Genetic Algorithms for the Graduate TA Assignment Problem

Jesse Hostetler

CSE 421 Spring 2009

Outline

- The Graduate TA Assignment Problem (GTAAP)
- Formulating GTAAP as a Weighted CSP
- Genetic Algorithm Approach to Solution
- Results
- Future Work

The Graduate TA Assignment Problem

- Definition:
  - Given:
    - A set of teaching tasks *T*
    - A pool of teaching assistants *A*
    - A set of constraints restricting the assignments of members of *A* to members of *T*
    - A set of relations $P : T \times A \rightarrow \{0,1,2,3,4,5\}$ describing the *preference* of each TA for each task
- Solution: A set of assignments that satisfies all constraints
- Ideal solution: A solution that maximizes total TA preference for assigned courses

Outline

- ✓ The Graduate TA Assignment Problem (GTAAP)
- ➢ Formulating GTAAP as a Weighted CSP
- Genetic Algorithm Approach to Solution
- Results
- Future Work

Formulating GTAAP as a Weighted CSP

- Weighted CSP (WCSP), defined:
  - Instead of *constraints*, we have *cost functions*
  - Objective: minimize total cost
  - "Hard" constraint violation → cost of "infinity"
- WCSP is a *combinatorial optimization* problem

Formulating GTAAP as a Weighted CSP

- Mapping GTAAP to WCSP:
  - Domain: The set of TAs plus the special value "unassigned"
  - Variables: The teaching tasks for which TAs are needed
  - Constraints: Both "hard" and "soft"…

Formulating GTAAP as a Weighted CSP

- Hard constraints:
  - Mutex: Two teaching tasks cannot have the same TA
    - Example: Course A and Course B occur at the same time
  - Overlap: The task cannot during a course for which the TA is registered as a student
  - Taking Course: The TA may not be enrolled in a course associated with the task
    - Example: TA enrolled in Course C cannot be the grader for Course C
  - Capacity: The TA's workload may not exceed the total number of hours for which the TA was hired to work
  - Certification: The TA assigned to the task must have ITA certification

Formulating GTAAP as a Weighted CSP

- Soft constraints:
  - Preference: Each TA rates their preference for each task from 0 to 5, with 5 being highest
    - Because WCSP is about minimizing cost, we remap preference:

$$remapped\_pref = \begin{cases} 5 - old\_pref & \text{if } old\_pref \neq 0 \\ \text{“infinity”} & \text{if } old\_pref = 0 \end{cases}$$

    - A preference of 0 is treated as a hard constraint (i.e. a cost of “infinity”)

Outline

- ✓ The Graduate TA Assignment Problem (GTAAP)
- ✓ Formulating GTAAP as a Weighted CSP
- ➢ Genetic Algorithm Approach
- Results
- Future Work

