6. Tabu Search
6.4 Tabu Search for TSP

Fall 2010

Instructor: Dr. Masoud Yaghini

Outline

- Basic TS Components
- Advanced Strategies
- References

Basic TS Components

Solution Representation

- A solution can be represented by a vector indicating the order the cities are visited
  - Example:

1	5	3	4	2	6	7	8

  - Numbers in the solution vector are interpreted as cities and not as positions in the solution vector

Initial Solutions

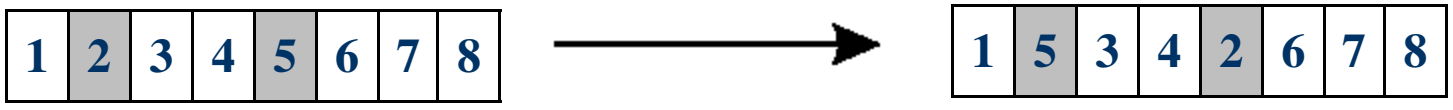
- A good feasible, yet not-optimal, solution to the TSP can be found quickly using a greedy approach (**the nearest-neighbor heuristic**).
- Starting with the first node in the tour, find the nearest node.
- Each time find the nearest unvisited node from the current node until all the nodes are visited.

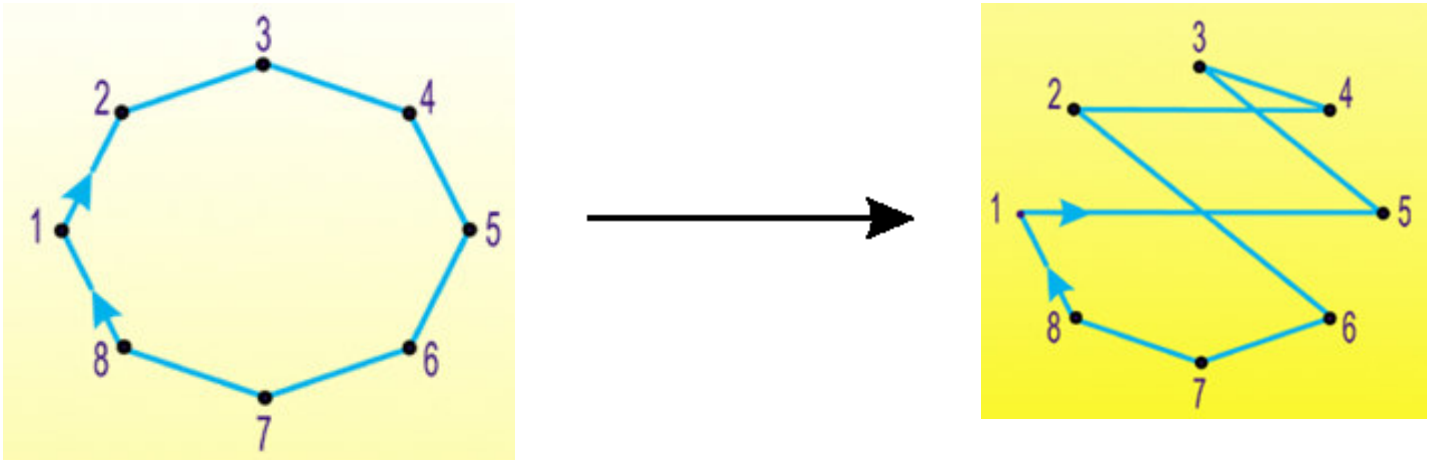
Neighborhood Structure

- A neighborhood to a given solution is defined as any other solution that is obtained by a pair wise exchange of any two nodes in the solution.
- This always guarantees that any neighborhood to a feasible solution is always a feasible solution (i.e, does not form any sub-tour).
- If we fix node 1 as the start and the end node, for a problem of N nodes, there are C^N_2 such neighborhoods to a given solution.
- At each iteration, the neighborhood with the best objective value (minimum distance) is selected.
- Neighborhood cardinality: $\binom{n}{2}$

Neighborhood Structure

- Neighborhood solution obtained by swapping the order of visit of cities 2 and 5



