SAND2014-2828C

Exceptional service in the national interest

CSYS 300 – COMPLEX SYSTEMS FUNDAMENTALS, METHODS & APPLICATIONS

Optimization: Heuristic Methods and Applications

Nathanael Brown
Sandia National Laboratories, New Mexico (USA)

SAND 2013-

Sandia National Laboratories is a multi-program laboratory managed and operated by Sandia Corporation, a wholly owned subsidiary of Lockheed Martin Corporation, for the U.S. Department of Energy's National Nuclear Security Administration under contract DE-AC04-94AL85000.

CSYS 300 – COMPLEX SYSTEMS FUNDAMENTALS, METHODS & APPLICATIONS

Optimization: Heuristic Methods and Applications

Outline of Presentation

- Brief Biographical Note
- Where this Section Fits in the Structure of the Complex Systems Course
- Overview of heuristic optimization techniques (Tabu Search and GA)
- Small case study: MILP versus GA for Optimal Terrorist Attack
- Real-world case study: Mitigating Bridges in NMSZ
- Summary
- Question & Answer Session

CSYS 300 – COMPLEX SYSTEMS FUNDAMENTALS, METHODS & APPLICATIONS

Optimization: Heuristic Methods and Applications

Brief Biographical Note on Nathanael Brown

- Education: BSEE (UNM), MSEE (Purdue) focused on DSP/algorithms
- Non-SNL Work Experience
  - Intel Corp (2001 – 2003): Performance tuning expert system (patent 7,043,719)
  - Tera Research (1997 – 2000): Data analysis software
  - Amtech Systems Corp (1996-1997): RF tag reader embedded software
  - American Laser Games (1993 – 1996): Video game design
  - Compaq Computer Corporation (1991 – 1992): Handwriting recognition
- SNL Work Experience
  - Investment Planning LDRD (2010 – 2013): Development of a Modeling Framework for Infrastructures in Multi-Hazard Environments
  - Resilience LDRD (2010): Measurement and Optimization of Infrastructure Resilience
  - Systems tester development (2003 – 2010): Helped design/develop a suite of testers which utilizes multiple networked subsystems to capture and analyze real-time data from a unit under test

CSYS 300 – COMPLEX SYSTEMS FUNDAMENTALS, METHODS & APPLICATIONS

Structure of the Course

Focus of this session

- Fundamentals of Complex Systems ☑
- Methods
  - Modeling Techniques ☑
  - Approaches to Examining Complex Systems ☑
- Applications
  - Examples of the use of complex systems fundamentals to solve problems ☑
  - Learning how to use complex systems modeling tools □

*Note: These approaches represent a simplified set of complex systems concepts chosen for the CSYS 300 systems lectures. Please see the initial two lectures for additional detail and expanded references.

Heuristic Methods for Optimization – Overview

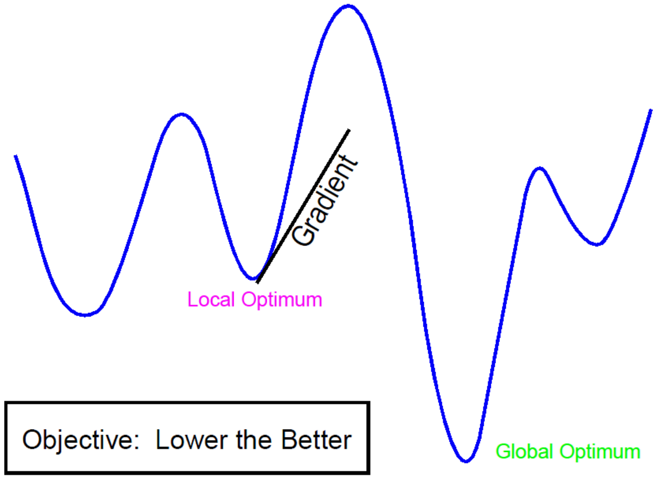
- What is a Heuristic Method (in the context of optimization)?
  - Approximate algorithms that do *not* guarantee global optimality
  - An iterative, non-deterministic algorithm used for solving combinatorial optimization problems (finding an optimal arrangement or ordering of a finite set of discrete objects)
  - Useful when solution space is non-convex and/or is dependent on multiple, possibly competing, objectives
- When should a Heuristic Method be used?
  - When standard techniques (e.g., MILP) are insufficient
  - Solution space is too large and can't be searched exhaustively
  - Objective calculation is very expensive (e.g., solution evaluation involves a complex simulation)
- Popular Heuristic Methods
  - **Simulated Annealing**, **Tabu Search**, **Genetic Algorithm**
  - Sait, S. M. and H. Youssef. 1999. "Iterative Computer Algorithms with Applications in Engineering: Solving Combinatorial Optimization Problems." Los Alamitos, CA: IEEE Computer Society.

Simulated Annealing – Overview

- General adaptive heuristic for solving combinatorial optimization problems, designed to escape local optima
- Algorithm is analogous to the physical annealing process: heating a solid to a high enough temperature to melt, then slowly cooling the liquid until it crystallizes
  - By cooling at a proper rate, proper crystal structure with perfect lattices will be created
  - The algorithm simulates the heating/cooling cycle of annealing
- Has been used to solve a variety of optimization problems
  - Traveling salesman problem (TSP)
  - Image Processing
  - Scheduling
  - Facilities layout
  - VLSI design
- Fairly easy to implement

Simulated Annealing – Algorithm Details

- From a random initial state, iteratively searches the local neighborhood for better solutions
- Starts with a heating cycle which allows for poor solutions to be accepted with a certain probability
  - Allows the escape from local optima by "hill climbing"
- The cooling cycle slowly decreases the probability of accepting "worse" solutions until only "better" solutions are accepted
- Utilized in an LDRD to determine the best recovery strategy for a damaged railway system based on: cost of repairs, duration of repairs, priority of commodity delivery
- Found to be within 1% – 2% of "true" optimal based on full enumeration



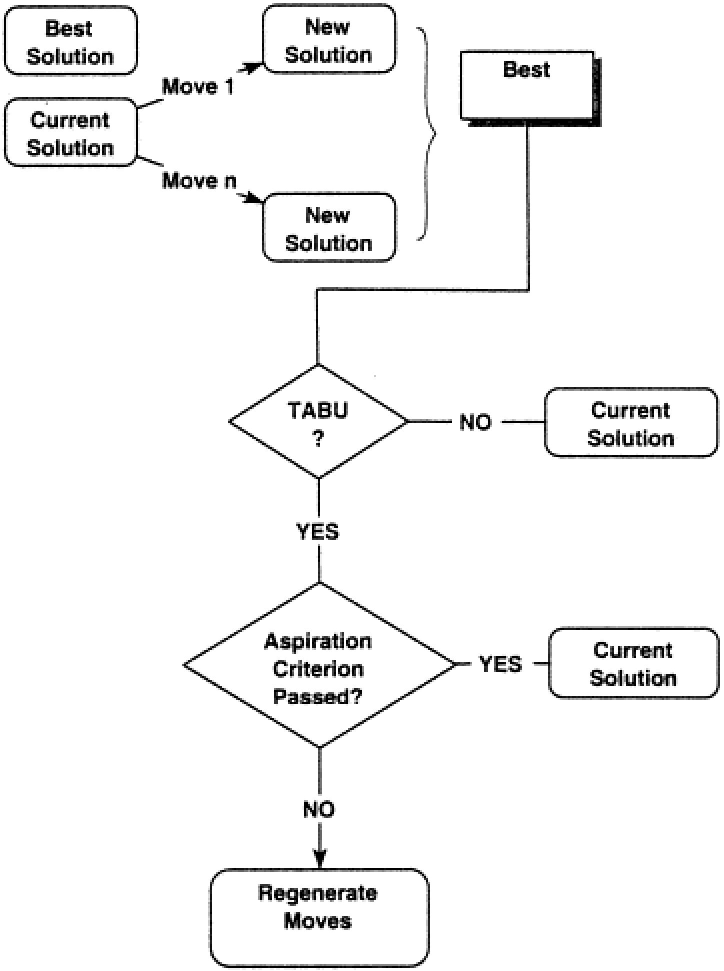
Hotrodanglican.blogspot.com

Tabu Search – Overview

- Generalization of local search used to explore very large solution spaces
- Algorithm samples different “neighborhoods” within the solution space via a sequence of “moves”
- Composed of three different memory components:
  - Short-term component prevents cycling (Tabu List)
  - Intermediate-term component is used for regional search **intensification**
  - Long-term component promotes global solution **diversification**
- Designed to escape local optima

