IE 607 Heuristic Optimization

Tabu Search

Origins

- Developed by Fred Glover in the 1970’s. Dr. Glover is a business professor at University of Colorado at Boulder.
- Developed specifically as a combinatorial optimization tool.

Inspiration

- Biological inspiration is ***memory*** – the ability to use past experiences to improve current decision making.
- Short term memory: recency based tabu list
- Long term memory: aspiration criteria, frequency based tabu list

Key Ideas

Is basically a ***single solution***, deterministic neighborhood search technique that uses memory (***tabu list***) to prohibit certain moves, even if they are improving. This makes tabu search a global optimizer rather than a local optimizer.

TS neighborhood: adjacent solutions that can be reached from any current solution by an operation called a move.

Components

- Encoding
- Objective function
- Move operator
- Neighborhood of candidate moves
- Termination criteria
- Tabu list(s)
- Aspiration criteria (optional)

Memory Aspects

- Recency (short term)

  →how recently was I here?
- Frequency (long term)

  →how often have I been here?
- Quality (aspiration)

  →how good is being here?
- Influence (aspiration)

  →how far away am I from where I have just been?

Canonical TS

```
Set tabu list length (t)
Randomly generate a single initial solution
Until stopping criteria is met {
        for i=1 to N {
                make a move to a neighboring solution
                evaluate
                record in ranked order, j=1 to N
                i++}
        for j=1 to N {
                if solution j is not tabu, go to 100
                else if solution j is better than best_so_far, go to 100
                j++}
100 add solution j to tabu list
    if tabu list length > t, remove oldest solution from the list
    if solution j is better than best_so_far, best_so_far = solution j
}
Return best_so_far
```

TS Example 1 – Single Machine Scheduling

Tabu List Specifics

- The main objective of the tabu list is to avoid ***cycles***, thus making a global optimizer rather than a local optimizer
- **Length**

  static→t=7 or t=$\sqrt{n}$, n: problem dimension

  dynamic→vary t (randomly or systematically) between t_{min} and t_{max} every x iterations; or choose t as previous rule, but determine t_{min} and t_{max} to be larger for attributes that are more attractive

> *Attribute*: certain move or solution components
> Have participated in generating past solutions.

Tabu List Specifics (cont.)

- **Content**

  from attributes, *to* attributes, *move* attributes, and *solution* attributes

  →each move attribute may be expressed as an ordered pair (from attribute + to attribute)

  →the more specific, the less restrictive
- **Frequency**

  tallying similar solutions through tabu content. Usually used in penalty form rather than strict tabu.

Neighborhood Specifics

- Complete (deterministic)
- Partial (probabilistic)
- First improvement
- Only improving
- Compound neighborhood

Aspiration Criteria Specifics

- Best so far
- Best in neighborhood
- Dissimilar to existing solution (diversification)
- Similar to existing solution (intensification)
- High influence – degree of change in structure or feasibility

Influence: measures the degree of change induced in solution structure or feasibility especially important during intervals of breaking away from local optimality

TS Example 2 - TSP

- A B C D E   encoding of 5 cities
- t = 3, move = swap position of 2 cities
- Tabu list possibilities:

  any move of that city

  any pair of cities swapped

  any pair of cities and their sequences

TS Example 2 – TSP (cont.)

B A C D E	t = {A, B}
	t = {A+B}
	t = {B(1), A(2)}
B A C E D	t = {D, E}
	t = {D+E}
	t = {E(4), D(5)}

TS Example 2 – TSP (cont.)

C A B E D	t = {**B**, C} **TABU**
	t = {B+C}
	t = {C(1), B(3)}
C B A E D	t = {**A, B**} **TABU**
	t = {**A+B**} **TABU**
	t = {A(3), B(2)}

TS Example 2 – TSP (cont.)

C A D E B t = {**B**, **D**} **TABU**

t = {B+D}

t = {B(5), D(3)}

For frequency based tabu list, might use a city in a given position or the swap of two cities, or any movement of a give city, e.g., if city A is put too often in position 2, or if cities C and D are swapped too often, or if city E is moved too often.

More Complex Strategies

- **Strategic Oscillation**

  →process of repeatedly approaching and crossing the boundary from different directions

  →generally moving between feasible and infeasible border

- **Backtracking**

  →keep track of elite solutions and restart at them but move in a different direction

- **Restart**

  →like backtracking, but start from purely random new solution

More Complex Strategies (cont.)

- **Path Relinking and Tunneling**

  →move step by step (by the shortest route) from one good solution to another solution (tunneling allows stepping through infeasible solutions). Then restart at one of these interim solutions.

  →if the good solutions are similar, this is intensification.

  →if the good solution are dissimilar, this is diversification.

Downsides of TS

- No mathematical theory of TS
- Single solution search method
- Tends to prematurely converge to local optimum – must counteract this by diversification, restart
- Can be sensitive to starting solution
- Can be sensitive to tabu list length and content
- Is most effective where there is a good neighborhood structure then can be exploited through proper moves
- Not very appropriate for continuous optimization

Websites

- http://spot.colorado.edu/~glover/
  Fred Glover's webpage at Univ. of Colorado
- http://www.cs.sandia.gov/opt/survey/ts.html
  short page with some C code for a version of TS
- http://www.sce.carleton.ca/netmanage/tony/ts.html
  page with links