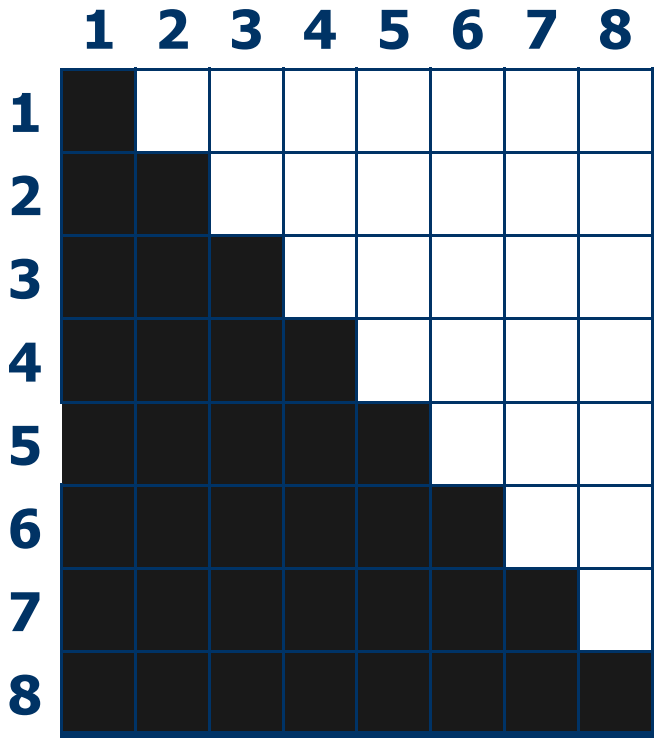


Tabu List

- To prevent the process from cycling in a small set of solutions, some attribute of recently visited solutions is stored in a Tabu List, which prevents their occurrence for a limited period.
- The attribute used is a pair of nodes that have been exchanged recently.
- A Tabu structure stores the number of iterations for which a given pair of nodes is prohibited from exchange.

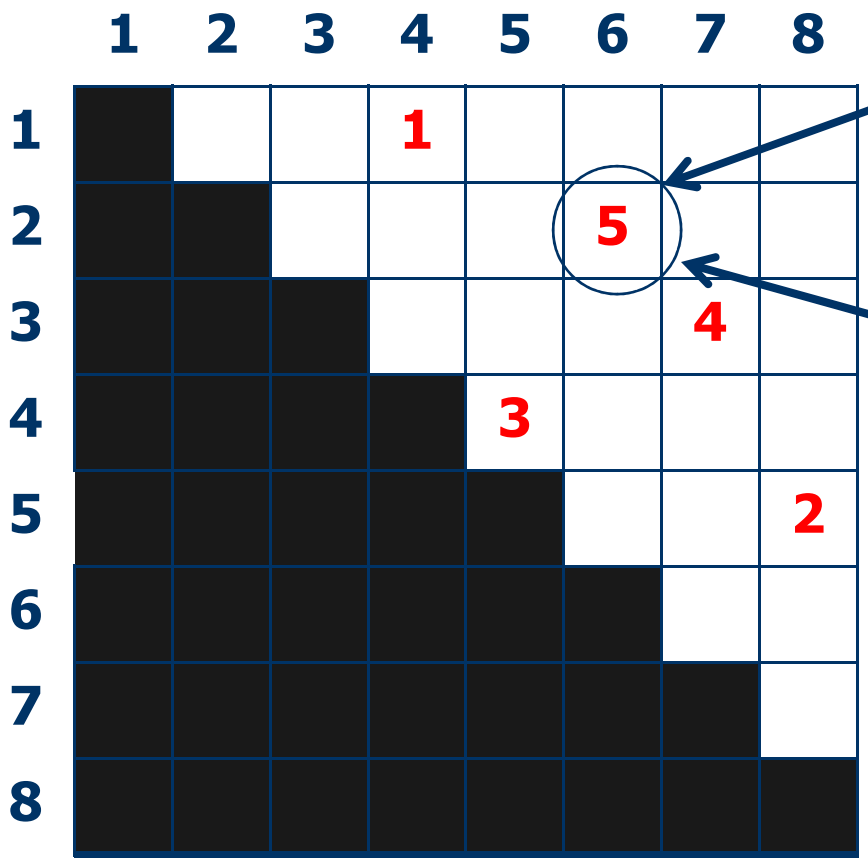
Tabu List

- The main memory component (short term memory) can be stored in a matrix where the swap of cities i and j is recorded in the i-th row and j-th column
- The memory can be organised as upper-triangular matrix



Short term memory

- An example of Short Term Memory (*M*)



Most Recent Swap

the number of remaining iterations that given swap stays on the Tabu list

Aspiration Criterion

- Tabus may sometimes prohibit attractive moves.
- It may, therefore, become necessary to revoke tabus at times.
- The criterion used is to allow a move, even if it is tabu, if it results in a solution with an objective value better than that of the current best-known solution.

Termination Criteria

- The algorithm terminates if there is no improvement in the solution for a pre-defined number of iterations

Advanced Strategies

A Speedup Technique

- For larger TSP problems if a small value ranging between 15 and 40 cities of the nearest-neighbor list is constructed, all local search can be done only on this nearest-neighbor list
- For example when we want to swap two cities,
  - First we choose a city and then
  - Check swapping only with the nearest-neighbor list

Diversification

- Quite often, the process may get trapped in a space of local optimum.
- To allow the process to search other parts of the solution space (to look for the global optimum), it is required to diversify the search process, driving it into new regions.
- This is implemented using **frequency based memory**.

