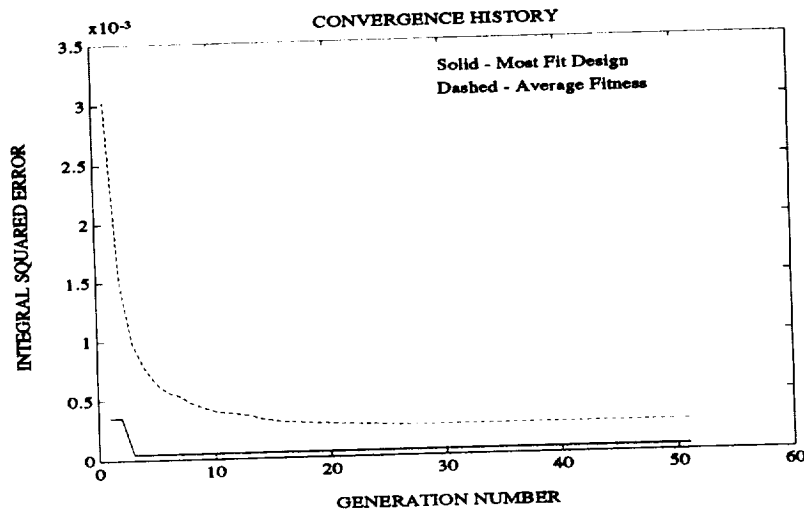
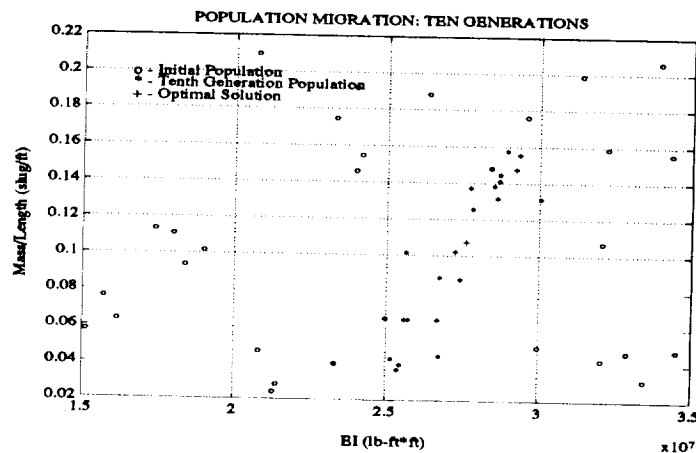- **Note Rapid Convergence: In reality, could stop at ~ten generations**

The above figure shows the convergence history of the Genetic Algorithm parameter estimation approach. The solid line shows the integral value of the most fit member of the population at any given generation. In a similar manner, the dashed line represents the integral value of the average member of the population at any given generation. A linear ranking scheme was utilized to transform the integral minimization problem into a fitness maximization problem. In this scheme, assuming a population size of 30, the member with the lowest integral value (best member) is assigned a fitness of 30; the member with the highest integral value (worst member) is assigned a fitness of 1. This linear ranking scheme was used to avoid the creation of a "super individual". As is the case with most applications of GA's, there is rapid convergence in early generations; this slows considerably as the generation number increases.

GA's POPULATION MIGRATION

⊛ **After Five Generations**

POPULATION MIGRATION: FIVE GENERATIONS

o - Initial Population
* - Fifth Generation Population
+ - Optimal Solution

Mass/Length (slug/ft) vs BI (lb-ft*ft) ×10⁷

⊛ **After Ten Generations**

POPULATION MIGRATION: TEN GENERATIONS

o - Initial Population
* - Tenth Generation Population
+ - Optimal Solution

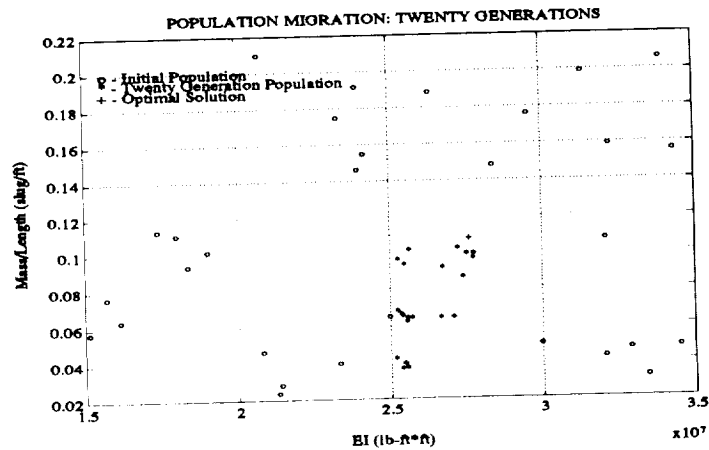Mass/Length (slug/ft) vs BI (lb-ft*ft) ×10⁷