Tabu Search – Algorithm Details

- Find "best" solution in current neighborhood by doing a local search
- Accept *best* solution as *current* if:
  - Not on the Tabu List
  - Better than current *best* solution
- If solution *is* accepted, continue searching in current neighborhood (intensification)
- If solution is *not* accepted, move to a new neighborhood (diversification)
- Repeat process a fixed number of times

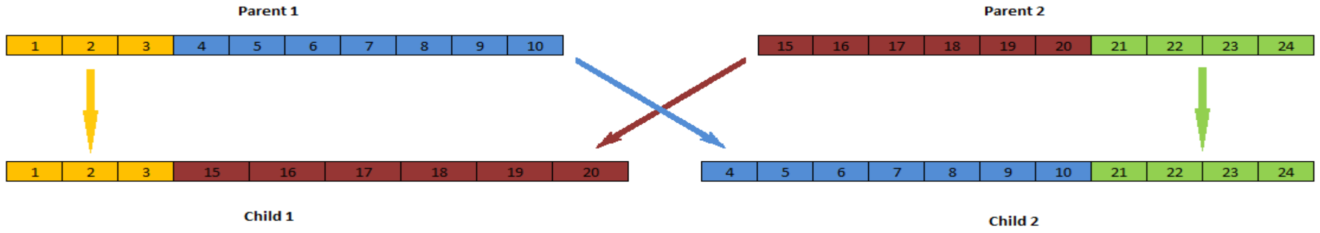


Genetic Algorithm (GA) – Overview

- Simulates the process of evolution based on the Darwinian theory of natural selection
- Individuals are represented by encoded strings (chromosomes) which mate to produce offspring with (mostly) the best characteristics of both parents
- Computationally straightforward and easy to implement
  - Requires no assumptions about the search space (e.g., continuity, existence of derivatives)
  - Only need objective function values to evaluate solution fitness
- Composed of the following steps:
  - Initialization
  - Selection
  - Crossover
  - Mutation
  - Termination

Genetic Algorithm (GA) – Algorithmic Details

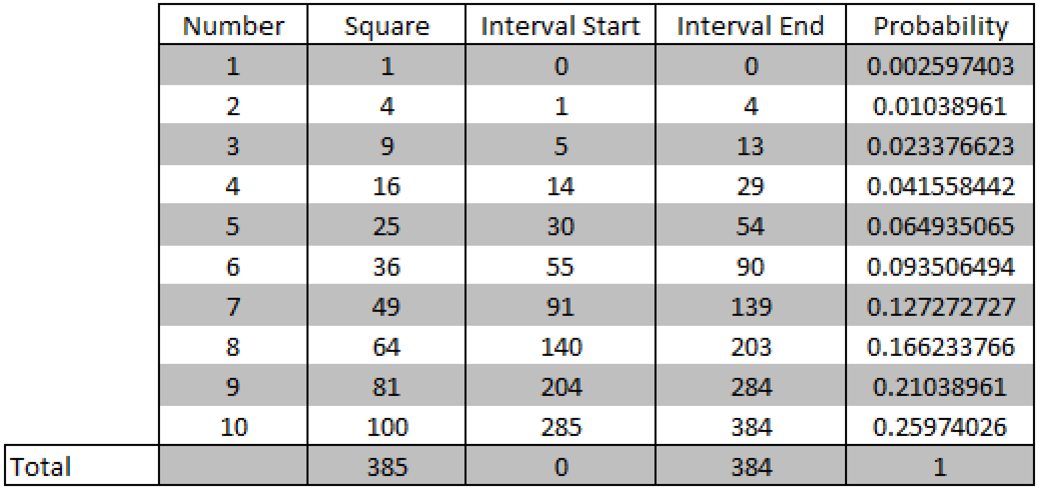
- **Initialization** – Create a population of randomly generated individuals (hundreds to thousands depending on search space)
- **Selection** – Choose a portion of the existing population to breed a new generation
  - Bias the selection such that fitter individuals are more likely to be selected
- **Crossover** – Each new individual is created from a pair of "parent" solutions by splitting and recombining the parent chromosomes



- **Mutation** – Randomly change a few genes in the child chromosome to promote diversity (on the order of 5%)
- **Termination** – Terminate after a fixed number of iterations or when there is no significant improvement

Implementing a Biased Selection

- Heuristic Algorithms often require making probabilistically-biased selections
- One technique is to assign contiguous, integer intervals to each item based on a bias factor and use a random number generator to select
- Example: 10 numbers to select from where probability of selection is proportional to the square of the number



	Number	Square	Interval Start	Interval End	Probability
	1	1	0	0	0.002597403
	2	4	1	4	0.01038961
	3	9	5	13	0.023376623
	4	16	14	29	0.041558442
	5	25	30	54	0.064935065
	6	36	55	90	0.093506494
	7	49	91	139	0.127272727
	8	64	140	203	0.166233766
	9	81	204	284	0.21038961
	10	100	285	384	0.25974026
Total		385	0	384	1

- Randomly select an integer value between 0 and 384; interval that value falls into indicates the target number

Terrorist Attack on a Hospital Network – Basic

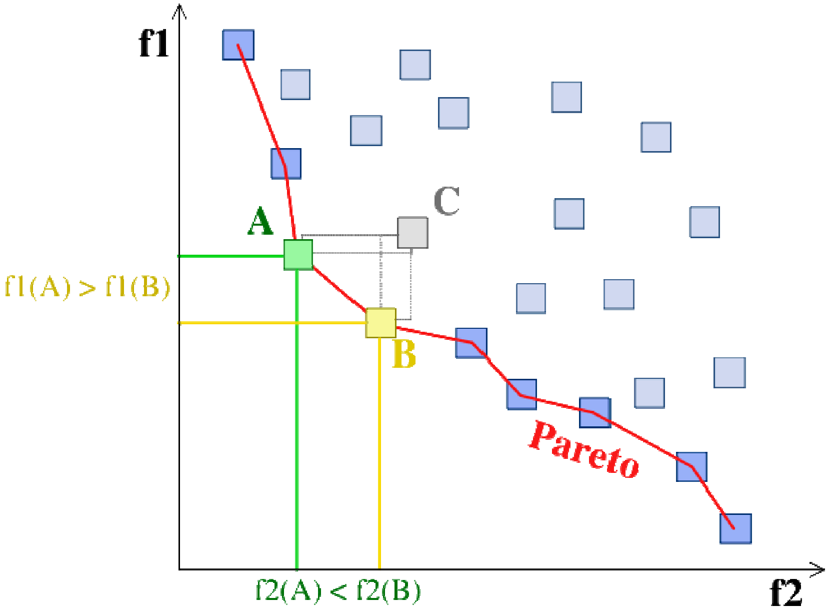
- Network consists of 20 hospitals within the Memphis area
- Terrorist has a maximum “budget” of 7 hospitals to disrupt service to the maximum number of people: 77,520 different permutations
  - 20 choose 7 binomial coefficient: $\binom{n}{k} = \frac{n!}{k!(n-k)!}$; $\binom{20}{7} = \frac{20!}{7!13!} = 77{,}520$
- Full enumeration requires 9.25 minutes for an exhaustive search
- CPLEX (MILP solver) finds optimal solution in 42 seconds
- GA solution searching 30% of solution space (102 seconds)
  - Initial population of 7752 (10% of total) with 20 crossover steps (776 per step)
  - 100 runs: 45% at optimal, 100% within 5.2% of optimal
- GA solution searching 10% of solution space (37 seconds)
  - Initial population of 775 (1% of total) with 90 crossover steps (78 per step)
  - 100 runs: 22% at optimal, 93% within 5.8% of optimal, outliers up to 17.7%
- Neither GA solution is *guaranteed* to be optimal

