Combining Genetic Algorithms & Simulation to Search for Failure Scenarios in System Models

*Kevin Mills**, Chris Dabrowski, Jim Filliben and Sandy Ressler
(work done at NIST prior to US Government Shutdown)

Cyber Security Technical Center Distinguished Speaker Series

October 16, 2013

**Today speaking in his capacity as a private citizen*

Outline

- Motivation & Context
- Method
- Case Study
  - *Koala* IaaS Cloud Simulator
  - Searching for Failure Scenarios
  - Evaluating Method
- Conclusions & Future Work

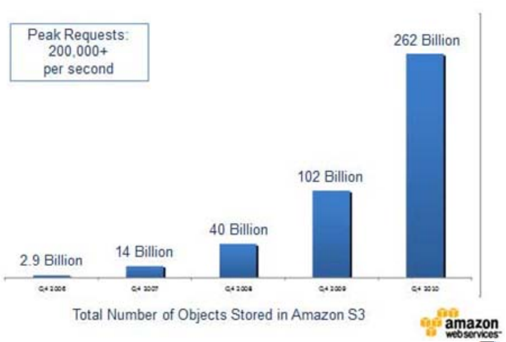
GROWING GLOBAL DEPENDENCE ON COMPLEX INFO. SYSTEMS

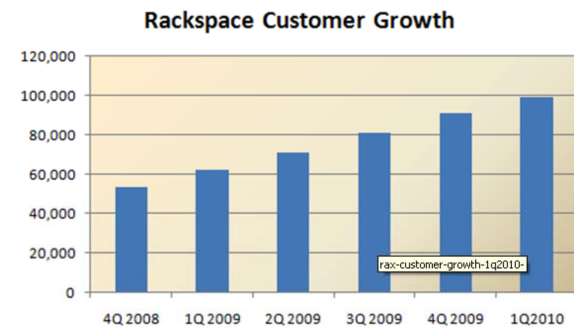
Netflix Moves Into the Cloud With Amazon Web Services

Kundra Outlines "Cloud First" Policy for U.S. Government

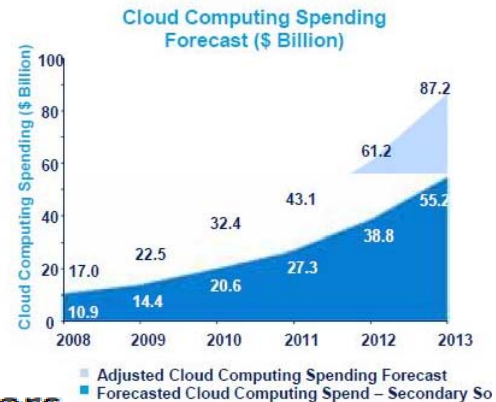
Cloud-Based Infrastructure as a Service Comes to Government

The Cloud Scales: Amazon S3 Growth





Rackspace Hits 100,000 Customers



SAP Exec: Here's Why We Spent $8 Billion On Two Cloud Companies

Microsoft Touts 'High Tens of Thousands' of Windows Azure Customers

Amazon cloud accessed daily by a third of all 'Net users

Cloud provider and the Internet becoming one and the same

IBM to battle Amazon in the public cloud

Oracle Transformed - It's Now All about the Cloud

Salesforce Chatterizes 10,000 Of Its Customers First Week After Public Launch

WE CAN BUILD & DEPLOY SUCH SYSTEMS, BUT CAN WE UNDERSTAND, PREDICT & CONTROL THEM?

Amazon EC2 Outage Explained and Lessons Learned

Posted by Abel Avram on Apr 29, 2011

EC2 OUTAGE REACTIONS SHOWCASE WIDESPREAD IGNORANCE REGARDING THE CLOUD

Rackspace outage was third in two days

SalesForce outages show SaaS customers dependence on providers' DR plans

A storm in Virginia ruined Friday night movie-watching in California. Welcome to the Cloud. (Photo: F

Google Talk, Twitter, Azure Outages: Bad Cloud Day

How did Amazon have a cloud service outage that was caused by generator failure?

Salesforce.com hit with second major outage in two weeks

BUSINESS

Microsoft's Azure Cloud Suffers Serious Outage

Storms, leap second trigger weekend of outages

AWS outages, bugs and bottlenecks explained by Amazon

Never-before-seen software bug caused flood of requests creating a massive backlog in the system

What's happened to the cloud?

Are major cloud outages in recent times denting confidence?

(Real) Storm Crushes Amazon Cloud, Knocks out Netflix, Pinterest, Instagram

BY ROBERT MCMILLAN 06.30.12 3:39 PM

According to the International Working Group on Cloud Computing Resiliency (IWGCR), the total downtime of 13 well-known cloud services since 2007 amounts to 568 hours, which has an economic impact of around $71.7 million dollars.

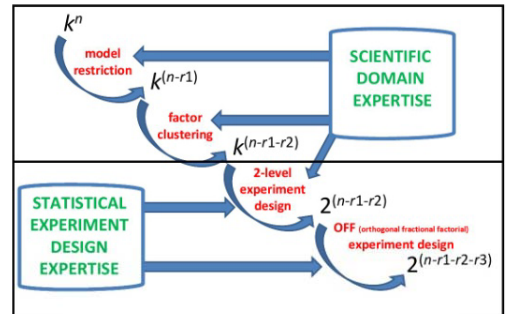
PAST NIST RESEARCH FROM 2006-2011

How can we understand the influence of distributed control algorithms on global system behavior and user experience?

- Mills, Filliben, Cho, Schwartz and Genin, **Study of Proposed Internet Congestion Control Mechanisms**, NIST SP 500-282 (2010).
- Mills and Filliben, "**Comparison of Two Dimension-Reduction Methods for Network Simulation Models**", *Journal of NIST Research* 116-5, 771-783 (2011).
- Mills, Schwartz and Yuan, "**How to Model a TCP/IP Network using only 20 Parameters**", *Proceedings of the Winter Simulation Conference* (2010).
- Mills, Filliben, Cho and Schwartz, "**Predicting Macroscopic Dynamics in Large Distributed Systems**", *Proceedings of ASME* (2011).
- Mills, Filliben and Dabrowski, "**An Efficient Sensitivity Analysis Method for Large Cloud Simulations**", *Proceedings of the 4th International Cloud Computing Conference*, IEEE (2011).
- Mills, Filliben and Dabrowski, "**Comparing VM-Placement Algorithms for On-Demand Clouds**", *Proceedings of IEEE CloudCom*, 91-98 (2011).

For more see: http://www.nist.gov/itl/antd/emergent_behavior.cfm

What to measure



Under what conditions


NIST Special Publication 500-282

Study of Proposed Internet Congestion Control Mechanisms

May 2010

http://www.nist.gov/itl/antd/Congestion_Control_Study.cfm

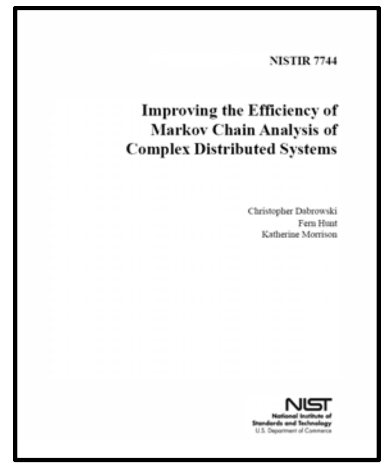
At an affordable cost

ONGOING NIST RESEARCH FROM 2012-PRESENT

How can we increase the reliability of complex information systems?

- **Research Goals**: (1) develop and evaluate **design-time methods** that system engineers can use to detect existence and causes of costly failure regimes prior to system deployment and (2) develop and evaluate **run-time methods** that system managers can use to detect onset of costly failure regimes in deployed systems, prior to collapse.

- **Recent Research**: investigating **design-time methods** –
  a. **Markov Chain Modeling + Cut-Set Analysis + Perturbation Analysis** (e.g., Dabrowski, Hunt and Morrison, "Improving the Efficiency of Markov Chain Analysis of Complex Distributed Systems", NIST IR 7744, 2010).
  b. **Anti-Optimization (AO) + Genetic Algorithm (GA) – TODAY'S PRESENTATION**


NISTIR 7744

Improving the Efficiency of Markov Chain Analysis of Complex Distributed Systems

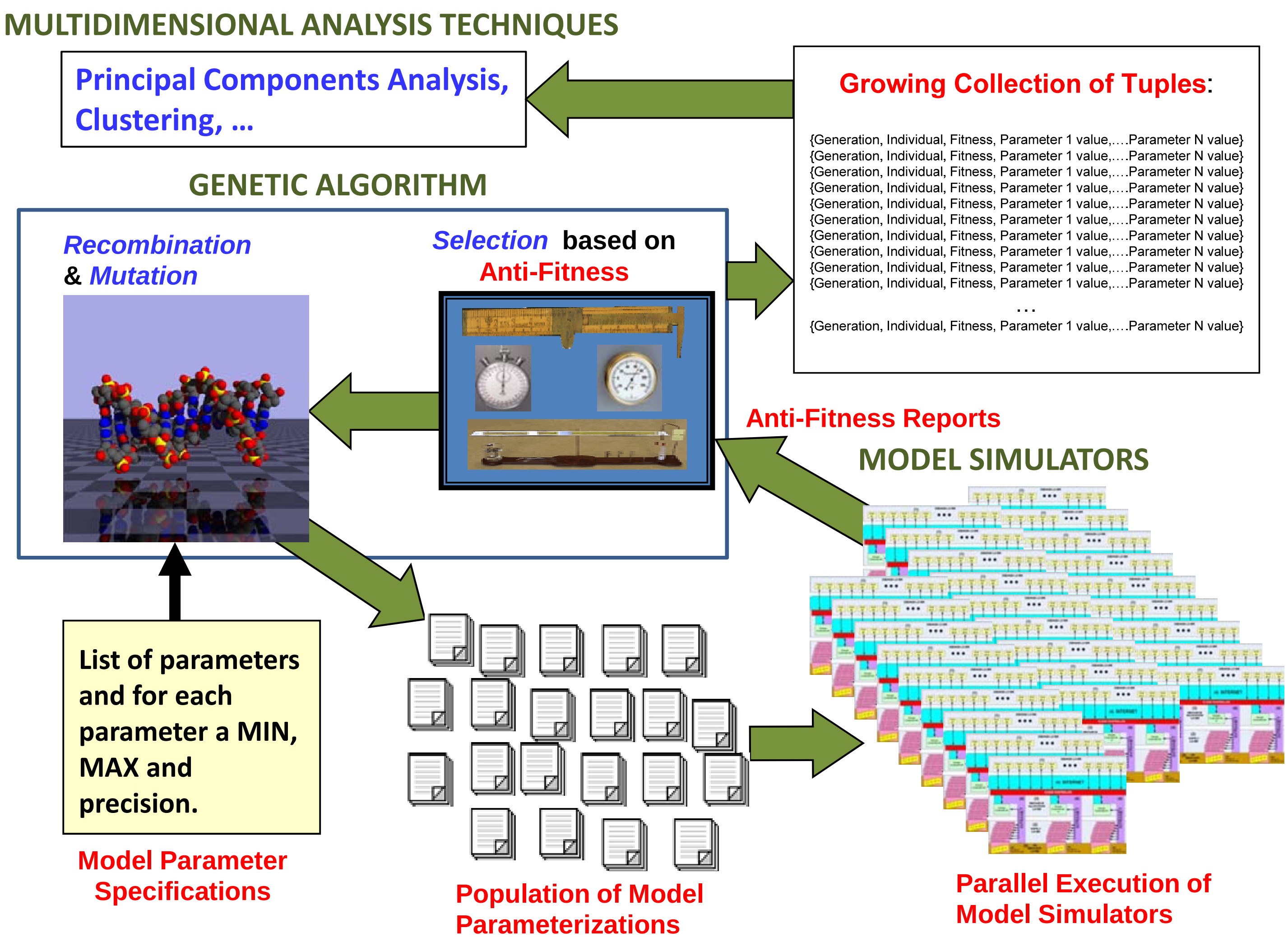
Christopher Dabrowski
Fern Hunt
Katherine Morrison