Genetic Algorithm Approach

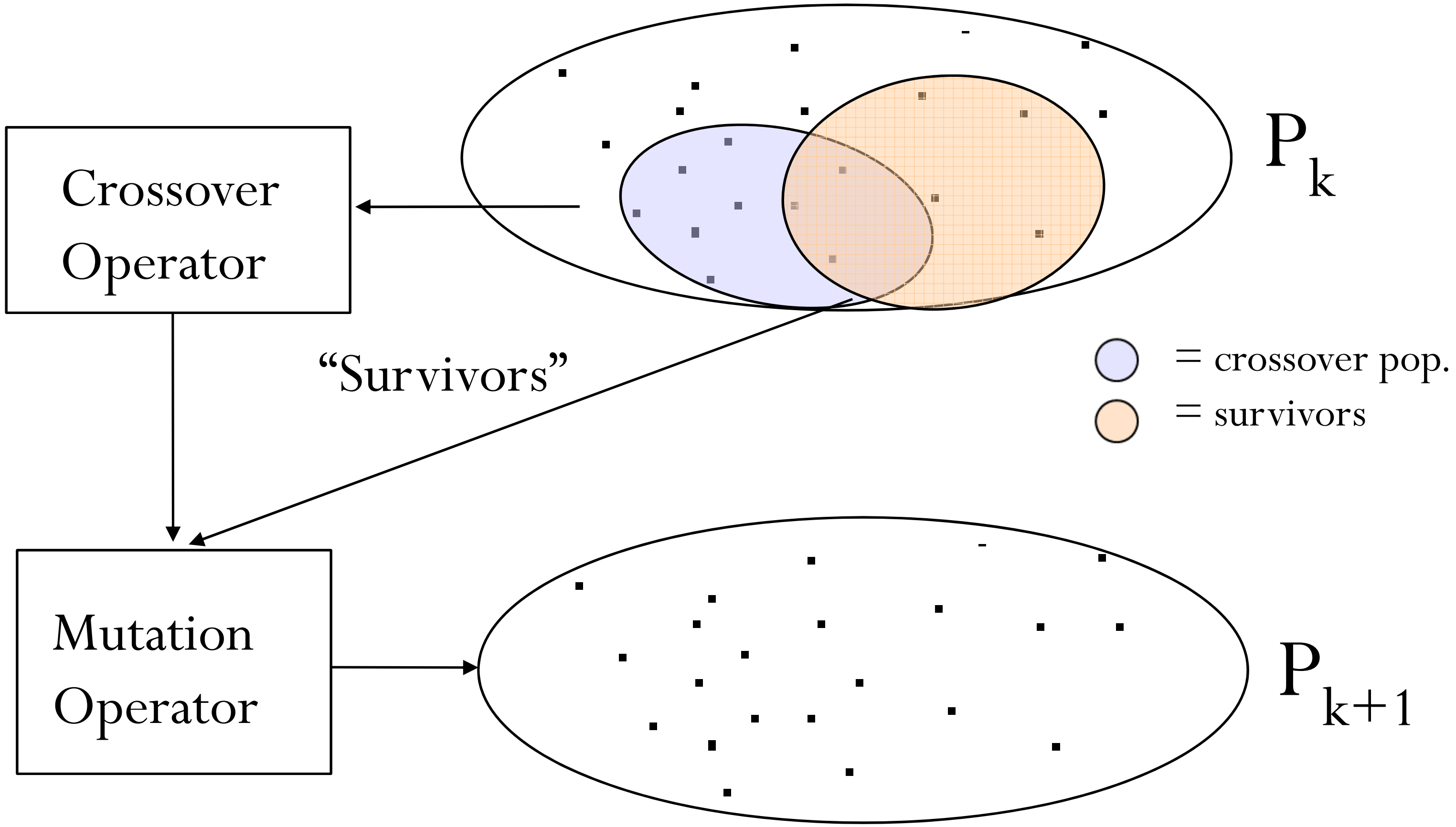
- How GAs optimize:
  - Exploit evolution to search the problem space
  - Successive generations of solutions inherit traits from their best predecessors
- Advantages:
  - Versatile - can optimize anything as long as appropriate genetic operators can be defined
    - Examples: TSP tours, neural network weights
  - Easy to parallelize

Genetic Algorithm Approach

- Components of a genetic algorithm:
  - Fitness function – A way of quantifying how “good” a solution is
  - Selection operator – Forms a new population from the current population such that individuals with a higher fitness are more likely to be chosen
  - Crossover operator – Combines traits of two individuals to form a new individual
  - Mutation operator – Changes the genome of a single individual