Terrorist Attack on a Hospital Network – Game Theoretic

- Use a “leader-follower” model to find the optimal reinforcement strategy for minimizing the worst terrorist attack.
- Assumptions
  - Same network and terrorist budget
  - Terrorist will not attack reinforced hospitals
  - Cost to reinforce a single hospital is 1 and protector can reinforce up to N-1 (19) hospitals
- Approach
  - Use CPLEX to find optimal attack given a reinforcement
  - Use a GA to determine the optimal investment strategy within solution space of size 2^{20} – 1 = 1,048,575 different permutations
  - Dual objective: hospital service impact and mitigation cost
- What is the “best” solution?
  - Could equally weight 2 objectives and get a single solution (adequate)
  - Use a Pareto Frontier to give a range of good solutions (better)

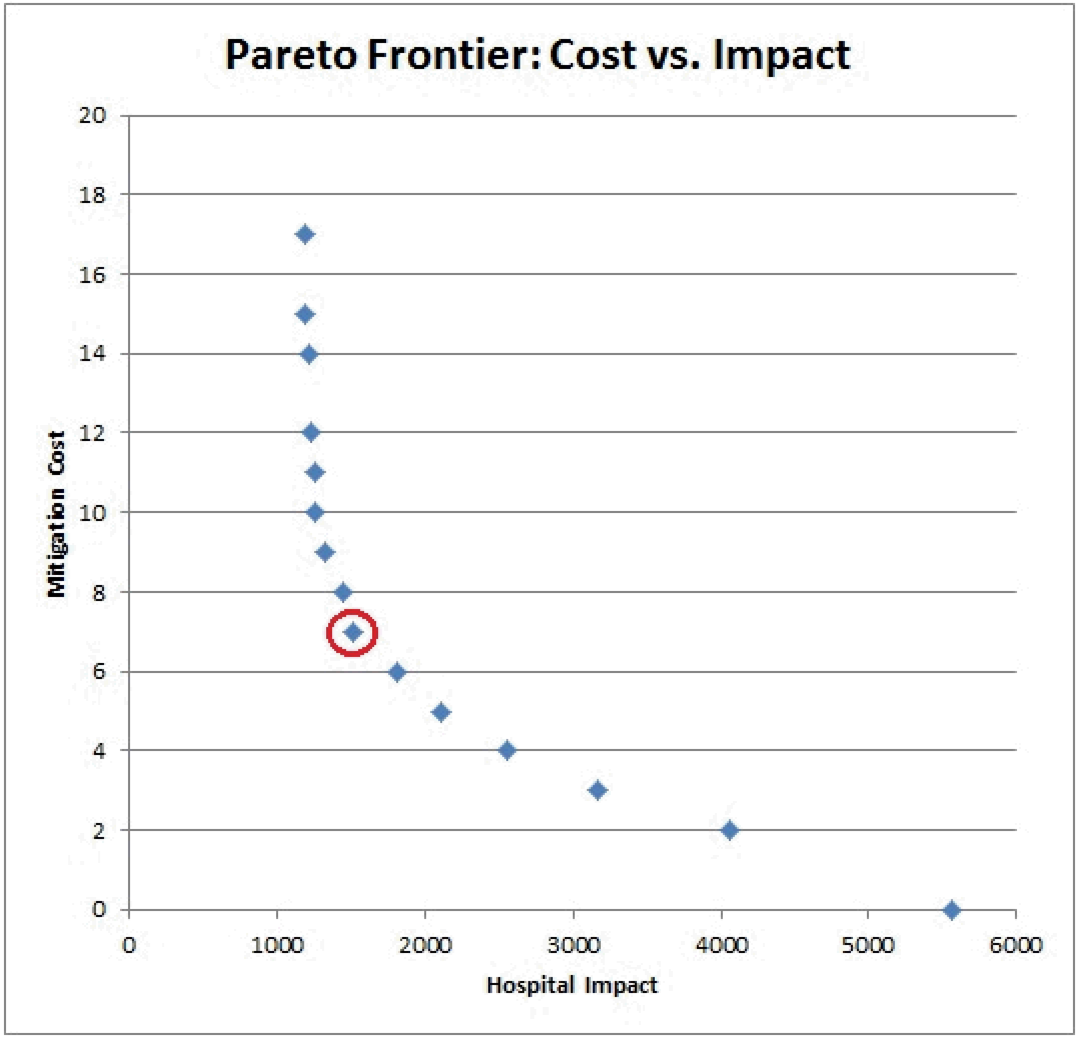
Fitness Evaluation – Pareto Frontier

- A Pareto Frontier consists of only those solutions which are *not* dominated by any other solutions
- Solution A dominates B if both conditions are true:
  - Solution A is no worse than B in all objectives
  - Solution A is strictly better than B in at least one objective
- Example: Solutions A, B and C with dual objectives f1 and f2 which must be minimized
  - Solution A does *not* dominate B because f1(A) > f1(B)
  - Solution A *dominates* C because f1(A) < f1(C) and f2(A) < f2(C)



Terrorist Attack on a Hospital Network – Pareto Frontier

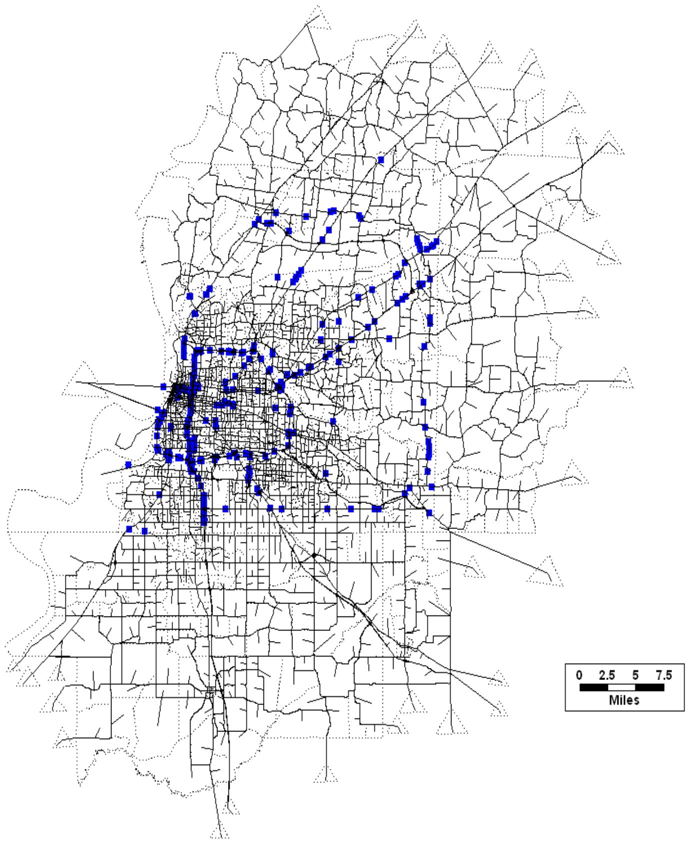
- Frontier is composed of 15 data points
- Equally weighted objectives approach would have selected mitigation cost of 7 (circled in plot)
- Chose 1000 mitigation strategies for initial population (roughly 0.1% of total solution space)
- 90 crossover steps using 100 "parent" solutions per iteration
- Solutions with cost 18 and 19 are non-Pareto