Dynamic Systems and Control Laboratory
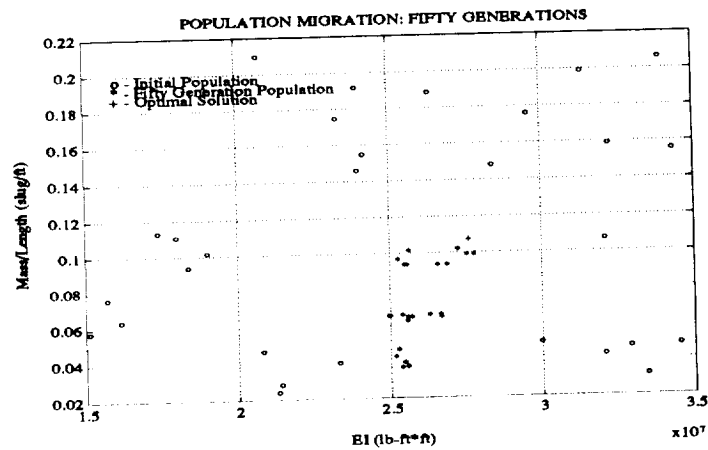University of Florida

This and the next slides show the population migration as a function of generation number. In the figures, the "o's" represent the location of the random, initial population. The "*'s" represent the population members at the stated generation number and the "+" indicates the global minima. After only five generations, most of the members have migrated into the valley. From generations, ten to fifty, it is seen that the migration of members is towards the global minima.

GA's POPULATION MIGRATION

● **After 20 Generations**

POPULATION MIGRATION: TWENTY GENERATIONS

o - Initial Population
o - Twenty Generation Population
+ - Optimal Solution

EI (lb-ft*ft)   x10⁷

● **After 50 Generations**

POPULATION MIGRATION: FIFTY GENERATIONS

o - Initial Population
o - Fifty Generation Population
+ - Optimal Solution

EI (lb-ft*ft)   x10⁷

Dynamic Systems and Control Laboratory
University of Florida

234

GA POPULATION/CONTOURS

● **Population At Generation 20 on Contour Plot**
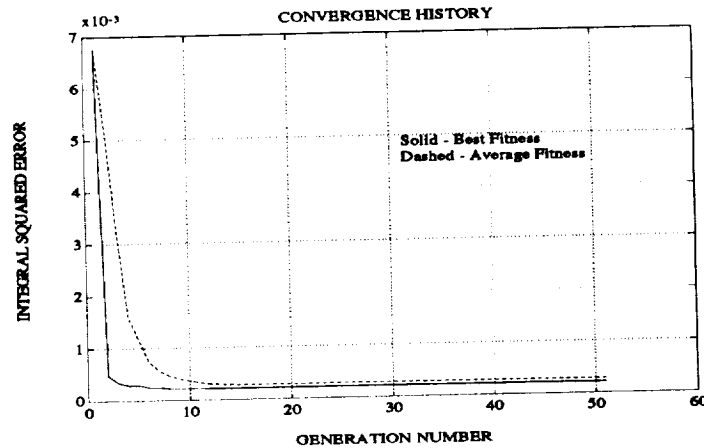