Diversification

- The use of frequency information is used to penalize non-improving moves by assigning a larger penalty (frequency count adjusted by a suitable factor) to swaps with greater frequency counts.
- This diversifying influence is allowed to operate only on occasions when no improving moves exist.
- Additionally, if there is no improvement in the solution for a pre-defined number of iterations, frequency information can be used for a pair wise exchange of nodes that have been explored for the least number of times in the search space.
- Thus driving the search process to areas that are largely unexplored so far.

Diversification

- Long term memory:
  - Maintain a list of t towns which have been considered in the last k best (worst) solutions
  - encourage (or discourage) their selections in future solutions
  - using their frequency of appearance in the set of elite solutions and the quality of solutions which they have appeared in our selection function

Diversification

- An example of Long Term Memory (*H*) last 50 iterations

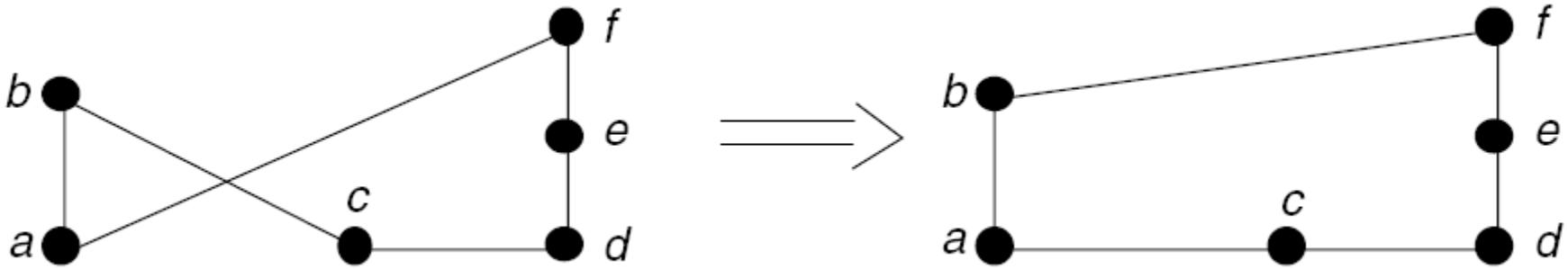
2	3	4	5	6	7	8	
0	2	3	3	0	1	1	1
	2	1	3	1	1	0	2
		2	3	3	4	0	3
			1	1	2	1	4
				4	2	1	5
					3	1	6
						6	7

Other Local Search Methods

- Swap operation of two cities on the tour for selecting the neighborhood is not the best choice for Tabu Search.
- Three other types of local search for the TSP:
  - **2-opt** neighborhood
  - **2.5-opt** neighborhood
  - **3-opt** neighborhood