NIST

http://www.nist.gov/itl/antd/upload/NISTIR7744.pdf

- **Ongoing (well, on hiatus at the present time): investigate run-time methods** based on approaches that may provide early warning signals for critical transitions in large systems (e.g., Scheffer et al., "Early-warning signals for critical transitions", *NATURE*, 461, 53-59, 2009).

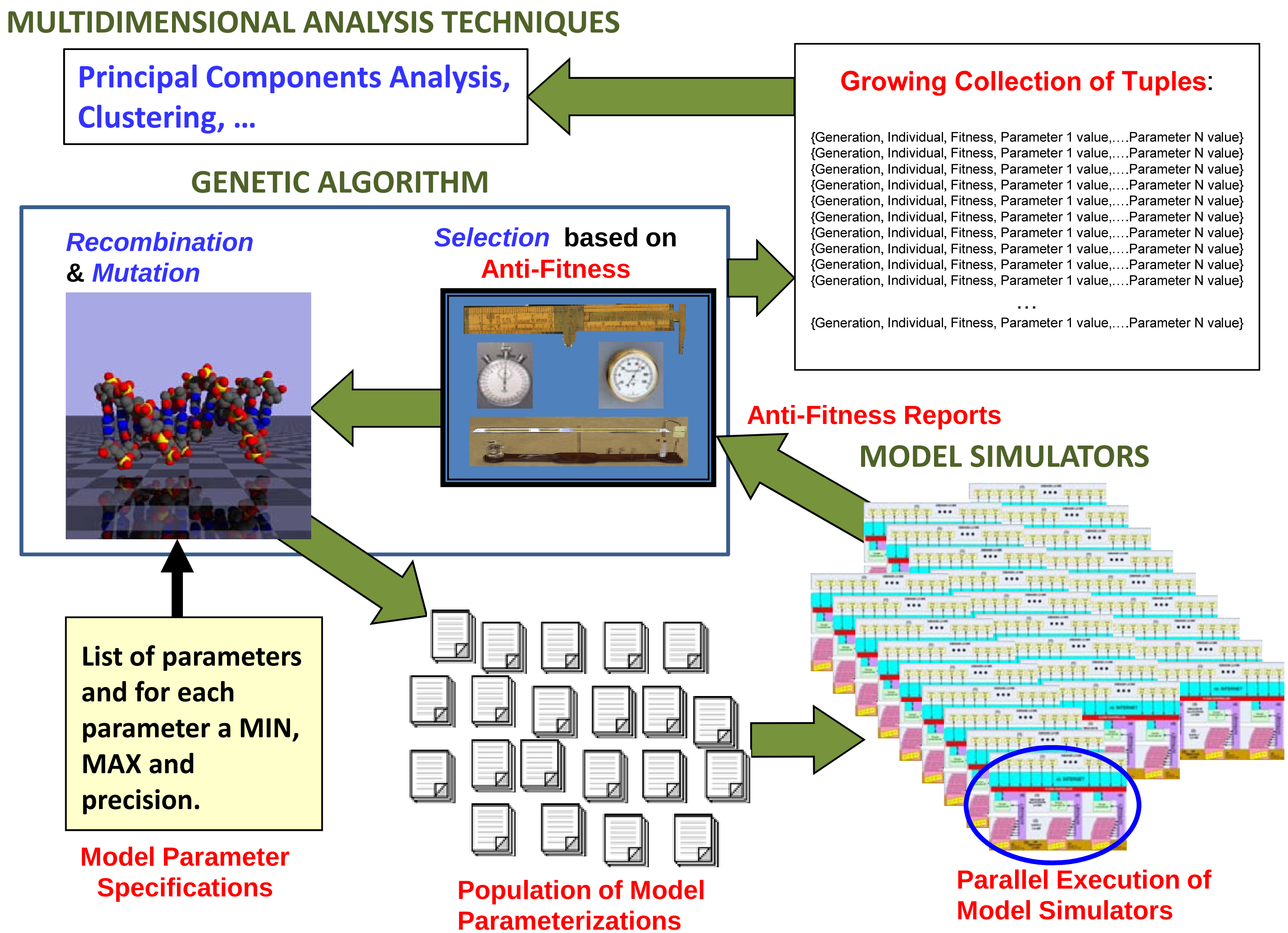
Method: Genetic Algorithm (GA) steers a population of simulators to search for parameter combinations that lead to system failure



Case Study Topics

1. *Koala* Simulator
2. *Koala* Parameters & Representation as Chromosomes
3. Genetic Algorithm
4. Population of *Koala* Simulators
5. Dynamics of GA Search
6. Analysis Method
7. Results from Four GA Searches

