Determining Important Control Parameters of a Genetic Algorithm

Andrea Haines

NIST

Information Technology Laboratory
Advisors: Kevin Mills & Jim Filliben
7 August 2012

Outline

- Objective
- Genetic Algorithm Background
- Part 1: Sensitivity Analysis
  - Problem Set
  - Experiment Design
  - Results
- Part 2: Exploring a Cloud Simulation's State Space for Failure Scenarios
- Acknowledgements

Objective

The Big Picture

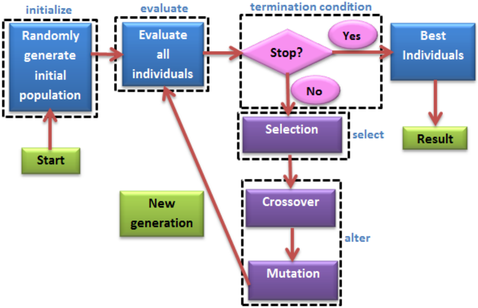
To use a classic Genetic Algorithm to search the large state space (6^{132}) of Koala, a cloud simulation, for settings that drive the model into behavioral directions that indicate system failure and/or degraded operations

How?

- A classic Genetic Algorithm was developed by Kevin Mills, using SLX (Extensible Simulation Language)
- Lacking prior definitive studies, we needed to determine the most important and best settings to use for 7 of the GA's control parameters
- Conducted a sensitivity analysis on the results of the GA on 60 optimization problems

What is a Genetic Algorithm?

GA Flowchart



Bit Representation

Crossover

Individual x	1011	10010111001	111001110
Individual x+1	0010	10111101001	101010011
New Individual x	0010	10010111001	101010011
New Individual x+1	1011	10111101001	111001110

Mutation

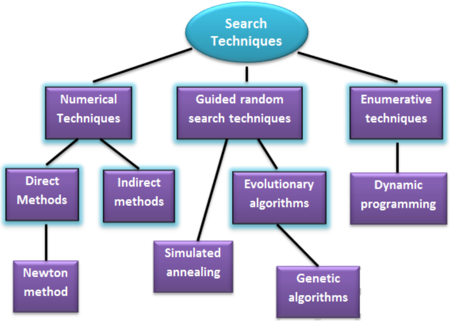
New Individual x	0000 10010111001 100011111
New Individual x+1	1001 10101100000 001001110

$P = \#$ of individuals in population
$t =$ generation $\#$
$P(t) =$ population of individuals for generation t
$t = 0$

initialize $P(t)$
evaluate $P(t)$
while (not termination condition)
begin
  $t = t + 1$
  select $P(t)$ from $P(t-1)$
  alter $P(t)$
  evaluate $P(t)$
end

Why Genetic Algorithms?

Classes of Search Techniques



Advantages of Genetic Algorithms

- Can search a large space
- Can work well when the search space is multimodal
- Can provide a “good” solution
- Can be useful for complex or loosely defined problems

Part 1: Sensitivity Analysis

7 Factors (Parameters) –4 levels (Settings)

	Level 1	Level 2	Level 3	Level 4
x_1 Population Size	50	100	150	200
x_2 Selection Method	SUS	T(.60)	T(.75)	T(.90)
x_3 Elite Selection %	0	2	4	8
x_4 Reboot Proportion	0	0.1	0.2	0.4
x_5 Crossover	0	1	2	3
x_6 Mutation	Adap	0.001	0.0055	0.01
x_7 Precision Scaling	1/2	1	2	4

Partial Problem List

60 numerical optimization problems from literature
The study started with 14 completed problems:

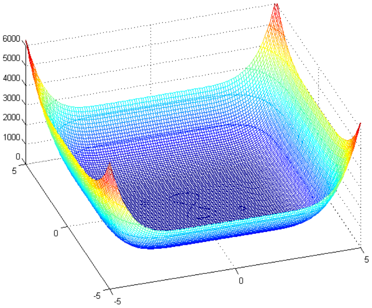
1. AxisParallelHyperEllisoid ($D = 15$)
2. AxisParallelHyperEllisoid ($D = 30$)
3. ChemicalReactor ($D = 5$)
4. ChemicalYield ($D = 2$)
5. DefectiveSprings ($D = 3$)
6. Morris ($D = 10$)
7. Morris ($D = 20$)
8. QuarticWithNoNoise ($D = 100$)
9. QuarticWithNormalNoise ($D = 100$)
10. QuarticWithUniformNoise ($D = 100$)
11. SchafferF6 ($D = 2$)
12. ShekelM5 ($D = 4$)
13. ShekelM7 ($D = 4$)
14. ShekelM10 ($D = 4$)

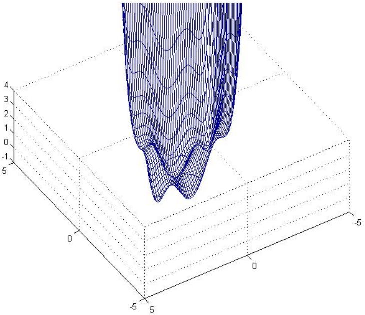
Problem Example

6-Hump Camel Function ($D = 2$)

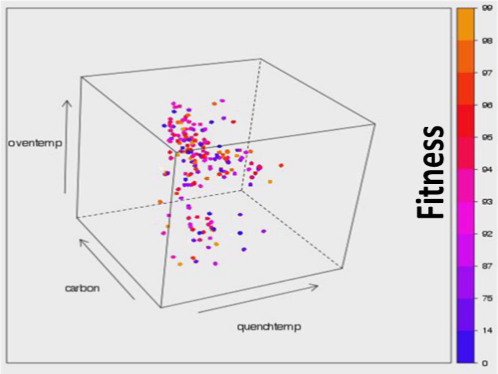
$$f(n, x) = 4x_1^2 - 2.1x_1^4 + \frac{1}{3}x_1^6 + x_1 x_2 - 4x_2^2 + 4x_2^4$$

```
public constant float parameterTraits [PARAMETER][TRAITS] =
{
        // MIN      MAX      PRECISION
        {-5,        5,       0.01},              // P1
        {-5,        5,       0.01}               // P2
```





5D Animation – Defective Springs Problem ($D = 3$)



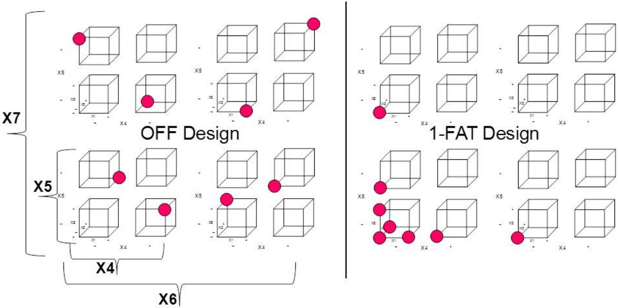
A Genetic Algorithm searching for an optimal combination of oven temperature, quench temperature and carbon concentration in a production process, where fitness is measured as the percentage of non-defective springs produced

Visualization by Sandy Ressler (NIST Math Division)

Experiment Design ($l = 4, k = 7, n = 1024$)