Earthquake Mitigation of a Highway Network – Problem Overview

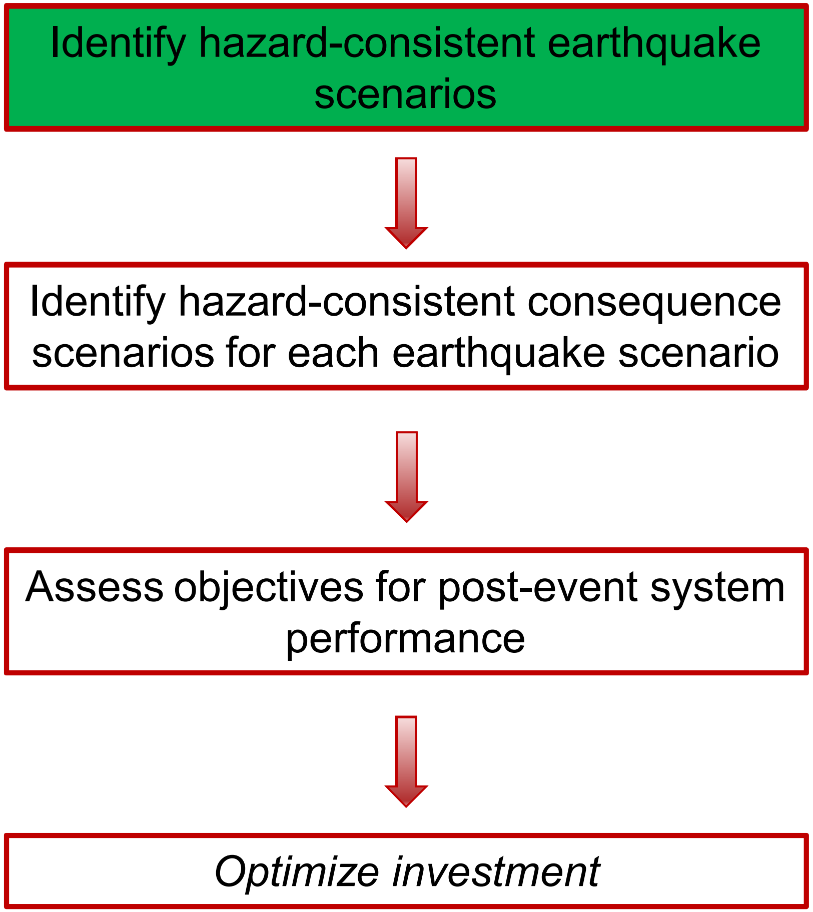
- Examine earthquake hazard in New Madrid Seismic Zone (NMSZ)
- Highway network overlaid with hospital locations
- Bridges are the most vulnerable components of the network
  - Damage affects both travel times and network connectivity
  - Investments can be made to reinforce bridges to reduce level of damage

Memphis Highway Network



Blue dots represent highway bridges (335 total)

Modeling Process



Identify hazard-consistent earthquake scenarios

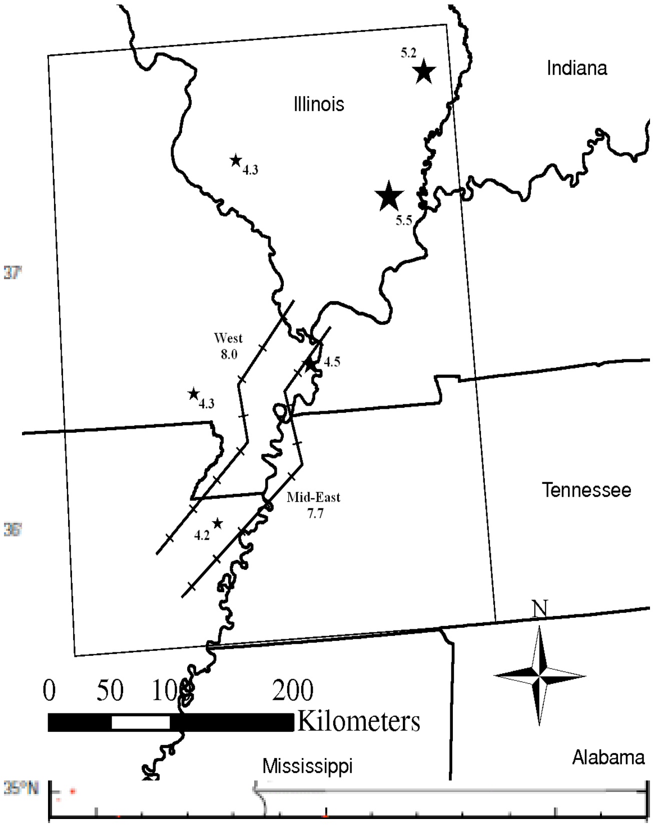
Identify hazard-consistent consequence scenarios for each earthquake scenario

Assess objectives for post-event system performance

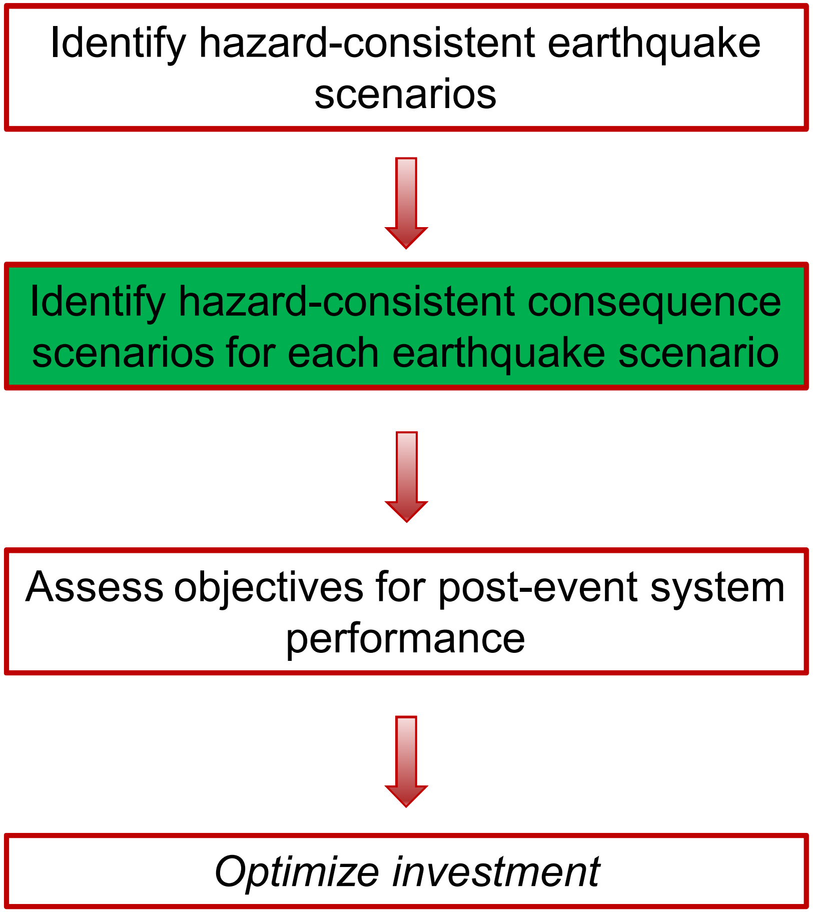
Optimize investment

Representing the Hazard

- Analysis requires a collection of earthquake scenarios
- Two sources of candidate earthquake scenarios
  - USGS has cataloged 433 historic events
  - USGS has developed 5 synthetic faults
    - We consider 4 possible magnitudes from each fault (20 total events)
- Computations preclude considering all 453 events
- Use optimization to select a representative subset of 8 events of which only 2 are significant