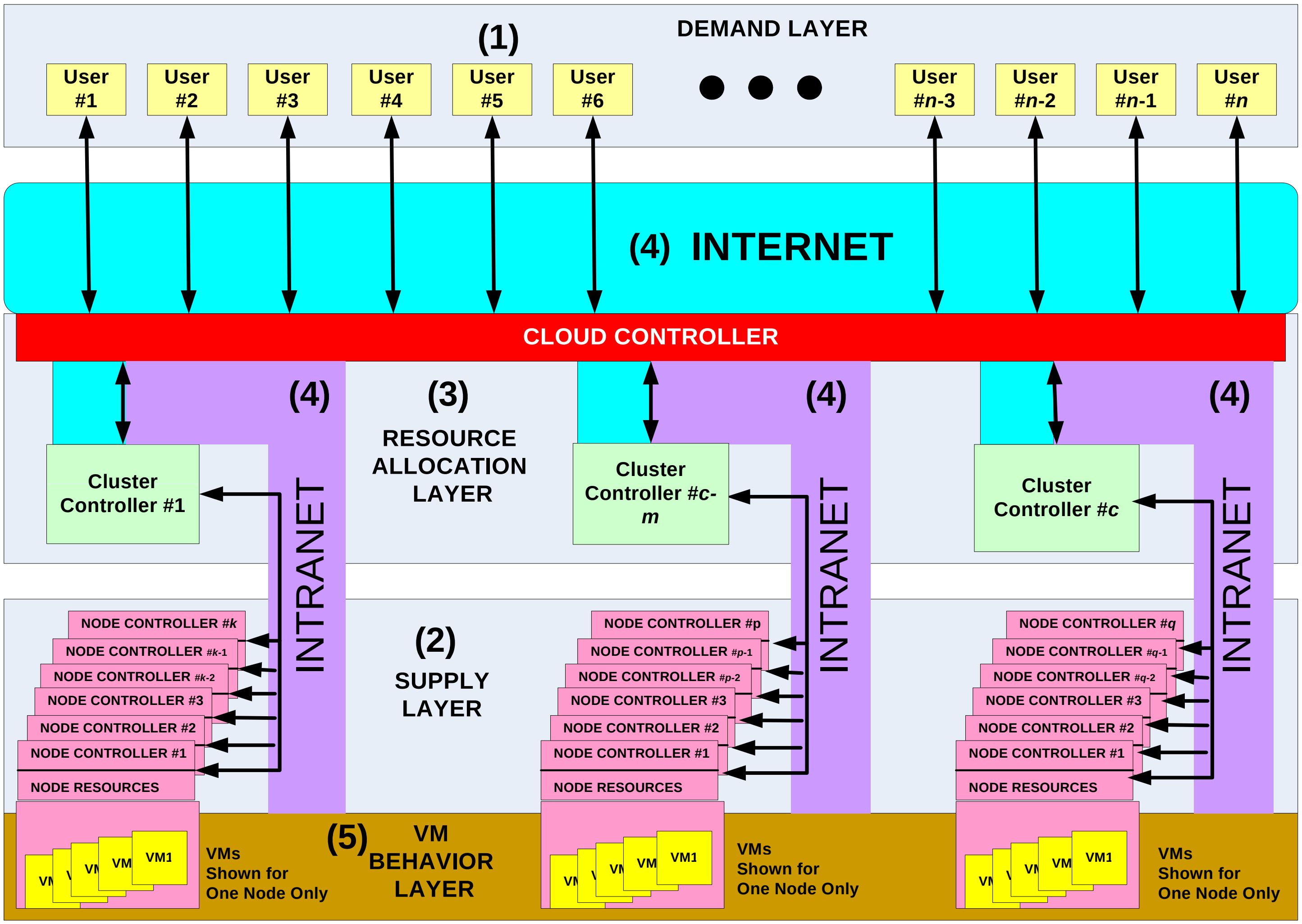
Topic 1: *Koala* Simulator



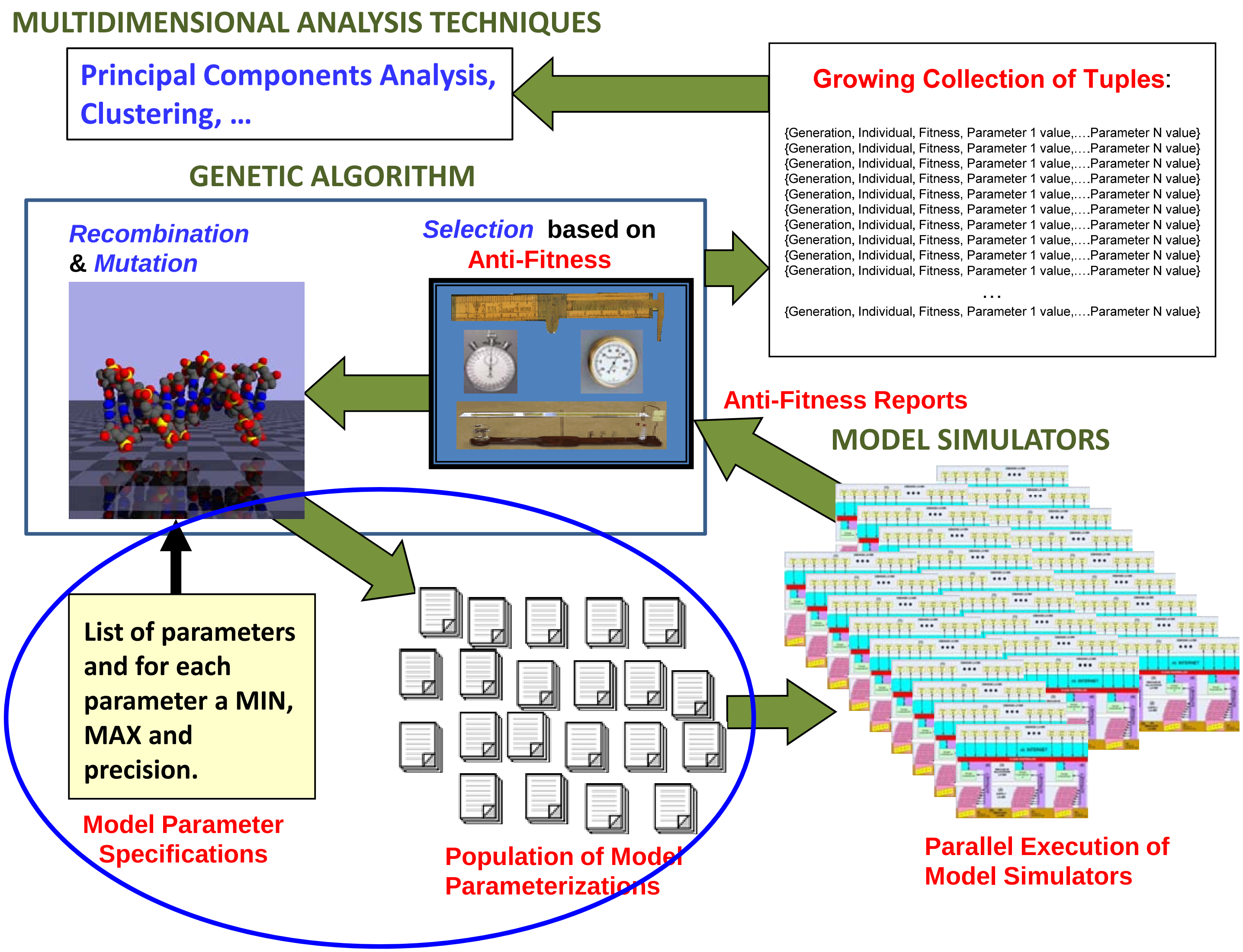
In our case study we defined anti-fitness as the proportion of arriving users not served

Schematic of *Koala* IaaS Cloud Simulator

Demand and supply layers modeled after Amazon EC2
Internal structure modeled after Eucalyptus v1.6



Topic 2: *Koala* Parameters & GA Representation as Chromosomes (i.e., bit strings)



Summary of *Koala* Parameters to Search Over

Test Case – Can GA find VM Leakage due to message loss and lack of orphan control?

Failure scenario found manually by accident and described in C. Dabrowski and K. Mills, "VM Leakage and Orphan Control in Open-Source Clouds", *Proceedings of IEEE CloudCom 2011*, Nov. 29-Dec. 1, Athens, Greece, pp. 554-559.

Model Element	Parameter Category: Behavior	Structure	Asymmetry	Failure	Total
User	28	2	4	0	34
Cloud Controller	21	4	5	0	30
Cluster Controllers	11	5	3	0	19
Nodes	6	0	0	14	20
Intra-Net/Inter-Net	4	11	2	9	26
Totals	70	22	14	23	129

Average # values per parameter is about 6, so search space is ≈ 6^{129}
i.e., ≈ 10^{100} scenarios are possible

- **adapted 125-parameter Koala IaaS simulator to be GA controllable**
- **added 4 *Koala* parameters to turn on/off logic to control (a) creation orphans, (b) termination orphans, (c) relocation orphans and (d) administrator actions**

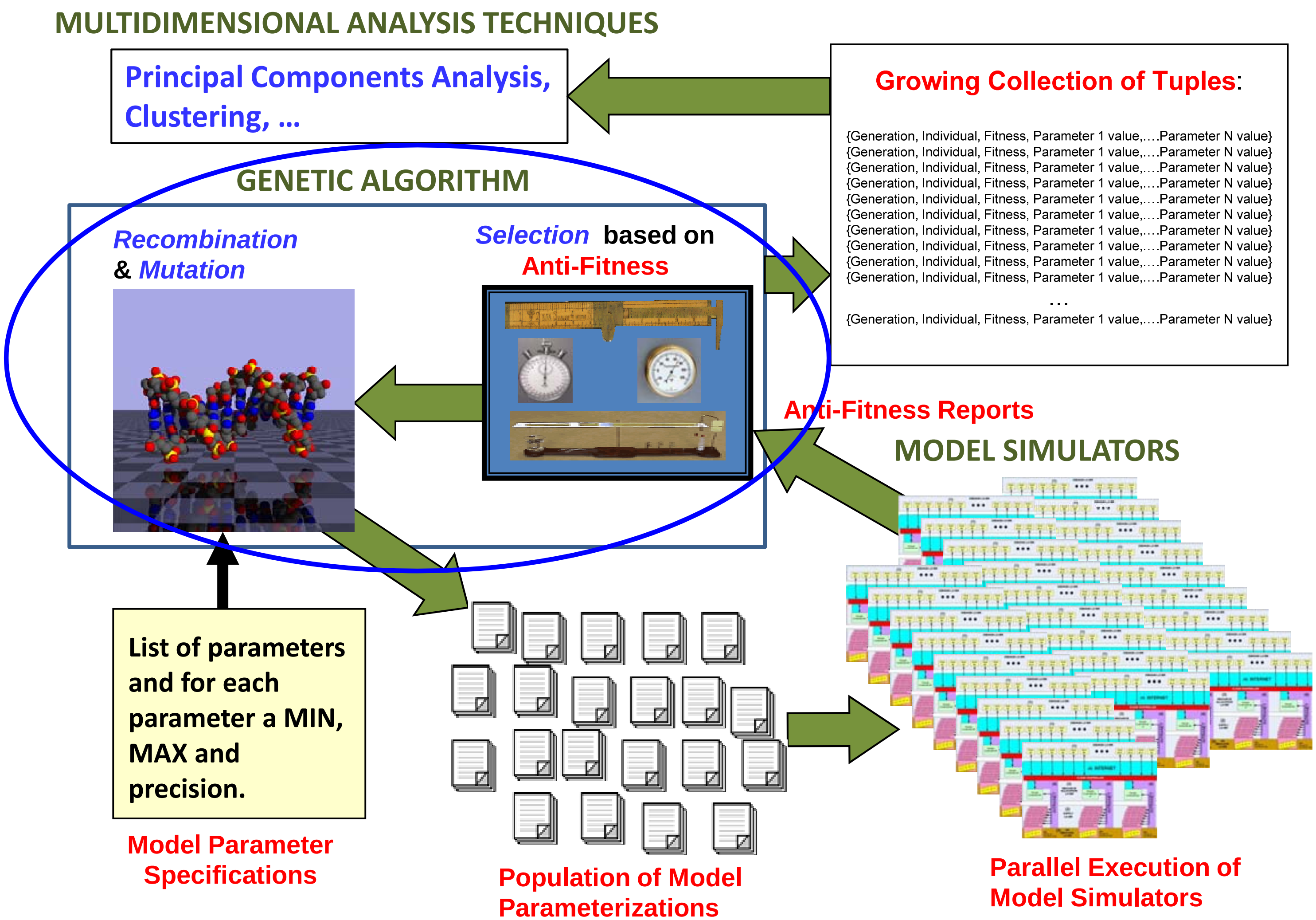
Sample Chromosome Specification

Koala Parameter Space (Size = 10^{100})

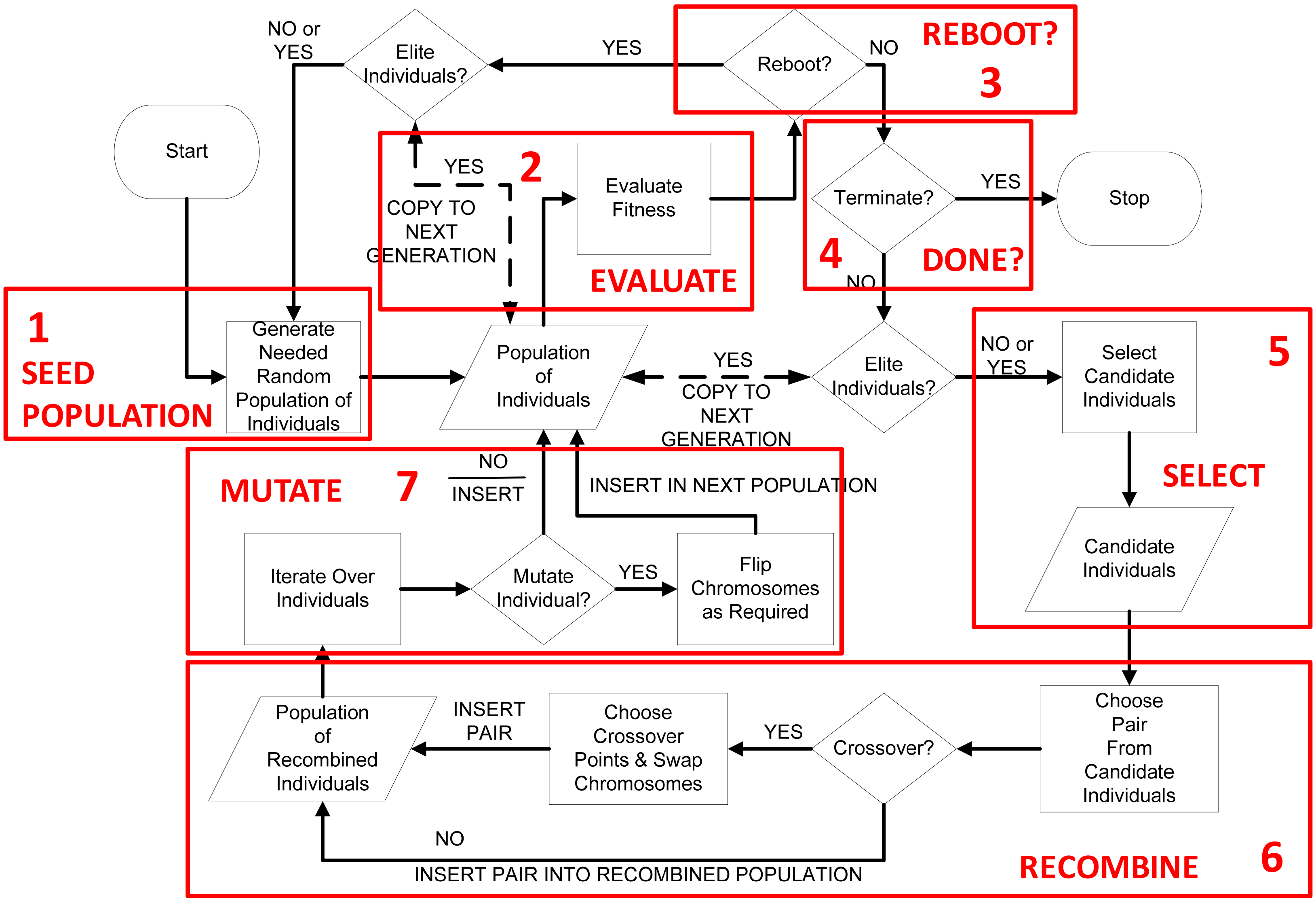
Genetic Algorithm Computed Chromosome Map (Size = 2^{334})