CONTOURS: INTEGRAL SQUARED ERROR
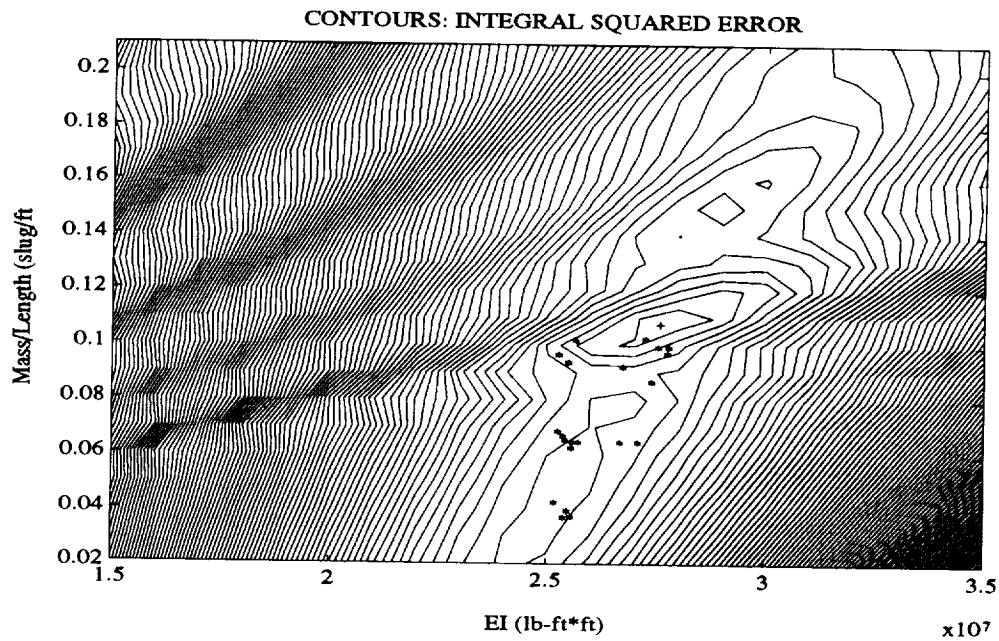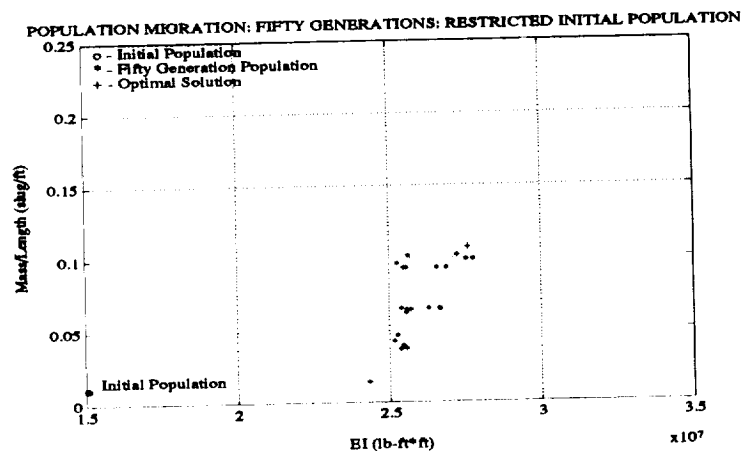
● **All Members in the Valley Searching For Minima**

The above figure superimposes the contour plot with the population location at generation number 20. From this figure, it is clear that the search is now confined to the valley.

GA's - CASE II

- Initial Population Forced to Low Values of EI and ϱ

- Convergence History

- Population Migration

In the previous figures, the initial population was created randomly. Thus, some of the initial members were possibly already in the valley. To truly judge the performance of the GA approach, the initial population in this case was constrained to low values of both design variables. In the lower figure, what appears as a solid dot is really all thirty initial members of the population. After 50 generations, it is obvious that the population has migrated into the valley. The top figure indicates that the majority of the migration was accomplished in the first ten generations.
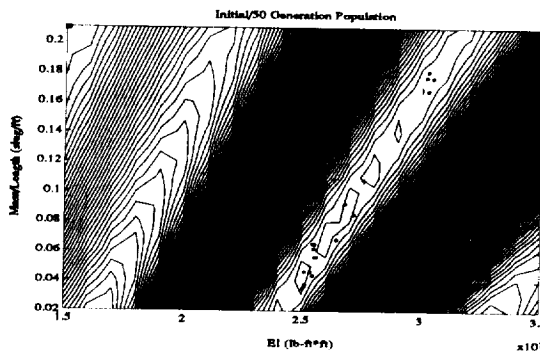
236

GA's - CASE III (5 sec data)