Modeling Process



Identify hazard-consistent earthquake scenarios

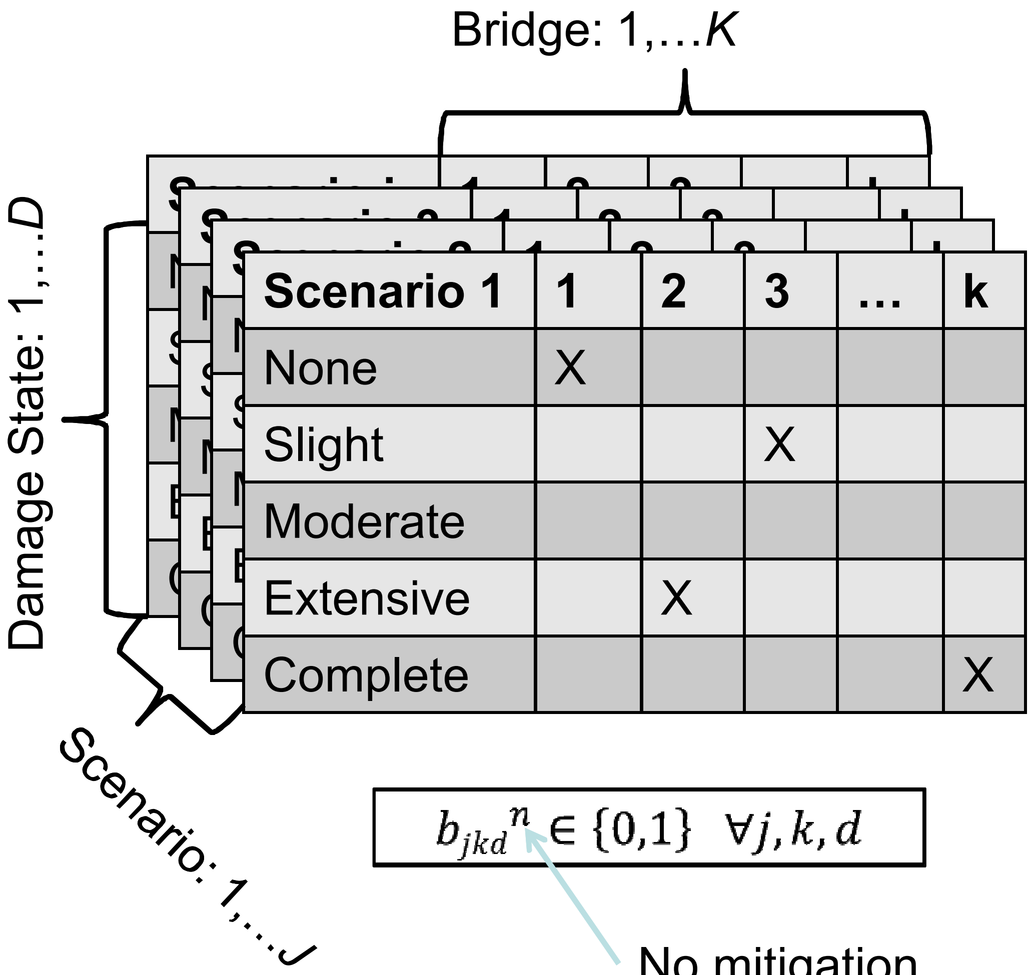
Identify hazard-consistent consequence scenarios for each earthquake scenario

Assess objectives for post-event system performance

Optimize investment

Consequence Scenarios: Before Mitigation

Scenario Outcome



Scenario 1	**1**	**2**	**3**	**…**	**k**
None	X				
Slight			X		
Moderate					
Extensive		X			
Complete					X

$$\boxed{b_{jkd}^{n} \in \{0,1\} \quad \forall j,k,d}$$

No mitigation

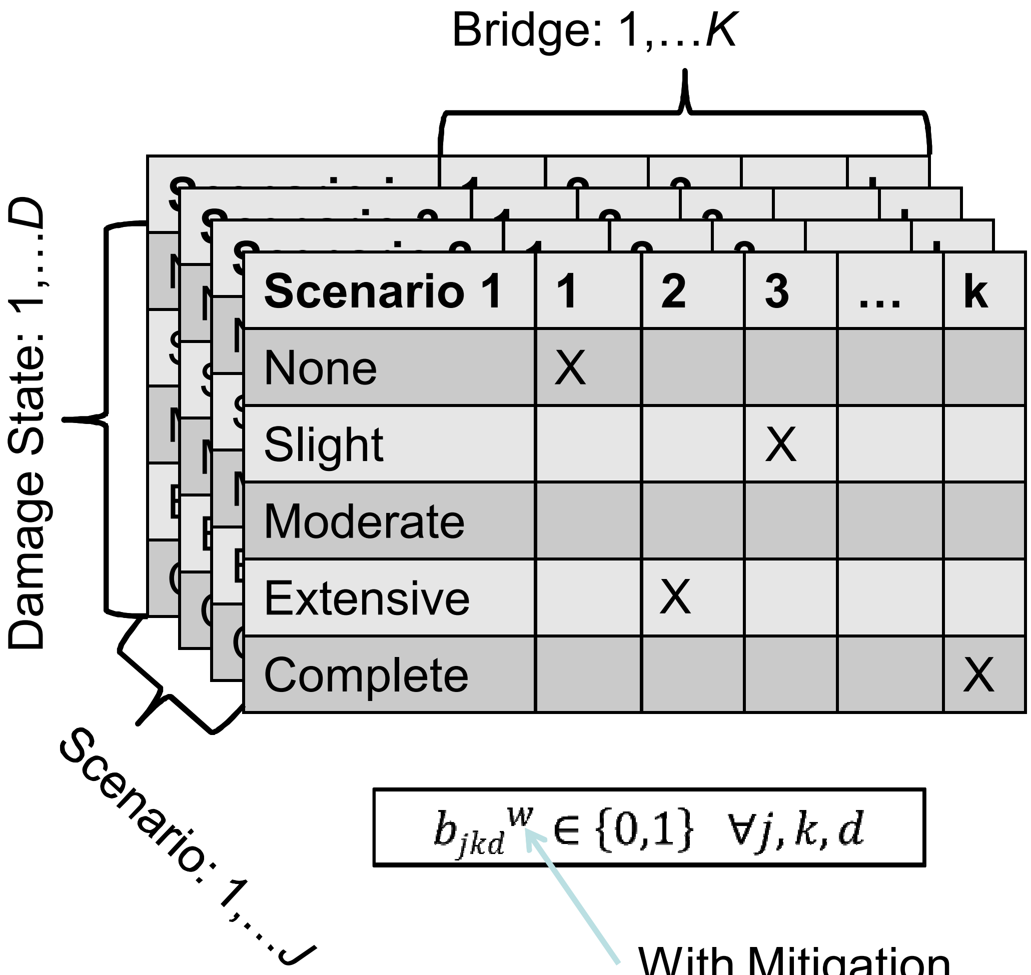
Scenario Probability

Scenario	**Probability**
1	0.3
2	0.2
3	0.15
…	
J	0.1
Total	1.0

$$\boxed{0 \le s_j \le 1}$$

Consequence Scenarios: After Mitigation

Scenario Outcome



Scenario 1	1	2	3	…	k
None	X				
Slight			X		
Moderate					
Extensive		X			
Complete					X