PARAMETER	MIN	MAX	PRECISION	#VALUES	LOW_BIT	HIGH_BIT	#BITS
P_CreateOrphanControlOn	0	1	1	2	36	36	1
P_TerminationOrphanControlOn	0	1	1	2	58	58	1
P_RelocationOrphanControlOn	0	1	1	2	11	11	1
P_AdministratorActive	0	1	1	2	330	330	1
P_clusterAllocationAlgorithm	0	5	1	6	31	33	3
P_describeResourcesInterval	600	3600	600	6	81	83	3
P_nodeResponseTimeout	30	90	30	3	210	211	2
P_TerminatedInstancesBackOffThreshold	3	6	1	4	56	57	2
P_TerminationBackOffInterval	180	360	60	4	88	89	2
P_TerminationRetryPeriod	600	1200	300	3	316	317	2
P_StaleShadowAllocationPurgeInterval	600	3600	600	6	242	244	3
P_cloudAllocationCriteria	0	3	1	4	321	322	2
P_clusterShadowPurgeLimit	1	21	5	5	290	292	3
P_instancePurgeDelay	180	600	60	8	98	100	3
P_clusterEvaluationResponseTimeout	60	120	30	3	14	15	2
P_MaxPendingRequests	1	10	1	10	72	75	4
P_CloudTerminatedInstancesBackOffThreshold	3	6	1	4	169	170	2
P_CloudTerminationBackOffInterval	180	360	60	4	40	41	2
P_CloudTerminationRetryPeriod	3600	10800	1800	5	297	299	3
P_ClusterShutdownGracePeriod	86400	2.59E+05	43200	5	147	149	3
●	●	●	●	●	●	●	●
P_RequestEvaluatorTimeoutWaitProportion	0.1	0.4	0.1	4	145	146	2
P_RequestEvaluatorClusterMinimumResponse	0.6	0.9	0.1	3	269	270	2
P_MaxRelocationDuratonProportion	0.65	0.95	0.1	4	90	91	2
P_MaximumRelocateDescribeRetries	4	16	2	7	254	256	3
P_AverageCloudAdministratorAttentionLatency	28800	86400	14400	5	308	310	3
P_AverageCloudAdministratorShutdownDelay	300	900	300	3	45	46	2
P_avgTimeToClusterCommunicationCut	2.88E+06	2.88E+07	2.88E+06	10	217	220	4

