# Stochastic Sampling Search

Vincent A. Cicirello, Ph.D.
Professor of Computer Science
cicirelv@stockton.edu https://www.cicirello.org/

1

Lesson 1

## ITERATIVE SAMPLING

2

## Problem Types

- We'll focus again on optimization problems.
  - See notes on local search for more details.
- More specifically, we'll focus on optimization problems with the following two properties:
  - It is easy to generate **some** solution to the problem.
    - E.g., it is easy to generate some tour of the cities of a TSP.
  - It is computationally intractable to find the optimal solution to the problem.
    - E.g., the TSP is an NP Hard problem (i.e., no polynomial time algorithm is available, and none is likely to exist).

3

## Stochastic Sampling Search

- Stochastic Sampling Search is a class of search algorithms that randomly sample the space of possible solutions to the problem.
- We'll look at:
  - Iterative Sampling [Langley]
  - Heuristic Biased Stochastic Sampling [Bresina]
  - Value Biased Stochastic Sampling [Cicirello and Smith]
  - Heuristic Equivalency [Gomes et al]

4

## Iterative Sampling [Langley]

- Start with an empty solution.
- Iteratively add elements to the solution growing a partial solution into a complete solution.
- All decisions are made uniformly at random from among alternatives.
- Repeat entire process N times and keep the best of the solutions.

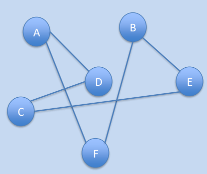
**Procedure**

1. Initialize S to an empty solution.
2. Pick randomly from available options.
3. Add chosen option to the partial solution S.
4. If S is not a complete solution, repeat at step 2.
5. If first iteration, store S as the best solution thus far, and repeat at Step 1.
6. Else if S is better than best solution found thus far, keep S, discard the old best , and then repeat at Step 1.

5

## Iterative Sampling Example

- Consider the traveling salesperson as an example.
- Begin with an empty solution, or in this case let's begin with city A in the solution.
  - Doesn't matter where we start since solution will be a cycle.
- Pick a random city from among the rest and add an edge to it from A.
- And repeat, adding a random city after that one, and so forth.




Then, repeat R times, returning the best of those R solutions.

6

Lesson 2

## SOLUTION CONSTRUCTION HEURISTICS

7

## An idea from Operations Research….

- Operations Research (OR) is a field devoted to problems related to optimizing industrial and business processes.
- Scheduling is a general class of problems of interest to both AI and OR.
- OR has produced the concept of a “dispatch scheduling policy.”
- Dispatch scheduling avoids the computationally hard problem of actually searching for an optimal schedule.
  - Essentially treats it as hopeless (e.g., scheduling problems often change dynamically, so an optimal solution now is obsolete a few minutes from now)
- Dispatch policies are heuristics that select from the available tasks or jobs, the one that should be scheduled next, and simply does so.
  - NO SEARCH

8

## Dispatch Scheduling Concept

- Given a set of tasks, $T = \{t_1, t_2, \ldots, t_n\}$, that must be scheduled over time.
- Find the schedule, S (assume for simplicity that we mean ordering over the tasks, but could be more complex), such that V(S) is minimized.
- V(S) is our optimization function, and can include a wide variety of functions from the field of scheduling.
  - E.g., makespan (total length of the schedule), tardiness, weighted tardiness, lateness, earliness—tardiness, number of late jobs, and many, many others.

9

## Dispatch Scheduling Concept

- Dispatch Scheduling:
  - H(t, s) is a dispatch policy that takes a task, t, and a partial schedule s, and returns an evaluation of how critical it is for task t to be the next task scheduled.
- Procedure:
  1. Initialize S to an empty schedule.
  2. Evaluate remaining tasks and pick the one with max value of H.
     - Assuming the higher values of H mean better choice (choose by min value otherwise).
  3. Add the chosen task to the partial schedule S.
  4. If there is at least 1 task not yet scheduled, repeat at step 2.