$$b_{jkd}^{w} \in \{0,1\} \quad \forall j,k,d$$

With Mitigation

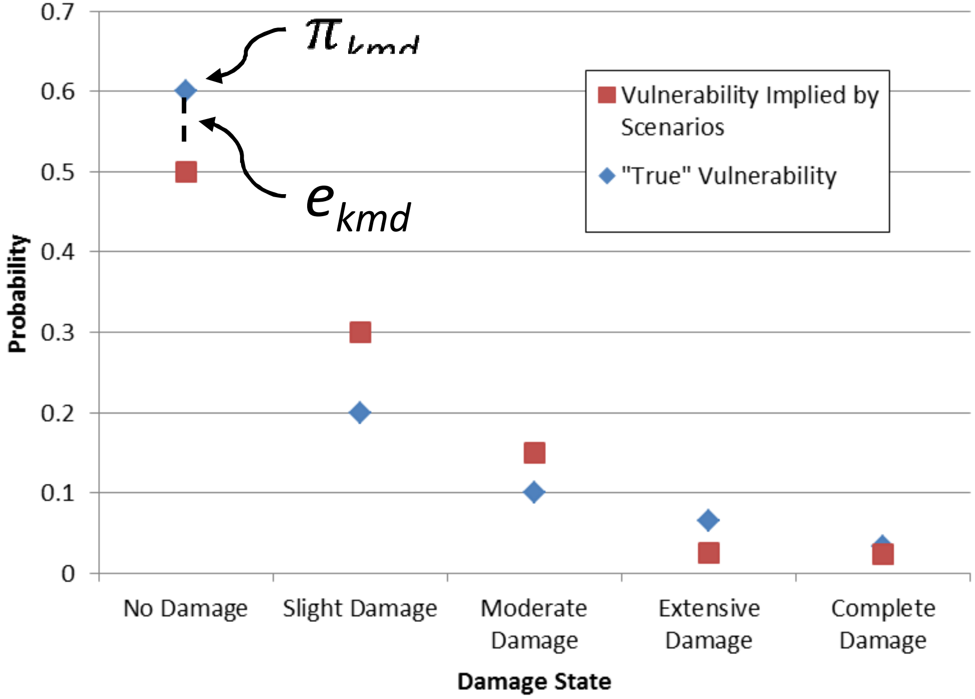
Scenario Probability

Scenario	Probability
1	0.3
2	0.2
3	0.15
…	
J	0.1
Total	1.0

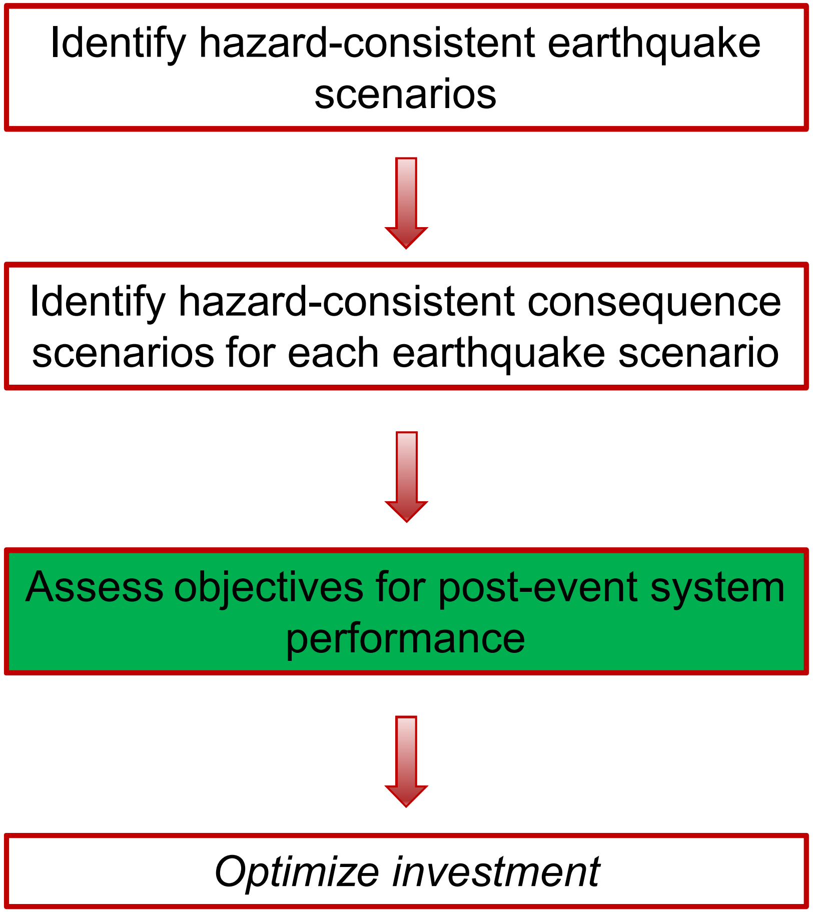
$$0 \leq s_j \leq 1$$

Approach

- Use optimization to identify a set of consequence scenarios and the probability of occurrence for each that "match" marginal distribution of damage for each component and the correlation between components



Modeling Process



Identify hazard-consistent earthquake scenarios

Identify hazard-consistent consequence scenarios for each earthquake scenario

Assess objectives for post-event system performance

Optimize investment

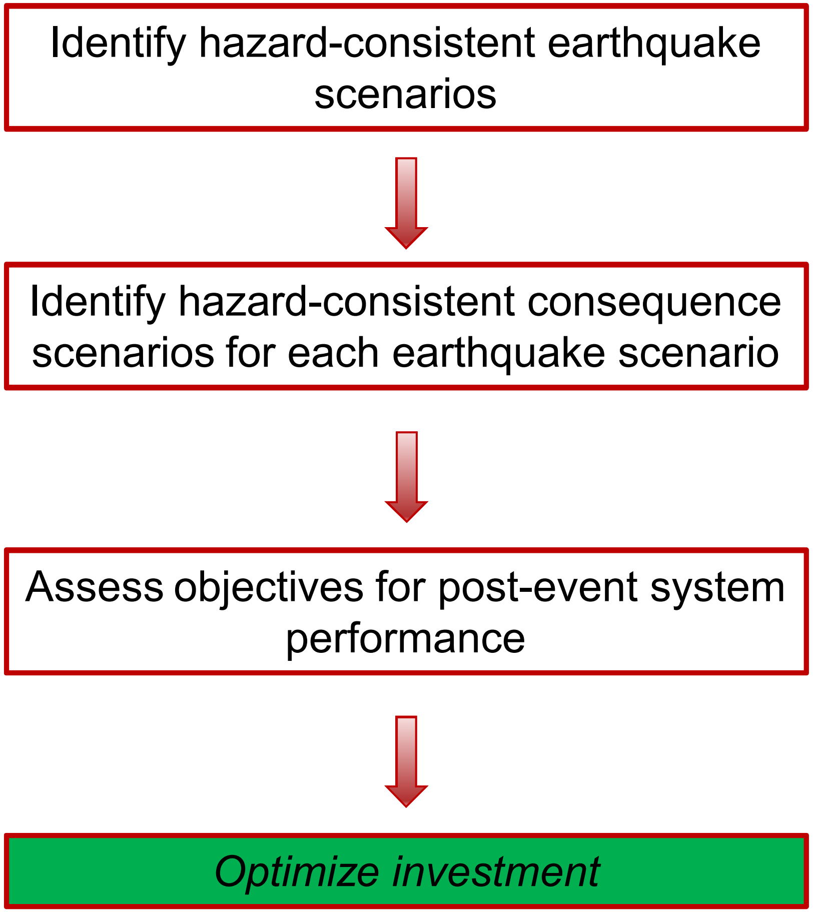
Objective Functions

- Determine optimal investment strategy which minimizes:
  - Travel Time Objective - travel time across the network
  - Hospital Connectivity Objective - travel time and disconnections from major population centers to nearby hospitals
  - Investment Cost Objective – bridge mitigation costs
- Investment cost objective is simply the sum of the mitigation costs
- The other two objectives require an assessment of the post-event network performance