Genetic Algorithm Approach

- Generalized genetic algorithm



Genetic Algorithm Approach

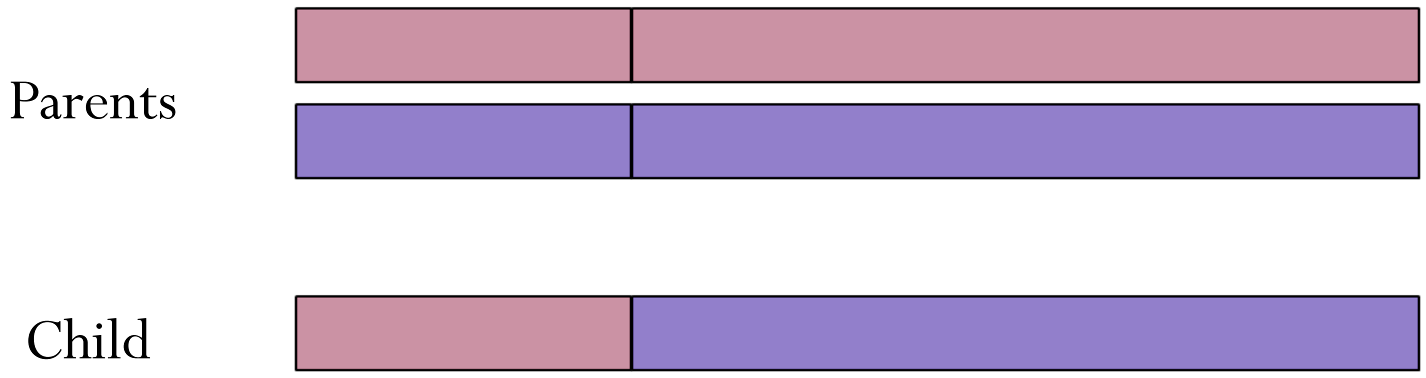
- Fitness function:
  - Must account for:
    - Hard constraint violations
    - Unassigned tasks
    - TA preference
  - In terms of "badness," assume:
    - (hard violation) > (unassigned task) > (low TA preference)
    - Violating one hard constraint is worse than leaving all tasks unassigned
    - Leaving one task unassigned is worse than assigning all TAs to courses with preference 1
  - Solution:
    - Treat "unassigned" as a TA with preference "cost" one greater than maximum possible for normal TAs.
    - Treat cost of "infinity" as one greater than cost of leaving all tasks unassigned.
    - Fitness = 1 / (total cost)

Genetic Algorithm Approach

- Selection operator:
  - Tournament selection:
    - Choose a pair of individuals uniformly at random without replacement.
    - Select the most-fit with probability p and the least-fit with probability $1-p$
  - Advantages over other methods:
    - Insensitive to magnitude of fitness difference
    - Preserves more variability in population than i.e. Roulette selection

Genetic Algorithm Approach

- Crossover operator:
  - Point crossover – Given two parents A and B, choose a point in the genome. Then, the child's genome consists of parent A's genome before selected point concatenated with parent B's genome after selected point.



Genetic Algorithm Approach

- Mutation operator:
  - "Value change" mutation – Choose a variable in the genome uniformly at random and replace its value with a different value chosen uniformly at random.
  - Before:
    - 1 5 18 2 6 7 **19** 20 0 13 24 …
  - After:
    - 1 5 18 2 6 7 **4** 20 0 13 24 …

Outline

- ✓ The Graduate TA Assignment Problem (GTAAP)
- ✓ Formulating GTAAP as a Weighted CSP
- ✓ Genetic Algorithm Approach
- ➢ Results
- Future Work