10

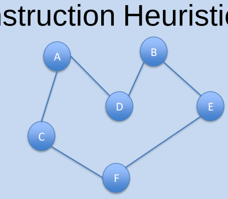
## Solution Construction Heuristics

- To use a more general term than dispatch policy....
- We’ll define a **Solution Construction Heuristic** as a heuristic or rule used to construct a solution to a problem.
  - Dispatch policies are examples for scheduling problems.
- E.g., For the Traveling Salesperson Problem (TSP) you might use distance to a city as the heuristic (where lower values are better):
  1. Initialize S to an empty tour of the cities.
  2. Evaluate remaining cities and pick the city, c, with min value of H, where H(c, S) is the distance to city c from the last city in the partial tour S.
  3. Add the chosen city to the partial tour S.
  4. If there is at least 1 city not yet visited, repeat at step 2.

11

## TSP Solution Construction Heuristic

- This example is with the TSP although not usually used for this problem.
- Let our heuristic be distance to nearest city.
- Start at city A.
- Add edge to closest city to A.
- And then closest city to that, etc.




Much better than the solution with the earlier example of Iterative Sampling.

12

Lesson 3

# HBSS, VBSS, AND HEURISTIC EQUIVALENCY

STOCKTON UNIVERSITY www.stockton.edu

13

## Randomizing a Solution Construction Heuristic

- A solution construction heuristic constructs 1 solution to the problem without any search.
- The quality of that solution might be good, or it might not be.
  - Heuristics are not perfect, and for any heuristic you can create an example where it will do poorly.
- Can we use it in some way to guide a search?
  - E.g., we saw heuristics for game playing are effective because we use them to evaluate thousands of terminal states.
  - E.g., we saw variable ordering and value ordering heuristics for constraint satisfaction don't necessarily lead directly to a solution but do tend to cut down search necessary.
- We'll look at multiple ways of using solution construction heuristics for search, and we'll do so by randomizing the heuristic.

STOCKTON UNIVERSITY www.stockton.edu

14

## Heuristic Biased Stochastic Sampling

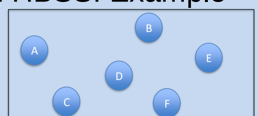
- Heuristic Biased Stochastic Sampling (HBSS) [Bresina]
  - Variation of Iterative Sampling
  - Uses a heuristic to bias the random decisions
- Each decision is made as follows:
  - Sort your choices by the solution construction heuristic.
  - Assign each a rank based on their position in the sorted list
    - First in list is rank 1, second in list is rank 2, etc.
  - Choose randomly from among the choices but bias by a function of the ranks.
    - The probability of choosing $c_i$ is defined as:
    - $P(c_i) = \frac{1/rank(c_i)^B}{\sum_{j=1}^{N} 1/j^B}$, where there are N choices, and B is a parameter

STOCKTON UNIVERSITY www.stockton.edu

15

## Random Decision in HBSS: Example

- Example assumes parameter B=1.
- Step 1: Sort the options by heuristic (in this case by distance to A).
  C, D, B, F, E
- Step 2: Assign ranks.
  - rank(C)=1, rank(D)=2, rank(B)=3, rank(F)=4, rank(E)=5
- Step 3: Compute $\sum_{j=1}^{N} 1/j^B$
  - N in this case is 5.
  - So we have $1 + \frac{1}{2} + \frac{1}{3} + \frac{1}{4} + \frac{1}{5} = \frac{137}{60}$.




We are at city A, where do we go next?

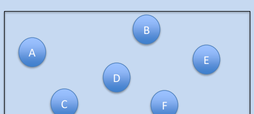
- Step 4: Determine the selection probabilities.

$\mathrm{P(C)} = \frac{1}{137/60} = \frac{60}{137}$, $\mathrm{P(D)} = \frac{1/2}{137/60} = \frac{30}{137}$,
$\mathrm{P(B)} = \frac{1/3}{137/60} = \frac{20}{137}$, $\mathrm{P(F)} = \frac{1/4}{137/60} = \frac{15}{137}$,
$\mathrm{P(E)} = \frac{1/5}{137/60} = \frac{12}{137}$.

STOCKTON UNIVERSITY www.stockton.edu

16

## Random Decision in HBSS: Example




We are at city A, where do we go next?

Step 4: Determine the selection probabilities.