Dynamic Traffic Assignment (DTA)

- Models the dynamic movement of vehicles over a network
  - An extension of Static User Equilibrium
  - Requires a collection of origin-destination tables over a defined time interval (time-varying demand)
  - Li, A., Nozick, L., Davidson, R., Brown, N., Jones, D., and Wolshon, B., “An Approximate Solution Procedure for Dynamic Traffic Assignment”, Journal of Transportation Engineering, in press.
- Comparable to industry standard but much faster
  - Validated against DynusT and measured link counts during Katrina evacuation
  - Two day Katrina evacuation runs in about 10 seconds on 8 cores versus 110 minutes for DynusT
- “Essentially, all models are wrong, but some are useful.”
  - George E. P. Box (statistician, 1919 - 2013)

Modeling Process



Identify hazard-consistent earthquake scenarios

Identify hazard-consistent consequence scenarios for each earthquake scenario

Assess objectives for post-event system performance

Optimize investment

Investment Planning Optimization: Multi-Objective 2-Stage Stochastic Program

- Use Pareto Efficiency to evaluate each solution composed of 3 objectives
  - Avoids use of arbitrary weighting method
  - Goal is to minimize the Euclidean distance to the Pareto Frontier
- Overall objective is a weighted aggregate across 40 earthquake scenarios
  - Two earthquake events (remaining 6 do not create a sufficient level of damage to highway bridges in Memphis)
  - Each earthquake has 20 scenarios where each scenario has damage states for each bridge with/without reinforcement
  - The objective for a scenario is weighted by the scenario's probability of occurrence
- Decisions variables: the set of bridges to reinforce

Computational Challenges

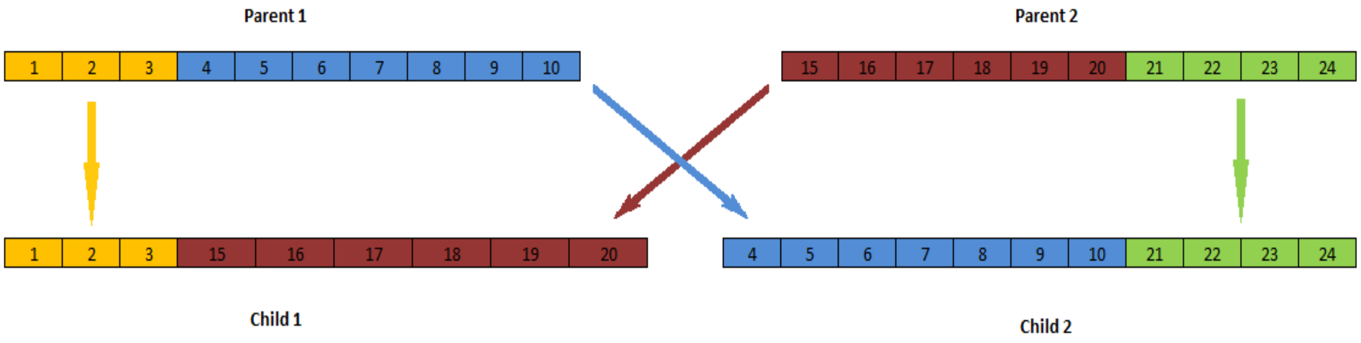
- DTA requires 11 minutes for each consequence scenario at 15 minute time intervals
- A single run requires 40 scenario evaluations at 11 minutes each = 440 minutes (7 hrs 20 min)
- There are 286 bridges that can be reinforced
  - 2^{286} different permutations ≈ 10^{86} ≈ number of particles in the universe!
- Impossible to examine all permutations

Tabu Search Implementation

- Heuristic method for finding a near optimal solution within a large search space which may contain local optima
- Each iteration examines a single mitigation strategy and the surrounding neighborhood
- Use threading to take advantage of multiple cores
- Scenario differencing
  - Recompute values only for those scenarios that change due to a bridge reinforcement
- Subsample the evaluation space
  - Sample top 10 (25%) of earthquake scenarios based on greatest number of bridges impacted
  - Bias bridge selection towards those that impact the most scenarios
  - Run DTA at 1 hour time intervals
- Use parallel processing to cover more of the solution space

Parallel Processing

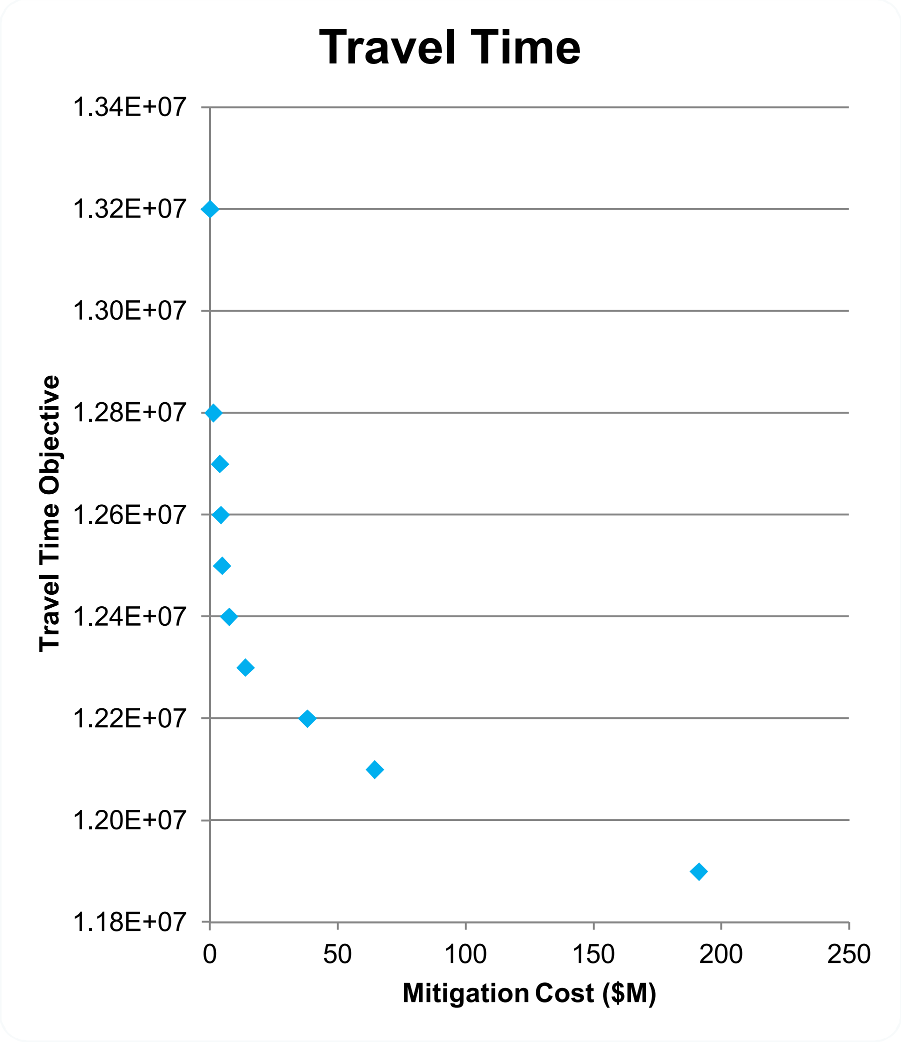
- Spawned 100 processes across a collection of compute nodes using RMI (Remote Method Invocation) and MPI (Message Passing Interface)
- Each instance starts with a random bridge mitigation strategy
- Best solution from each instance sent back to master which generates a new set of starting points via genetic crossover