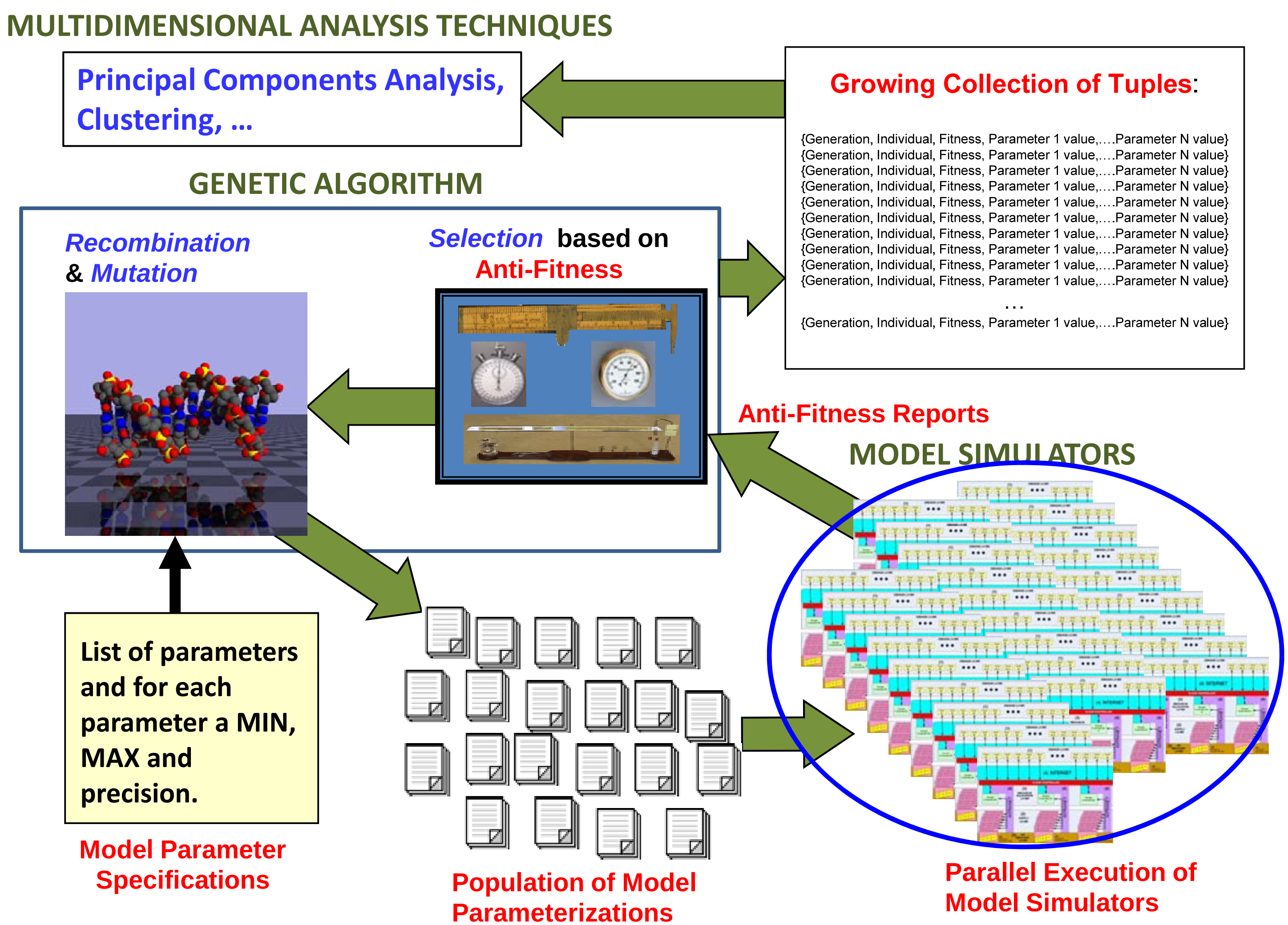
Topic 3: Genetic Algorithm



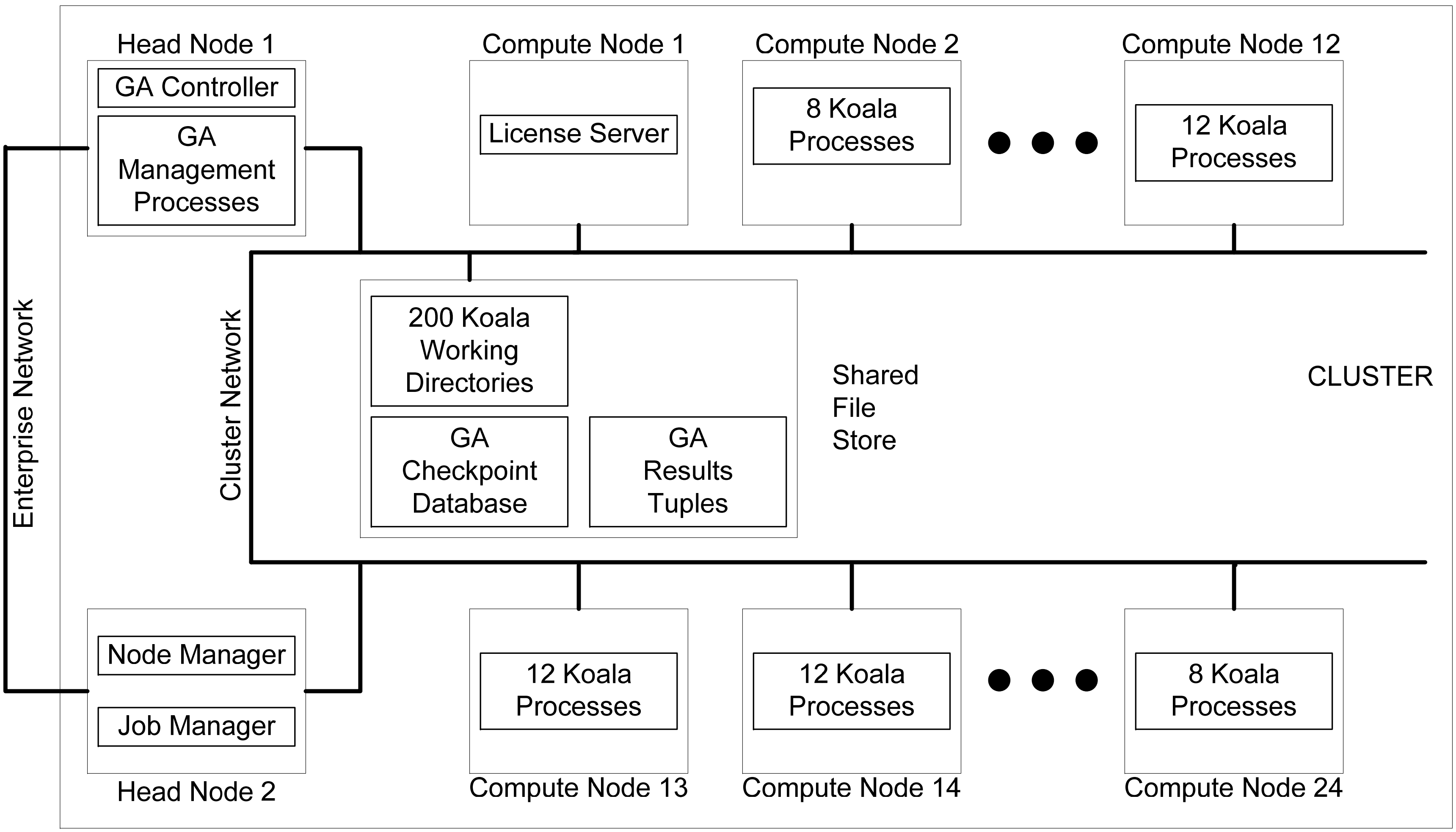
Genetic Algorithm Flow Chart



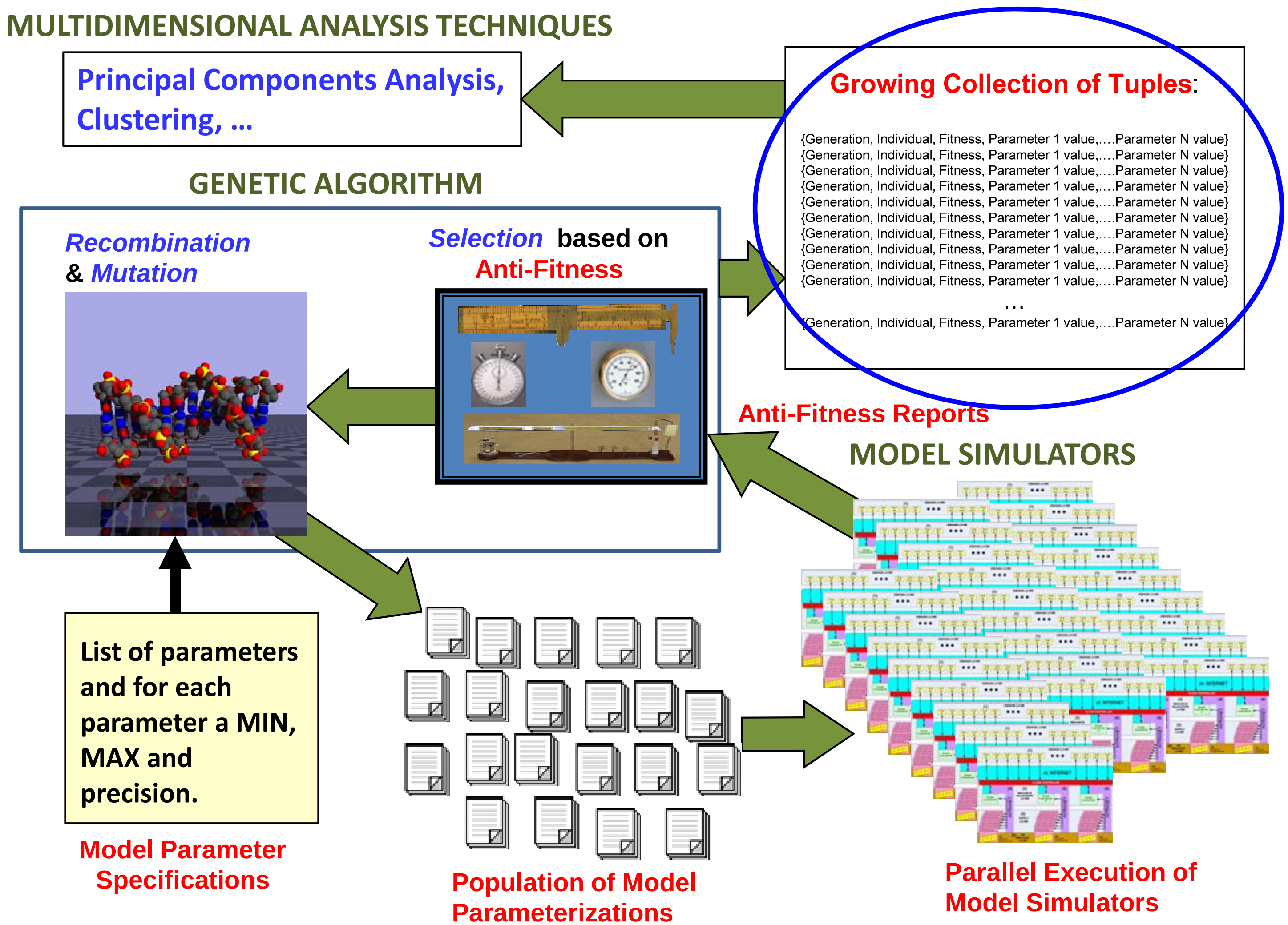
Topic 4: Population of *Koala* Simulators



Population of *Koala* Simulators Deployed on a High Performance Computing Cluster



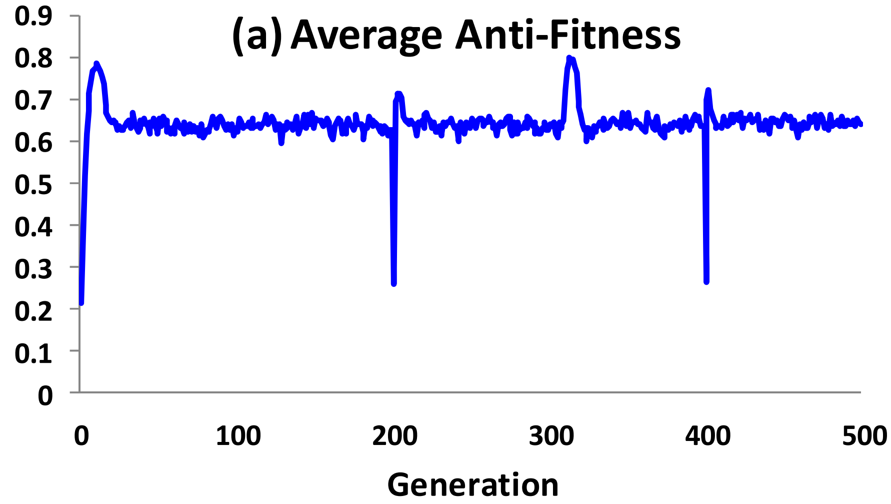
Topic 5: Dynamics of GA Search

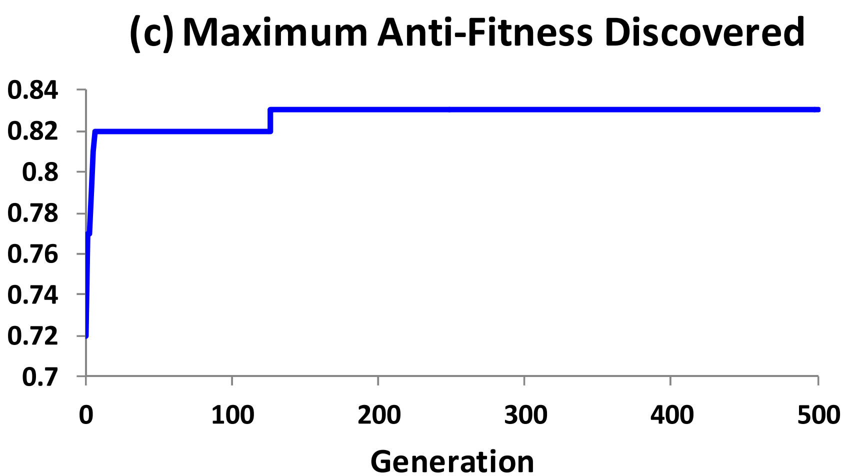


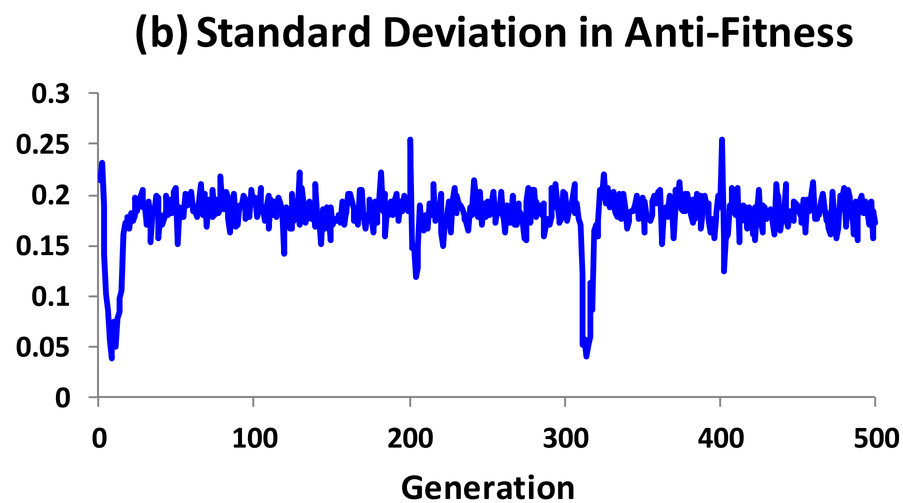
One GA Search over 500 Generations

GENETIC ALGORITHM CONTROL PARAMETERS

Generations	500
Population Size	200 Individuals
Elite Per Generation	16 Individuals
Reboot After	200 Generations
Selection Method	Stochastic Uniform Sampling
# Crossover Points	3
Mutation Rate	0.001 ≤ Adaptive ≤ 0.01