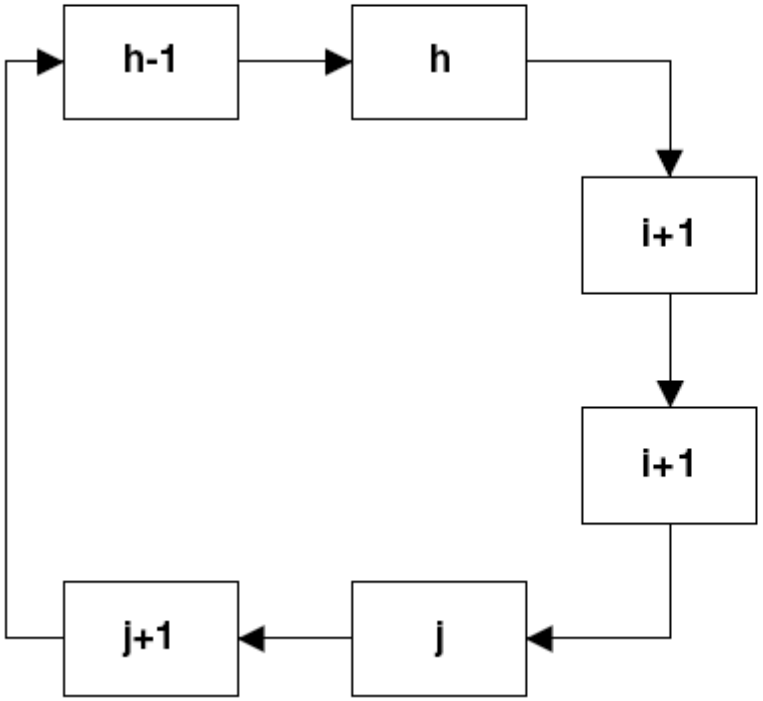
2-opt neighborhood

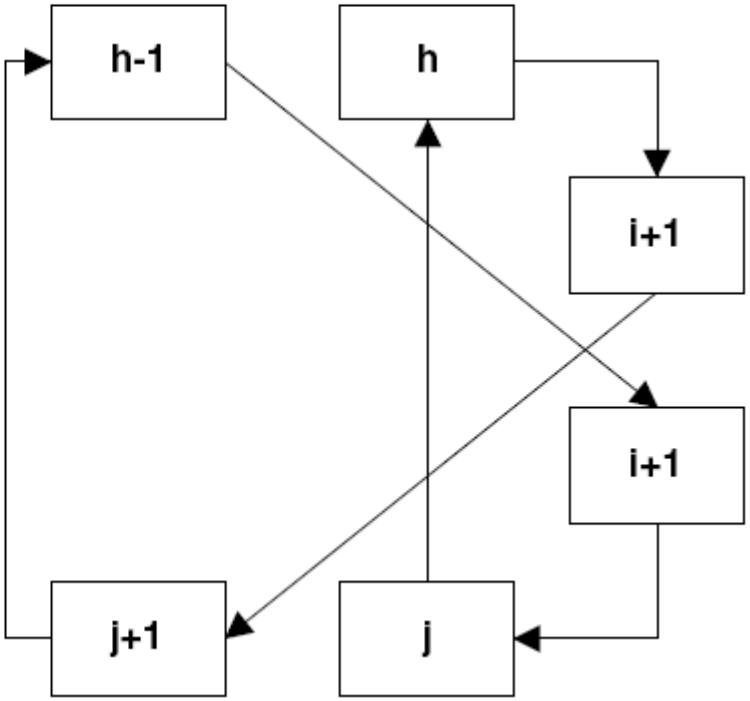
- The **2–opt neighborhoods** in the TSP
- Given a candidate solution s, the **TSP 2–opt neighborhood** of a candidate solution s consists of the set of all the candidate solutions s' that can be obtained from s by exchanging two pairs of arcs in all the possible ways.
- Example: the pair of arcs (b, c) and (a, f) is removed and replaced by the pair (a, c) and (b, f)



3-opt neighborhood

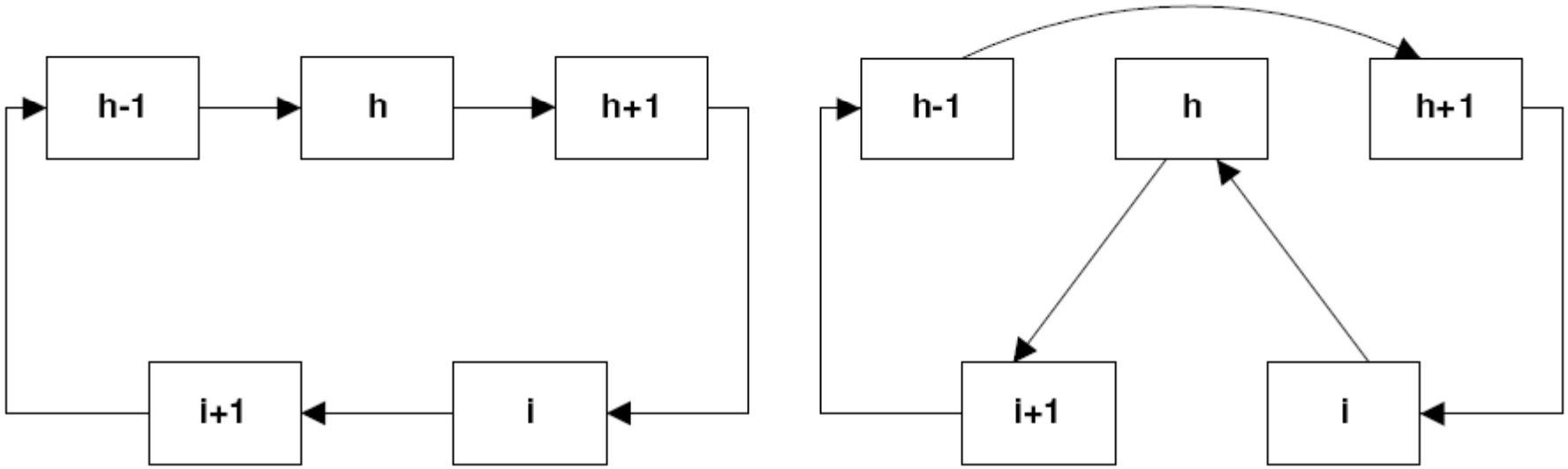
- The **3-opt neighborhood** consists of those tours that can be obtained from a tour s by replacing at most three of its arcs.
- In a 3-opt local search procedure 2-opt moves are also examined. Example:





2.5-opt neighborhood

- 2.5-opt checks whether inserting the city between a city i and its successor, as illustrated in the figure below, results in an improved tour.



The End