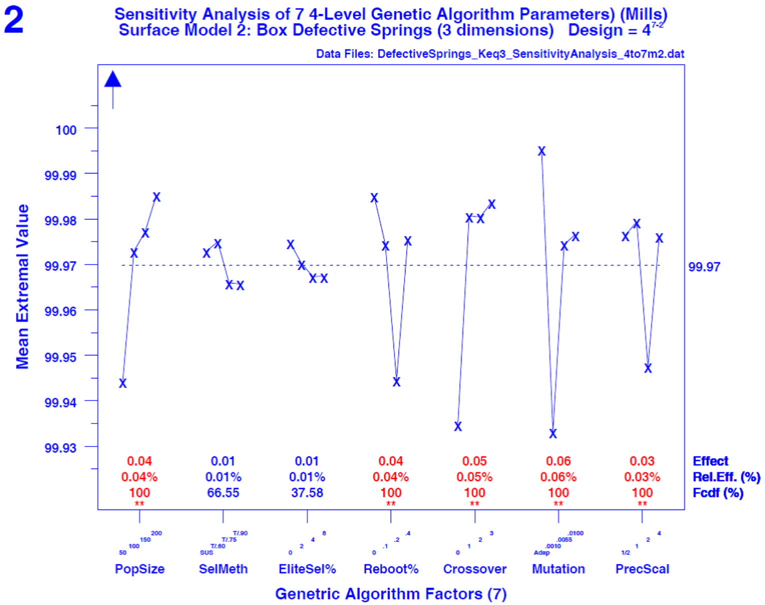
- 4^{7-2} Orthogonal Fractional Factorial (OFF) Design
- Each parameter setting level is used $1024/4 = 256$ times
- $60 \times 1024 = 61440$ runs, instead of $60 \times 16384 = 983040$

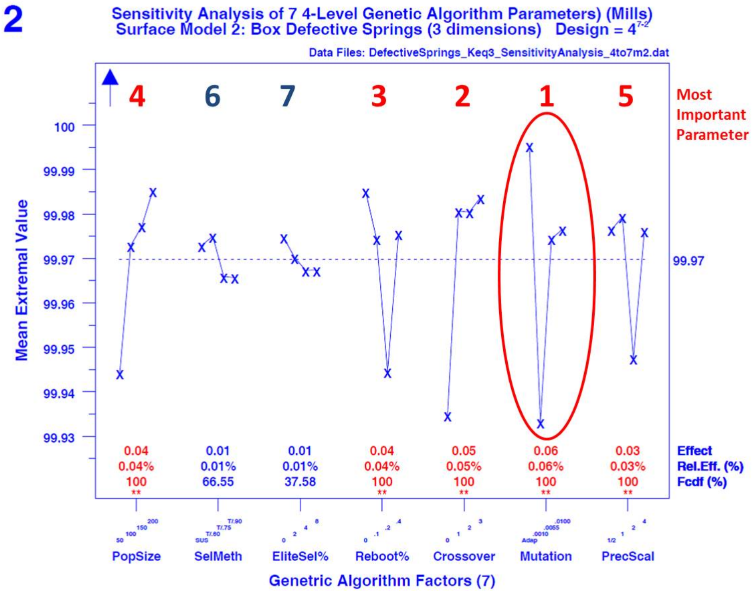
Example:
2-level OFF Design
2^{7-4}

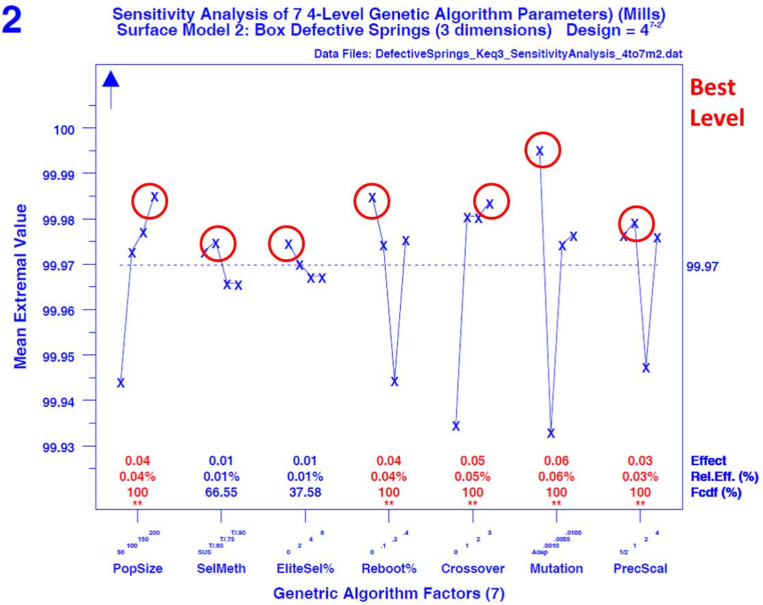


Benefits of OFF Design:

- Superior coverage & robustness when compared with 1-Factor-at-a-Time
- Minimizes variation in effect estimates

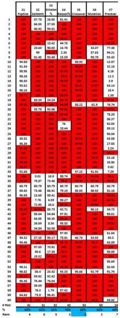

2
Sensitivity Analysis of 7 4-Level Genetic Algorithm Parameters) (Mills)
Surface Model 2: Box Defective Springs (3 dimensions) Design = 4^{7-2}
Data Files: DefectiveSprings_Keq3_SensitivityAnalysis_4to7m2.dat
Mean Extremal Value
100
99.99
99.98
99.97
99.96
99.95
99.94
99.93
99.97
0.04 0.04% 100 **
0.01 0.01% 66.55
0.01 0.01% 37.58
0.04 0.04% 100 **
0.05 0.05% 100 **
0.06 0.06% 100 **
0.03 0.03% 100 **
Effect
Rel.Eff. (%)
Fcdf (%)
50 100 150 200
SUS T/.60 T/.75 T/.90
0 2 4 8
0 .1 .2 .4
0 1 2 3
Adap .0010 .0055 .0100
1/2 1 2 4
PopSize
SelMeth
EliteSel%
Reboot%
Crossover
Mutation
PrecScal
Genetic Algorithm Factors (7)


2
Sensitivity Analysis of 7 4-Level Genetic Algorithm Parameters) (Mills)
Surface Model 2: Box Defective Springs (3 dimensions) Design = 4^{7-2}
Data Files: DefectiveSprings_Keq3_SensitivityAnalysis_4to7m2.dat
4 6 7 3 2 1 5
Most Important Parameter
Mean Extremal Value
100
99.99
99.98
99.97
99.96
99.95
99.94
99.93
99.97
0.04 0.04% 100 **
0.01 0.01% 66.55
0.01 0.01% 37.58
0.04 0.04% 100 **
0.05 0.05% 100 **
0.06 0.06% 100 **
0.03 0.03% 100 **
Effect
Rel.Eff. (%)
Fcdf (%)
50 100 150 200
SUS T/.60 T/.75 T/.90
0 2 4 8
0 .1 .2 .4
0 1 2 3
Adap .0010 .0055 .0100
1/2 1 2 4
PopSize SelMeth EliteSel% Reboot% Crossover Mutation PrecScal
Genetic Algorithm Factors (7)


2
Sensitivity Analysis of 7 4-Level Genetic Algorithm Parameters) (Mills)
Surface Model 2: Box Defective Springs (3 dimensions) Design = 4^7-2
Data Files: DefectiveSprings_Keq3_SensitivityAnalysis_4to7m2.dat
Best Level
Mean Extremal Value
100
99.99
99.98
99.97
99.96
99.95
99.94
99.93
99.97
0.04 0.04% 100 **
0.01 0.01% 66.55
0.01 0.01% 37.58
0.04 0.04% 100 **
0.05 0.05% 100 **
0.06 0.06% 100 **
0.03 0.03% 100 **
Effect
Rel.Eff. (%)
Fcdf (%)
PopSize
SelMeth
EliteSel%
Reboot%
Crossover
Mutation
PrecScal
Genetric Algorithm Factors (7)

Data Summary

Robustness



Factor	p<0.01	Best Level on Individual Functions*			
		Level 1	Level 2	Level 3	Level 4
Population Size (x_1)	41 (68%)	0 (0%)	3 (4%)	13 (19%)	53 (76.8%)
Selection Method (x_2)	32 (53%)	39 (53%)	6 (8%)	15 (20%)	14 (18.9%)
Elite Selection % (x_3)	32 (53%)	14 (18%)	12 (16%)	21 (28%)	29 (38.2%)
Reboot Proportion (x_4)	42 (70%)	30 (45%)	4 (6%)	4 (6%)	29 (43.3%)
Crossover (x_5)	53 (88%)	2 (3%)	14 (20%)	7 (10%)	46 (66.7%)
Mutation (x_6)	50 (83%)	40 (63%)	4 (6%)	12 (19%)	8 (12.5%)
Precision Scaling (x_7)	25 (42%)	25 (40%)	17 (27%)	5 (8%)	16 (25.4%)