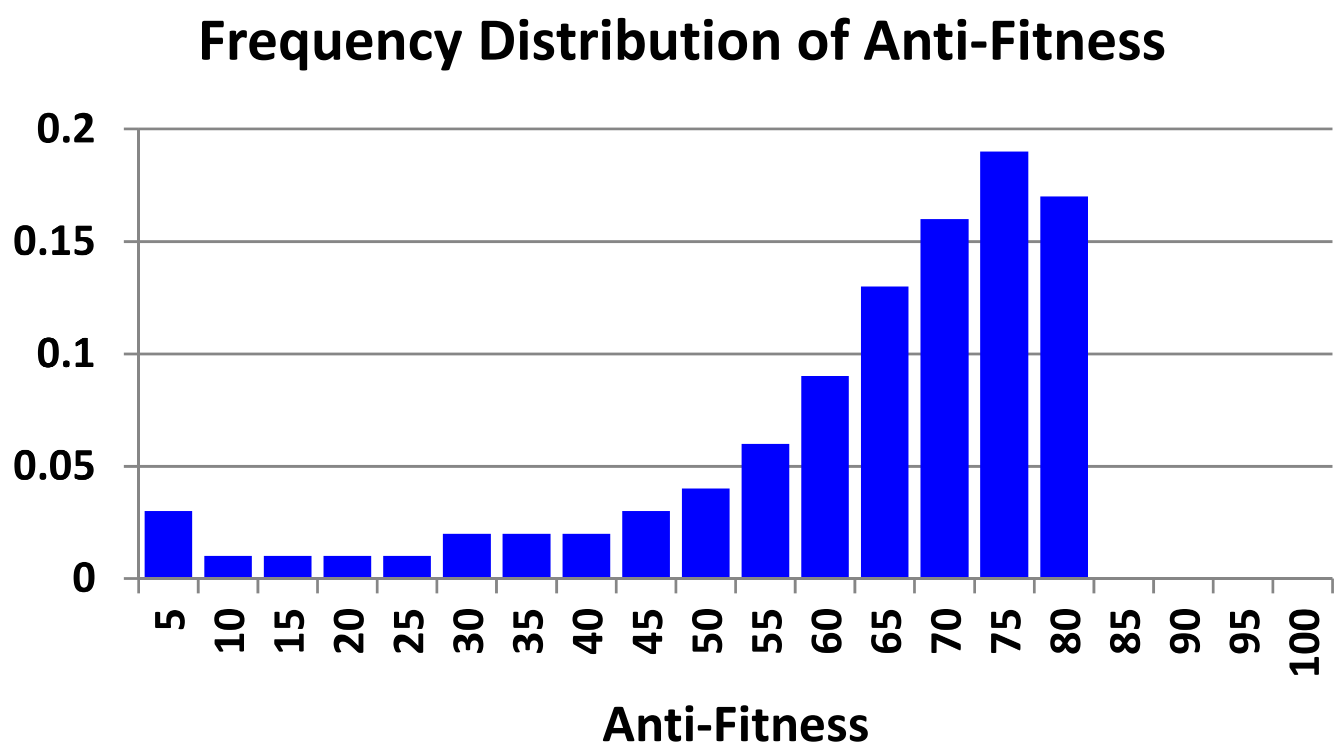






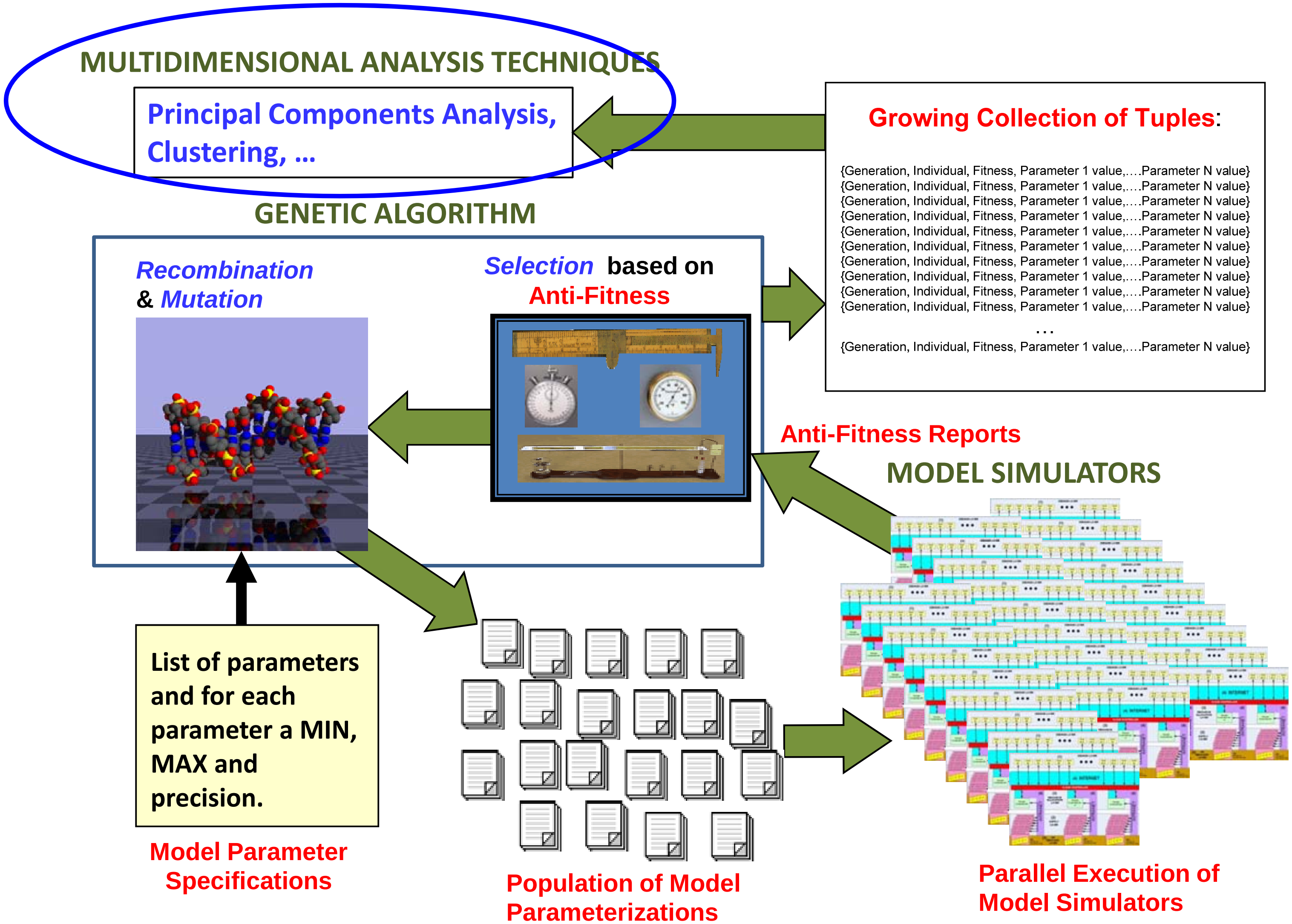
Assessment of Search Conducted by GA

(based on 10^5 scenarios, i.e., 200 individuals x 500 generations)



- **84% of scenarios exhibit anti-fitness ≥ 0.50**
- **Only 8% of scenarios are duplicate (equals elite-selection percentage)**
- **For *Koala* simulator, failure scenarios appear within first 100-200 generations**

Topic 6: Analysis Method



Differential Probability Analysis

Let ***C*** be the set of collected tuples, each containing a vector of parameter value (PV) pairs and a corresponding anti-fitness value, ***f***

Segment ***C*** into high-pass (***H***) and low-pass (***L***) subsets, where:

$$\boldsymbol{H} = \{\boldsymbol{x} \in \boldsymbol{C} \mid \boldsymbol{f_x} > 0.70\} \text{ and } \boldsymbol{L} = \{\boldsymbol{x} \in C \mid \boldsymbol{f_x} < 0.15\}$$

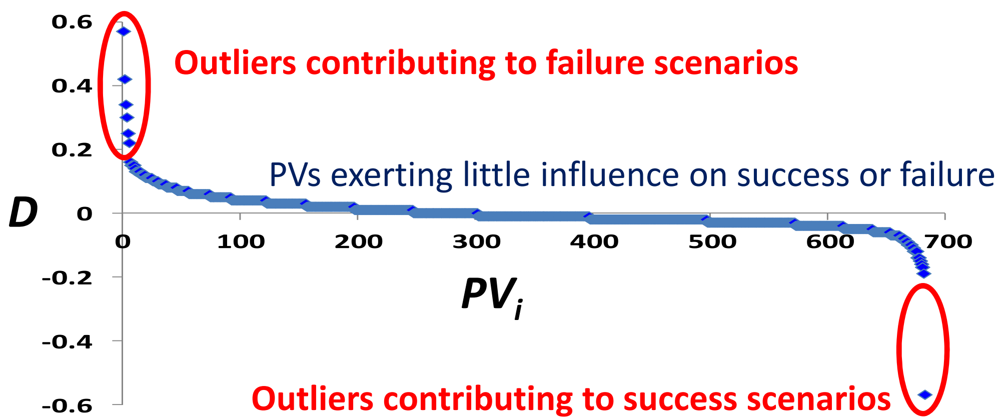
For each PV estimate the probability of occurrence in ***H*** and ***L***:

$$P(PV_i \mid \boldsymbol{f} > 0.70 = \mid PV_i \in \boldsymbol{H} \mid / \mid \boldsymbol{H} \mid \text{ and } P(PV_i \mid \boldsymbol{f} > 0.15) = \mid PV_i \in \boldsymbol{L} \mid / \mid \boldsymbol{L} \mid$$