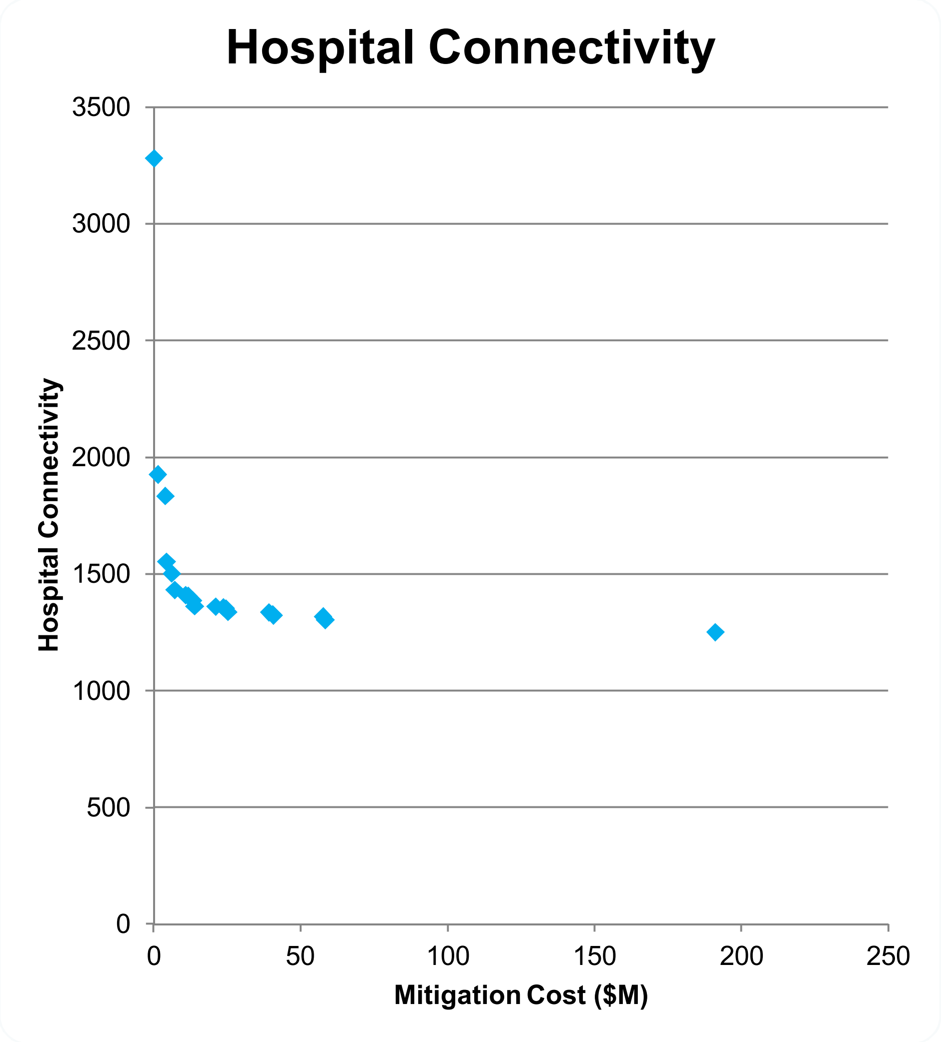


- Add diversification by performing variable-rate genetic mutation
  - Used a *similarity ratio* to determine the rate of mutation
  - Ratio of bridges that are *usually* reinforced/unreinforced to total number of bridges across current collection of solutions

Pareto Frontier (2D)

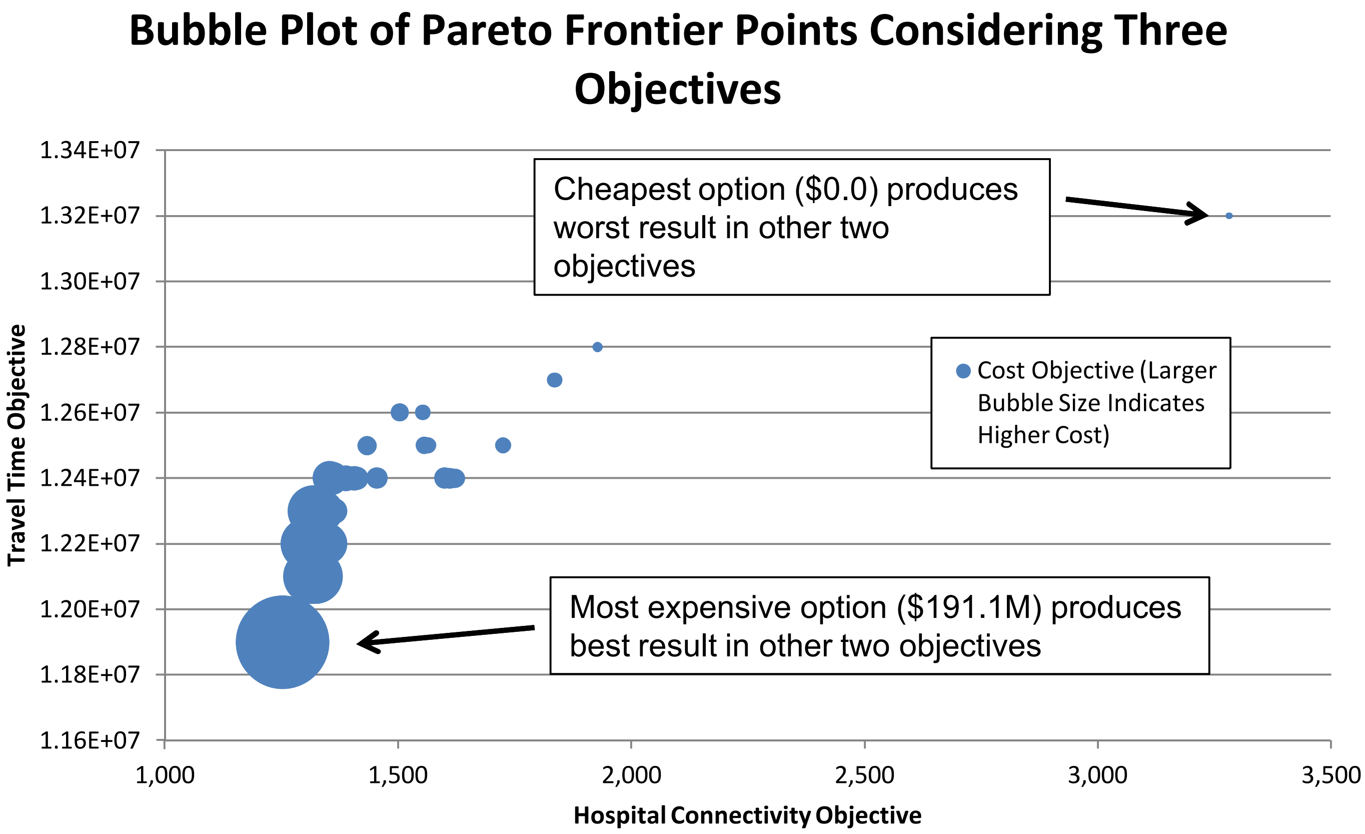
- Pareto frontier when only 2 of the 3 objectives are examined





Pareto Frontier (3D)

- Tradeoffs across all three objectives can be explored
- Relatively small tradeoffs between travel time and hospital connectivity compared to mitigation cost



Cheapest option ($0.0) produces worst result in other two objectives

Most expensive option ($191.1M) produces best result in other two objectives

CSYS 300 – COMPLEX SYSTEMS FUNDAMENTALS, METHODS & APPLICATIONS

Optimization: Heuristic Methods and Applications

QUESTIONS & ANSWERS

Nathanael Brown

Department 06131

Sandia National Laboratories

Albuquerque NM 87185-1188

njbrown@sandia.gov