$\mathrm{P(C)} = \frac{1}{137/60} = \frac{60}{137}$, $\mathrm{P(D)} = \frac{1/2}{137/60} = \frac{30}{137}$,
$\mathrm{P(B)} = \frac{1/3}{137/60} = \frac{20}{137}$, $\mathrm{P(F)} = \frac{1/4}{137/60} = \frac{15}{137}$,
$\mathrm{P(E)} = \frac{1/5}{137/60} = \frac{12}{137}$.

- So how do we implement the random decision using these biases?
- Generate a random value r uniformly from the interval [0.0,1.0).
- If $r < \frac{60}{137}$, choose city C.
- If $\frac{60}{137} \le r < \frac{90}{137}$, choose city D.
- If $\frac{90}{137} \le r < \frac{110}{137}$, choose city B.
- If $\frac{110}{137} \le r < \frac{125}{137}$, choose city F.
- If $\frac{125}{137} \le r$, choose city E.

STOCKTON UNIVERSITY www.stockton.edu

17

## Value Biased Stochastic Sampling (VBSS)

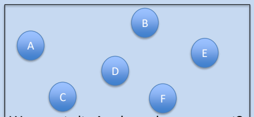
- Value Biased Stochastic Sampling (VBSS) [Cicirello & Smith]
  - Similar idea to HBSS.
  - But no ranking, no sorting…. Just use the heuristic values for the bias.
  - Has the effect of similar choices -> similar bias
  - With HBSS, even a slight discrepancy in heuristic causes one choice to have much more weight than another.
- Each decision is made as follows:
  - Let $H(c_i)$ be the heuristic value of choice $c_i$. [We'll assume lower H means better choice.]
  - Choose randomly from among the choices but bias by a function of the heuristic values.
    - The probability of choosing $c_i$ is defined as:
    - $P(c_i) = \frac{1/H(c_i)^B}{\sum_{j=1}^{N} 1/H(c_j)^B}$, where there are N choices, and B is a parameter

STOCKTON UNIVERSITY www.stockton.edu

18

## Random Decision in VBSS: Example

- Example assumes parameter B=1.
- Step 1: Compute heuristic value for all choices (in this case distance to city A)
  - H(B)=2, H(C)=1, H(D)=1.5, H(E)=3.1, H(F)=2.4.
- Step 2: Compute $\sum_{j=1}^{N} 1/H(c_j)^B$ .
  - We have $\frac{1}{2} + 1 + \frac{1}{1.5} + \frac{1}{3.1} + \frac{1}{2.4} = 2.906$
- Step 3: Determine the selection probabilities:
  - $P(B) = \frac{1/2}{2.906} = 0.17$, $P(C) = \frac{1}{2.906} = 0.34$, $P(D) = \frac{1/1.5}{2.906} = 0.23$, $P(E) = \frac{1/3.1}{2.906} = 0.11$, $P(F) = \frac{1/2.4}{2.906} = 0.14$
- You can then implement the actual random decision as before, with random r in [0.0, 1.0).



We are at city A, where do we go next?

STOCKTON
STOCKTON UNIVERSITY
www.stockton.edu

19

## Heuristic Equivalency

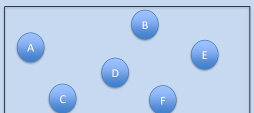
- Heuristic Equivalency [Gomes]
  - Computes heuristic value of all choices and finds the choice with best heuristic value.
  - Then finds all choices whose heuristic value is within X% of the best heuristic value.
  - Treats that set as equivalent and chooses uniformly at random from among them.

STOCKTON
STOCKTON UNIVERSITY
www.stockton.edu

20

## Heuristic Equivalency Example

- Step 1: Compute heuristic value for all choices (in this case distance to city A)
  - H(B)=2, H(C)=1, H(D)=1.5, H(E)=3.1, H(F)=2.4.
- Step 2: Find choice with best heuristic value.
  - In this case, city C, H(C)=1.
- Step 3: Find all of the cities whose heuristic value is within X% of the best heuristic value.
  - In this example, assume 50%.
  - Find all cities with H within 50% of H(C)=1.
  - That set is {C, D}.
- Step 4: Pick uniformly at random from that set.



We are at city A, where do we go next?

STOCKTON
STOCKTON UNIVERSITY
www.stockton.edu

21