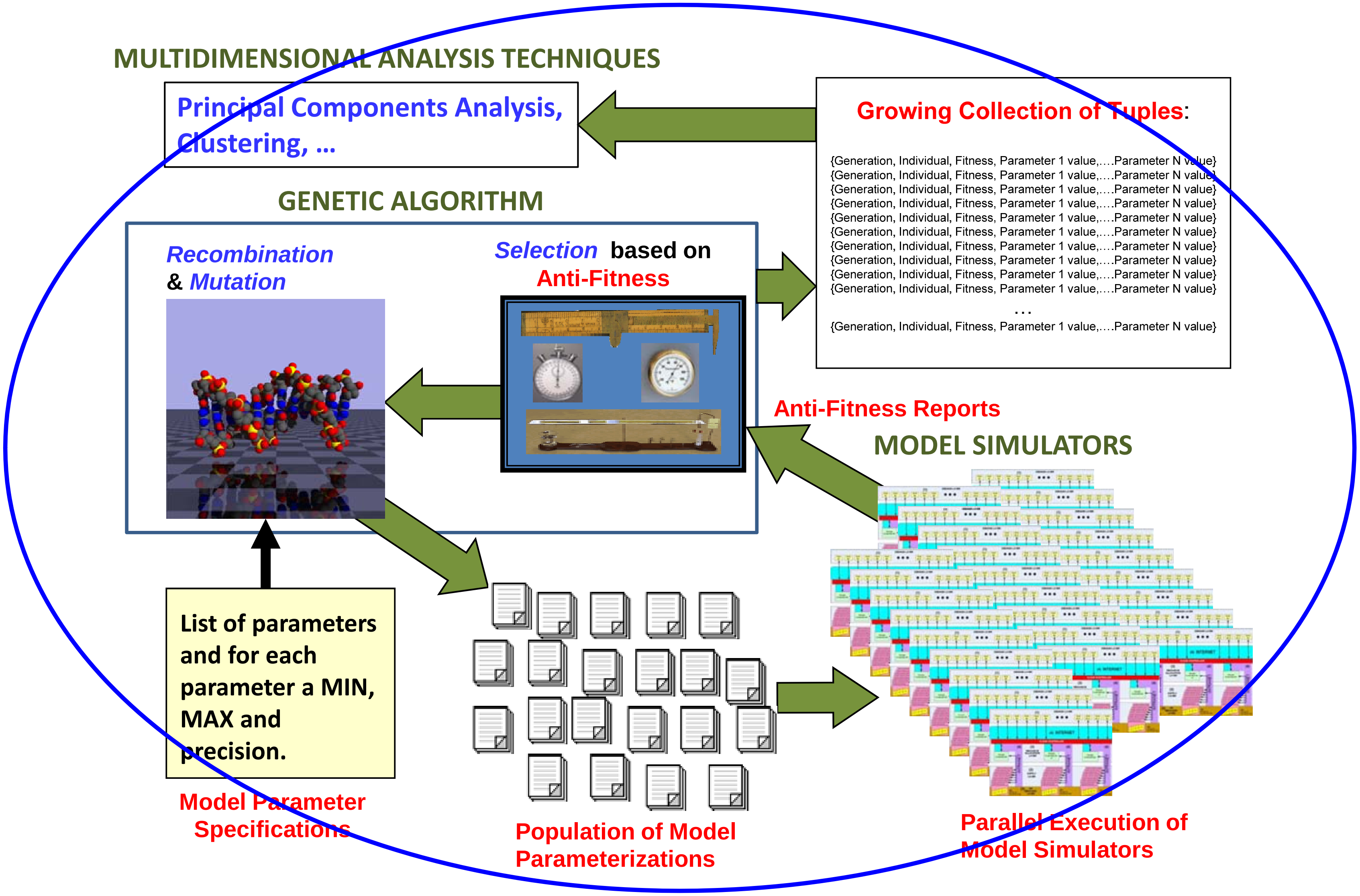
Then compute the estimated differential probability:

$$\boldsymbol{D} = P(PV_i \mid \boldsymbol{f} > 0.70) - P(PV_i \mid \boldsymbol{f} < 0.15)$$

Plot ***D*** for each PV pair

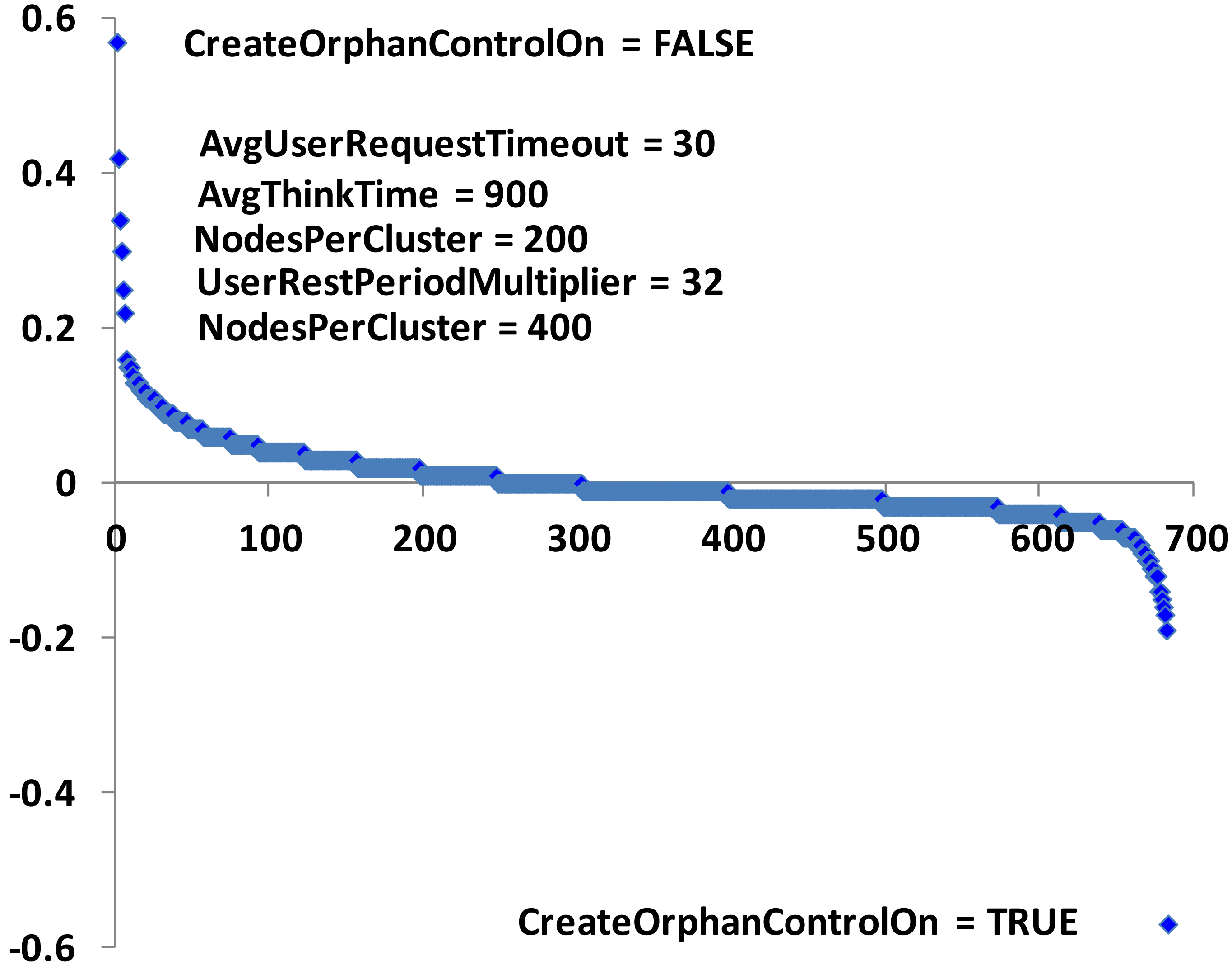


Topic 7: Results from Four GA Searches



Analysis of Results from GA Search 1 – 500 Generations

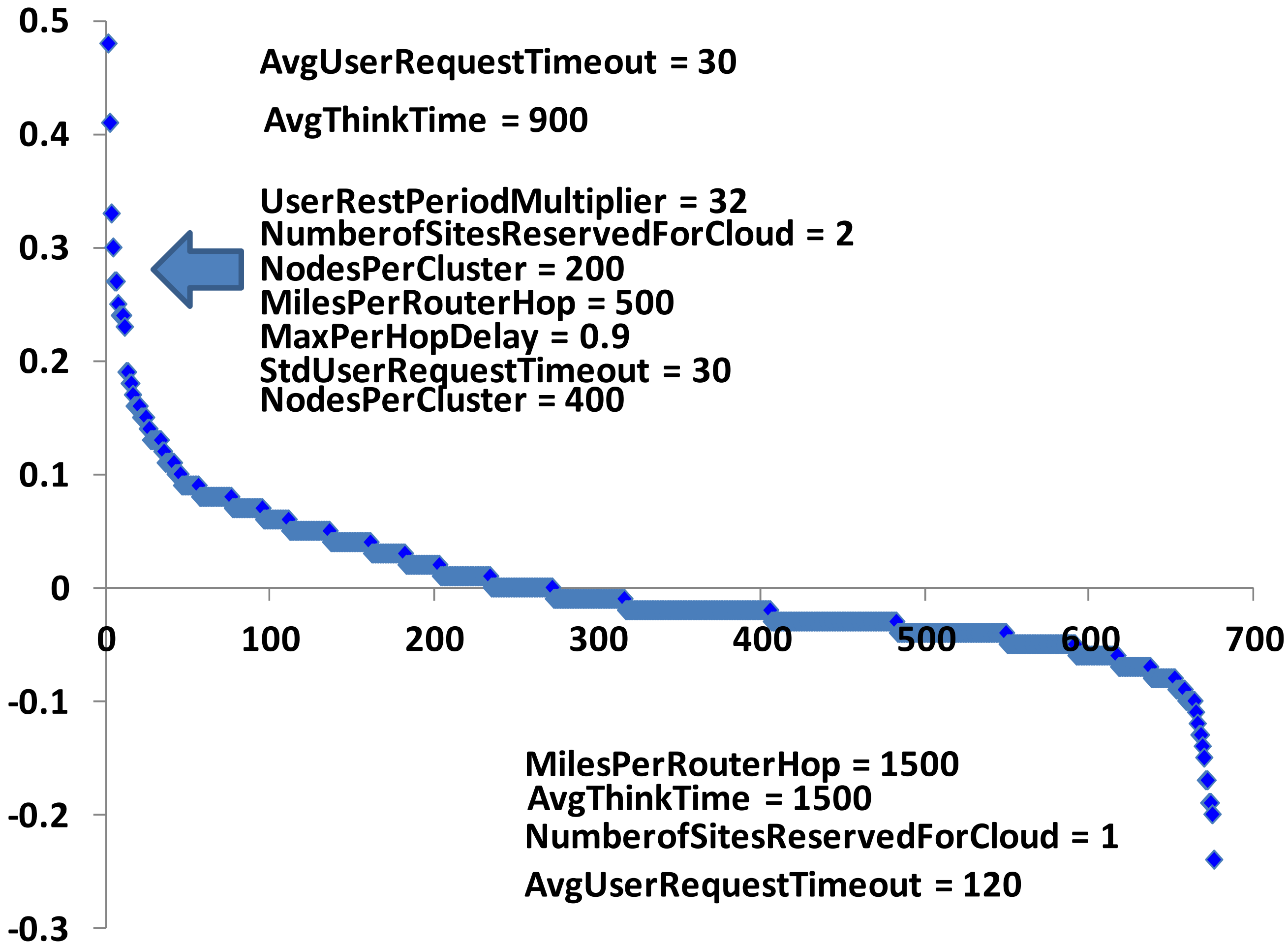
Seeking Known Failure Scenario – search duration 30 days



D (*y*-axis) for 684 PV pairs (*x*-axis) for first GA search–outlier PV pairs labeled.