*Percentages summed for the levels (1-4) sometimes exceeds 100 because there was no significant difference between the means

Conclusions

Rank	Factors	Best Level (Setting)	p<0.01
1	Crossover (x_5)	4 (3 points)	88%
	Mutation (x_6)	1 (Adaptive)	83%
2	Reboot Proportion (x_4)[a]	1 (0) or 4(0.4)	70%
	Population Size (x_1)	4 (200)	68%
3	Selection Method (x_2)[b]	1 (SUS)	53%
	Elite Selection (x_3)	4 (8%)	53%
4	Precision Scaling (x_7)[c]	1 (1/2 as fine)	42%

a. When the selection threshold (pressure) is too low, tournament selection has an inferior performance
b. Rebooting the population too frequently (after 0.1 or 0.2 of the generations) is not as good as not rebooting at all
c. Reducing the precision of the parameters can give good answers, likely by reducing the size of the search space

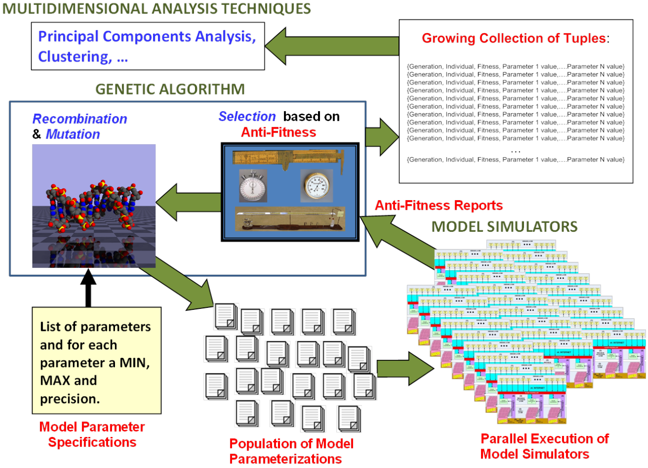
Part 2: Exploring the Koala State Space for Failure Scenarios

The Big Picture (In Progress)

To use the optimized classic Genetic Algorithm to search the large state space (6^{132}) of Koala, a cloud simulation, for settings that drive the model into behavioral directions that indicate system failure and/or degraded operations

Genetic Algorithms and the Cloud Model

How Genetic Algorithms Might Help

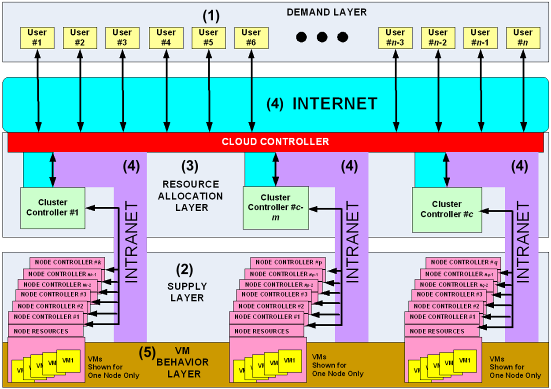


$$\text{Anti-Fitness} = \frac{\#\text{ of users who cannot be served}}{\#\text{ total users}}$$

Koala – Cloud Model

Koala attempts to allocate virtual machines that are requested by users onto physical nodes provided by a cloud

Schematic of Koala IaaS Cloud Computing Model



Chris Dabrowski & Kevin Mills (Networking Division) creators of Koala

Summary of Koala Parameters

Model Element	Total #
User	36
Cloud Controller	32
Cluster Controller	17
Node Controller	22
Internet/Intranet	25
	132