⊛ **Initial/20 Generation – Random Population**

Initial/20 Generation Population
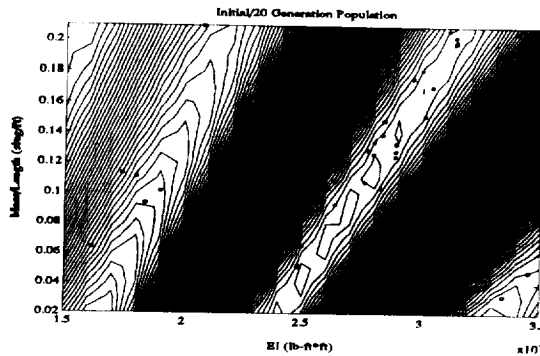
➡ **Population Converging To Valley; Not Yet at Optimal Solution**

⊛ **Initial/50 Generation Scewed Initial**

Initial/50 Generation Population

➡ **Population Converging, But Possibly To Local Solution (Would Need Many Generations To Achieve Optimal)**

Dynamic Systems and Control Laboratory
University of Florida

In Case III, the integral was evaluated over a five second interval. The upper plot shows an overlay of the contours, the initial population (same as in previous random case), and the final population after 20 generations. Again, the population has converged to the valley, but has not yet found the global optimal. In the lower figure, the initial population was constrained to have low values of EI, and high values of ρ. All thirty members are contained in the solid "dot" in the upper left corner of the figure. After 50 generations, all members are in the valley; however, they have not found the global minima. In fact, another 50 generations were run with minimal change in population location. This indicates that although the GA solution appears to perform better than gradient based algorithms for this particular cost function, it still can become trapped in local minima (although theoretically if the number of generations goes to infinity the global minima will be found (by default)).

CONCLUDING REMARKS

- Parameter Estimation Schemes Utilizing Measured Time Domain Data and Distributed Parmeter Modelling Investigated

- "Enhancements" to PDEMOD Include Wittrick-Williams Root Solving and Time-History Simulation

- Demonstrated That "Simple" Time Domain Cost Functions Yield "Difficult" Function Spaces

  ⇨ Need to Search For Improved Time Domain Cost Functions

- Gradient-Based Algorithms Experienced Difficulty With Local Minima and Divergence

- Genetic Algorithms Showed Promise In These Areas, But Used More Function Evaluations

- Note That Mismatch in Function Evaluations Is a Function of Number of Design Variables

  ⇨ Increasing Number of Design Variables Favors Genetic Algorithms

Dynamic Systems and Control Laboratory
University of Florida

In this work, parameter estimation schemes utilizing measured time domain data were investigated. The models used were developed using the PDEMOD approach. Two enhancements to PDEMOD-1 were made in order to develop the parameter estimation algorithms. The first (Wittrick-Williams) reduced the computational burden associated with solving for the structure eigenvalues. The second provided time-simulation capabilities. It was shown by example that a "simple" time-domain cost function actually yielded a difficult function space for the parameter estimation algorithms. The function space was multi-modal and exhibited characteristics similar to the classic "Banana Valley" problem. The gradient-based algorithms experienced severe difficulty. In fact, it was difficult to find starting conditions for which either gradient algorithm converged to the optimal solution. Conversely, the GA approach appeared to perform well. However, the GA used a much greater number of function evaluations. This would not be the case if there were a large number of design variables. For gradient based algorithms, the number of function evaluations per iteration increases approximately linearly with the number of design variables (i.e. each additional design variable requires a gradient calculation). However, because GA's do not require gradient calculations, the number of function evaluations per generation is independent of the number of design variables.

238

REFERENCES

[1] Taylor, L.W., "PDEMOD – Computer Software for Partial Differential Equation Modeling of Flexible Space Structures," Proceedings of the 2nd USAF/NASA Workshop on System Identification and Health Monitoring of Precision Space Structures," Pasadena, California, March, 1990.

[2] Wittrick, W.H. and Williams, F.W., "Natural Frequencies of Elastic Structures," *Journal of Mechanics and Applied Mathematics*, Vol. XXIV, 1971, pp. 265-285.

[3] Lee, K.Y., and Hossain, S.A., "Distributed Systems Approach To Identification of Flexible Structures," *AIAA Journal of Guidance, Control and Dynamics*, Vol. 10, No. 6, 1987, pp. 540-548

[4] Holland, J.H., (1975), *Adaptation in Natural and Artificial Systems*, The University of Michigan Press, Ann Arbor, Michigan.

[5] Zimmerman, D.C. (1991), "A Darwinian Approach to the Actuator Number and Placement Problem with Nonnegligible Actuator Mass," to appear, Journal of Mechanical Systems and Signals.