Analysis of Results from GA Search 2 – 205 Generations

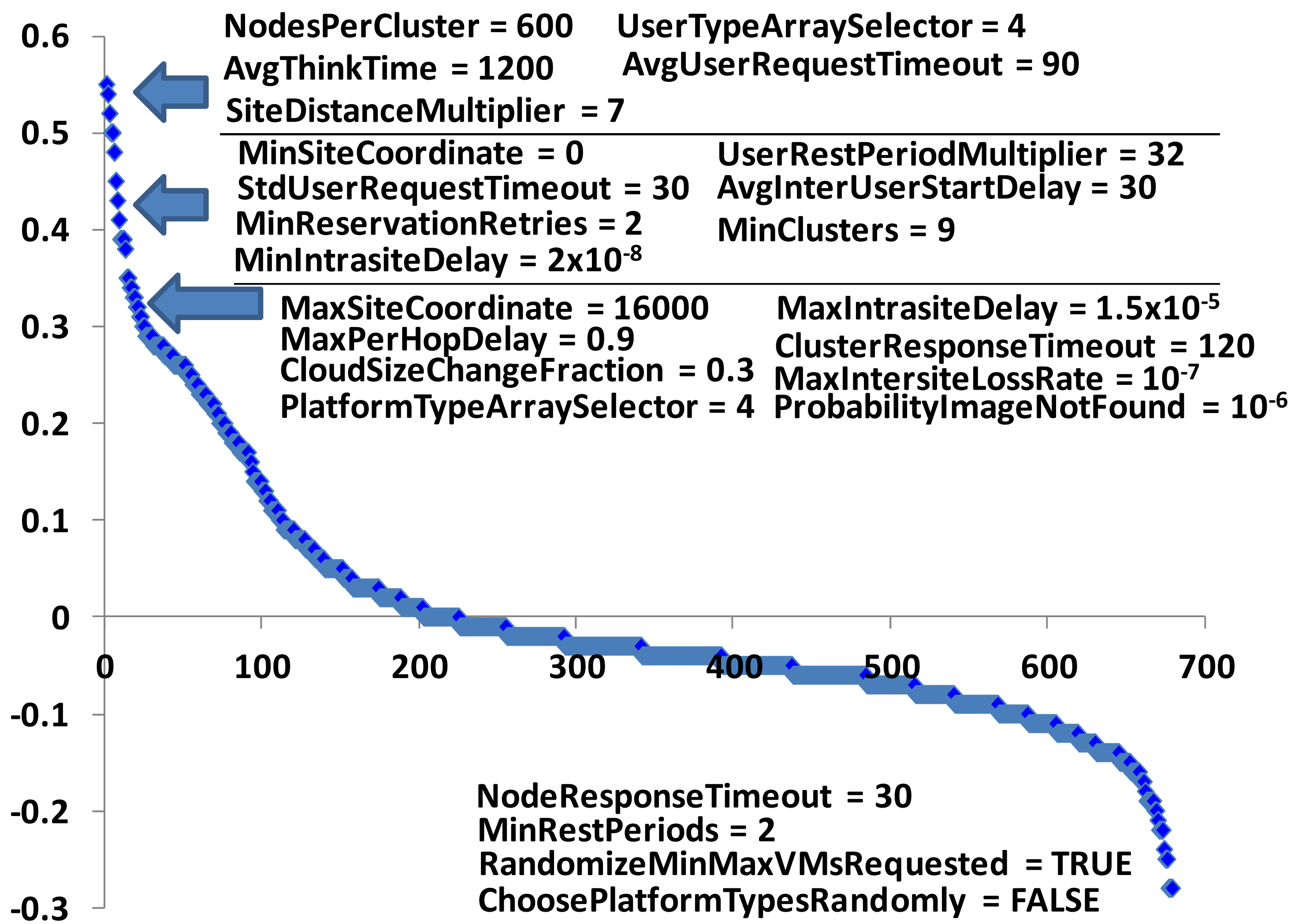
Seeking Previously Unknown Failure Scenarios – search duration 14 days



D (*y*-axis) for 677 PV pairs (*x*-axis) for first GA search–outlier PV pairs labeled.

Analysis of Results from GA Search 3 – 209 Generations

Nudging up Some Parameter Ranges and Seeking Additional Failure Scenarios - search duration 16 days



D (*y*-axis) for 680 PV pairs (*x*-axis) for first GA search–outlier PV pairs labeled.

Potential Issue Regarding Estimate of $P(PV_i \mid \boldsymbol{f} > 0.15) = |PV_i \in \boldsymbol{L}| / |\boldsymbol{L}|$

In scenarios 1-3, $|\boldsymbol{H}| \sim 10^4$, while $|\boldsymbol{L}| \sim 10^3$

To Increase $|\boldsymbol{L}|$:

- use GA to search for success scenarios
- combine those tuples with tuples collected when searching for failures

Conduct differential probability analysis on the combined tuple collection

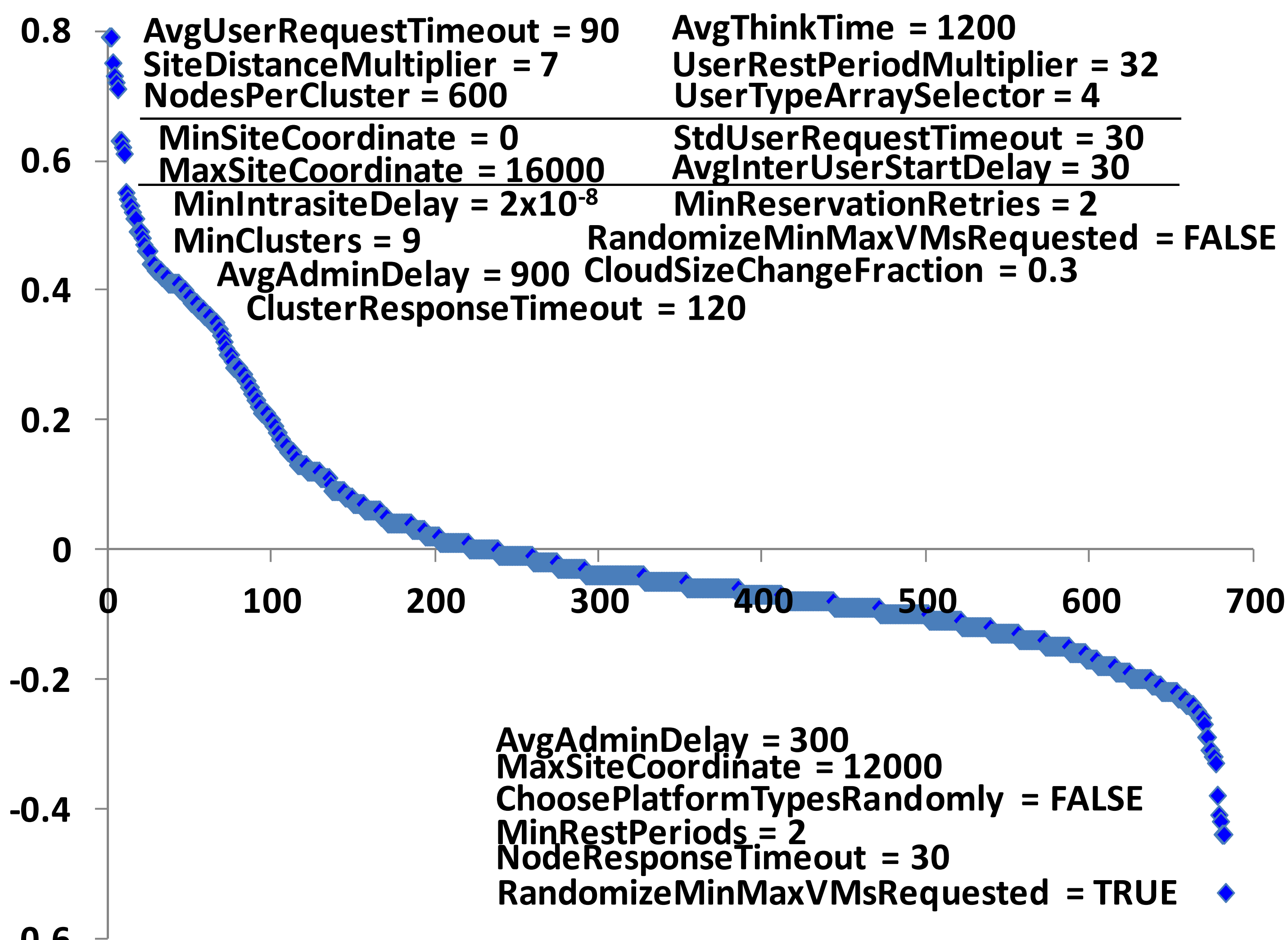
As an example, we augmented GA search 3 with a 4^{th} GA search looking for success scenarios, and combined the tuple collections from searches 3 and 4 to yield $|\boldsymbol{L}|$ = 42253 scenarios and $|\boldsymbol{H}|$ = 14601

Next slide, shows the differential probability analysis for the combined tuple collection

Analysis of Results from GA Search 4 – 209+205 Generations

Including additional PV pairs in *L* discovered by GA searching for success scenarios

- search duration 14 days



D (*y*-axis) for 683 PV pairs (*x*-axis) for first GA search–outlier PV pairs labeled.

Conclusions

SUMMARY:

- Defined a design-time method, combing GA search with simulation, to seek failure scenarios in system models
- Applied the method in a case study, seeking (and finding) a known failure scenario in an existing cloud simulator
- Iterated search to reveal previously unknown failure scenarios

FINDINGS:

- GA searches explored predominantly non-duplicative scenarios with high anti-fitness
- Uncovered evidence that GA search can reveal insights about optimal parameters settings, while simultaneously searching for failure scenarios
- GA search should be pursued only for systems with sufficient schedule time, and where failure scenarios have high cost

Future Work (I hope :-)

- Additional analysis methods need to be explored:
  - Use statistical and information-theoretic techniques to extract features from collected tuples
  - Apply clustering algorithms to suggest specific classes of failure scenarios
- Continue to explore our case study:
  - Uncover parameter subspaces where no failure scenarios can be found
  - Search under alternate definitions of anti-fitness
- Apply method to models of other complex systems:
  - Communication networks
  - Electrical grids
  - Epidemic networks
  - Network attack models
- Investigate run-time methods to provide early warning of incipient failures

QUESTIONS?

Project Web Page: http://www.nist.gov/itl/antd/emergent_behavior.cfm

Project Team:

Kevin Mills, computer scientist – kmills@nist.gov
Chris Dabrowski, computer scientist – cdabrowski@nist.gov
Jim Filliben, statistician – jfilliben@nist.gov
Sandy Ressler, information visualization specialist – sressler@nist.gov