GA-generated Koala Parameters (6^{132})

Sample Koala Chromosome Map

ID#	PARAMETER	MIN	MAX	PRECISION	#VALUES	LOW_BIT	HIGH_BIT	#BITS
1	P_largeNumberOfUserTypes	0	7	1	8	1	3	3
2	P_probabilityNodeNERA	1E-06	0.1	0.01	10	4	7	4
3	P_averageDelayUntilCrash	3600	14400	3600	4	8	9	2
4	P_smallClusterSizeFraction	0.05	0.25	0.05	5	10	12	3
5	P_RelocationOrphanControlOn	0	1	1	2	13	13	1
6	P_orphanDetectionTime	3600	10800	1800	5	14	16	3
7	P_MaxPendingRequests	1	10	1	10	17	20	4
8	P_minNodeFailureDuration	1800	7200	1800	4	21	22	2
9	P_maximumInterSiteDelayPerHop	0.1	1	0.1	10	23	26	4
10	P_minimumSiteCoordinate	-8000	-2000	2000	4	27	28	2
11	P_modeClusterCommunicationCutDuration	21600	36000	7200	3	29	30	2
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
127	P_averageNodeStartupDelay	60	300	60	5	325	327	3
128	P_AdministratorActive	0	1	1	2	328	328	1
129	P_intraSiteExtraDelayprobability	0	0.5	0.1	6	329	331	3
130	P_maximumBootTime	300	900	120	6	332	334	3
131	P_nodesPerCluster	200	2000	200	10	335	338	4
132	P_averageThinkTime	900	1800	300	4	339	340	2

Summary of the Summer

1. Part 1: Sensitivity Analysis
   - Implemented and tested 46 optimization problems
   - Analyzed data from all 60 problems
   - Determined the ranking for the 7 GA parameters and their best settings
2. Part 2: Koala Cloud Model
   - Compiled list of 132 parameters and specified ranges and precisions
   - Reflected changes into GA controller and generated a chromosome map
   - Wrote code to convert chromosome map into Koala parameters

Acknowledgements

- Kevin Mills
- Jim Filliben
- The SURF program and the SURF directors

Questions?

Other Problems

15. Ackley ($D = 10$)
16. Beale ($D = 2$)
17. Bohachevsky ($D = 2$)
18. Branin ($D = 2$)
19. Camel 3-hump ($D = 2$)
20. Camel 6-hump ($D = 2$)
21. Colville ($D = 4$)
22. Corana ($D = 4$)
23. Dekkers and Aarts ($D = 2$)
24. Easom ($D = 2$)
25. Gear ($D = 4$)
26. Goldstein-Price ($D = 2$)
27. Griewank ($D = 2$)
28. Griewank ($D = 10$)
29. Hartman ($D = 3$)
30. Hartman ($D = 6$)
31. Hosaki ($D = 2$)
32. Kowalik ($D = 4$)

Other Problems

33. Levy ($D = 15$)
34. Levy ($D = 30$)
35. Levy ($D = 60$)
36. McCormick ($D = 2$)
37. Michalewitz ($D = 15$)
38. Michalewitz ($D = 30$)
39. Michalewitz ($D = 60$)
40. Multimod ($D = 30$)
41. Neumaier3 ($D = 10$)
42. Paviani ($D = 10$)
43. Perm ($D = 10$)
44. Periodic ($D = 2$)
45. Plateau ($D = 5$)
46. Powell ($D = 4$)
47. Rosenbrock ($D = 15$)
48. Salomon ($D = 10$)
49. Schwefel3 ($D = 3$)
50. Shubert ($D = 2$)

Other Problems

51. Sinusoidal ($D = 10$)
52. SphereModel ($D = 15$)
53. SphereModel ($D = 60$)
54. Step ($D = 15$)
55. Step ($D = 30$)
56. Step ($D = 60$)
57. Watson ($D = 6$)
58. Zakharov ($D = 15$)
59. Zettl ($D = 2$)