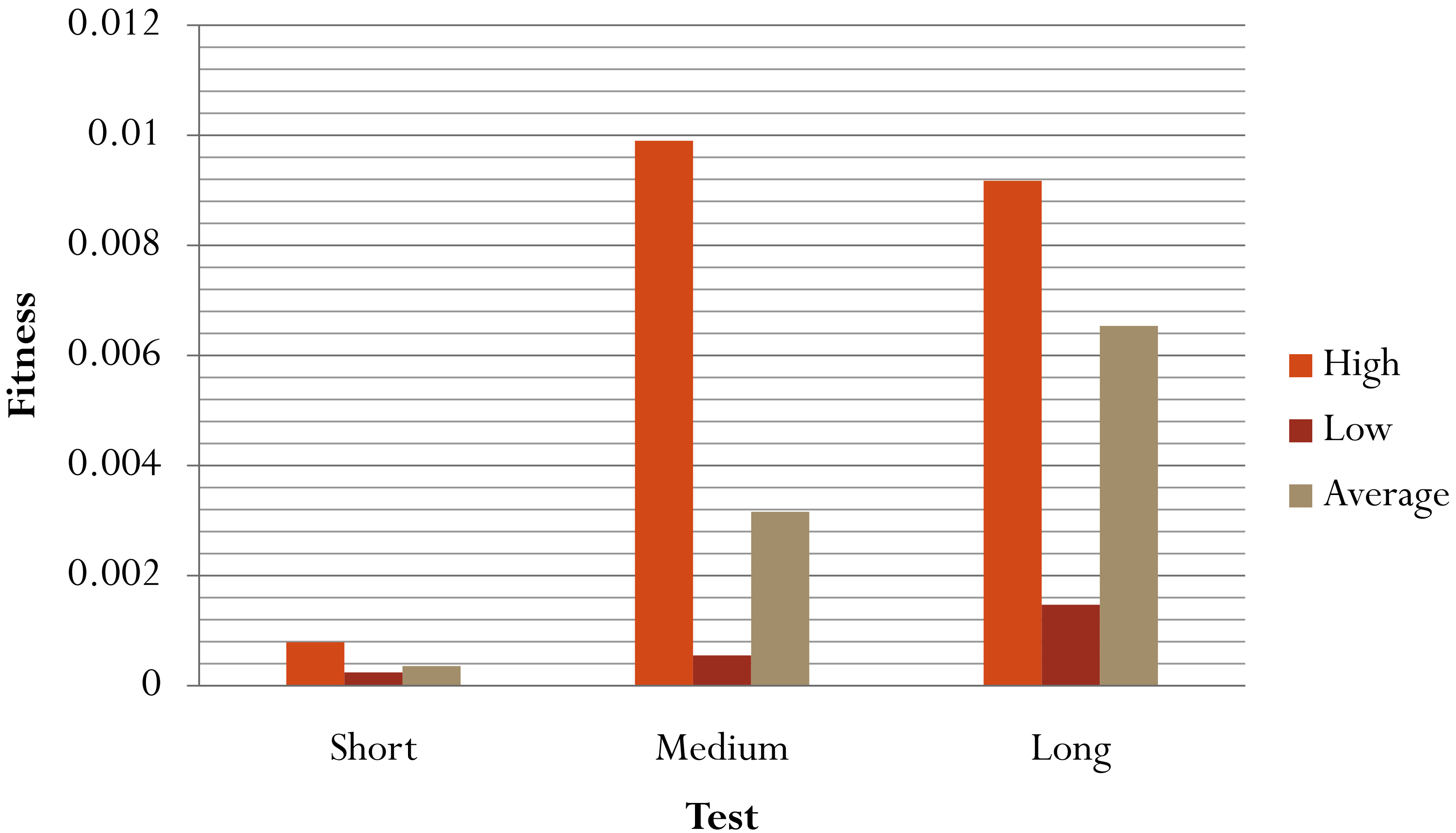
Results

- Methods:
  - Dataset: Spring 2009 CSE GTA Assignment Problem
    - 27 TAs plus 1 "unassigned"
    - 57 teaching tasks
    - 245 constraints
  - Algorithm parameters: selected based on prior experience
    - Crossover rate: 0.7
    - Mutation rate: 0.05
    - Tournament selection p: 1.0
  - Termination:
    - Specify maximum number of fitness function evaluations
    - Algorithm terminates after fully completing the generation that put it over the limit

Results

- “Short” tests
  - N = 128
  - Parameters: population size = 100, fitness evaluation limit = 1000
- “Medium” tests
  - N = 128
  - Parameters: pop. size = 100, fitness evaluation limit = 10000
- “Long” tests
  - N = 80
  - Parameters: pop. size = 1000, fitness evaluation limit = 50000, “die-off rate” = 0.02

Results



0.012
0.01
0.008
0.006
0.004
0.002
0
Fitness
Short
Medium
Long
Test
High
Low
Average

Results

Test	Runs	Solutions Found
Short	128	0
Medium	128	3
Long	80	14

Outline

- ✓ The Graduate TA Assignment Problem (GTAAP)
- ✓ Formulating GTAAP as a Weighted CSP
- ✓ Genetic Algorithm Approach
- ✓ Results
- ➢ Future Work

Future Work

- Comparison with other search techniques:
  - Simulated Annealing
  - Greedy Local Search
- New Mutation Operators
  - Can use any local search technique

Questions?