An attribute based Similarity Function for VRP Decision Support

Arne Løkketangen
Molde University College, Norway

Kjetil Fagerholt
Jarl Korsvik
Johan Oppen
David L. Woodruff

Outline

- We look at distance/similarity measures for solutions to combinatorial optimization problems
- More particularly, we consider the family of vehicle routing problems (VRP).
- Our goal is the specification of similarity measures between solutions to a VRP instance.
- We will illustrate the use of this with 2 examples
  - Find *dissimilar* solutions for presentation to a DM after the search
    - Rich VRP
  - Produce *similar* solutions *during* the search
    - Ship scheduling – *Rolling Horizon*

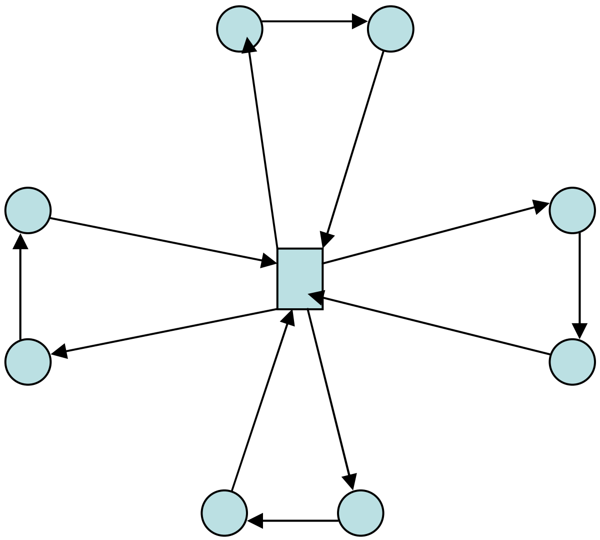
Decision Support Systems

- When we solve an optimization problem, we usually only have an approximate model of the real problem.
- There are often aspects of the problem that is not present in the model, for practical, political or other reasons
- The optimal solution might therefore only be of marginally more interest to a decision maker (DM) than other, good, solutions

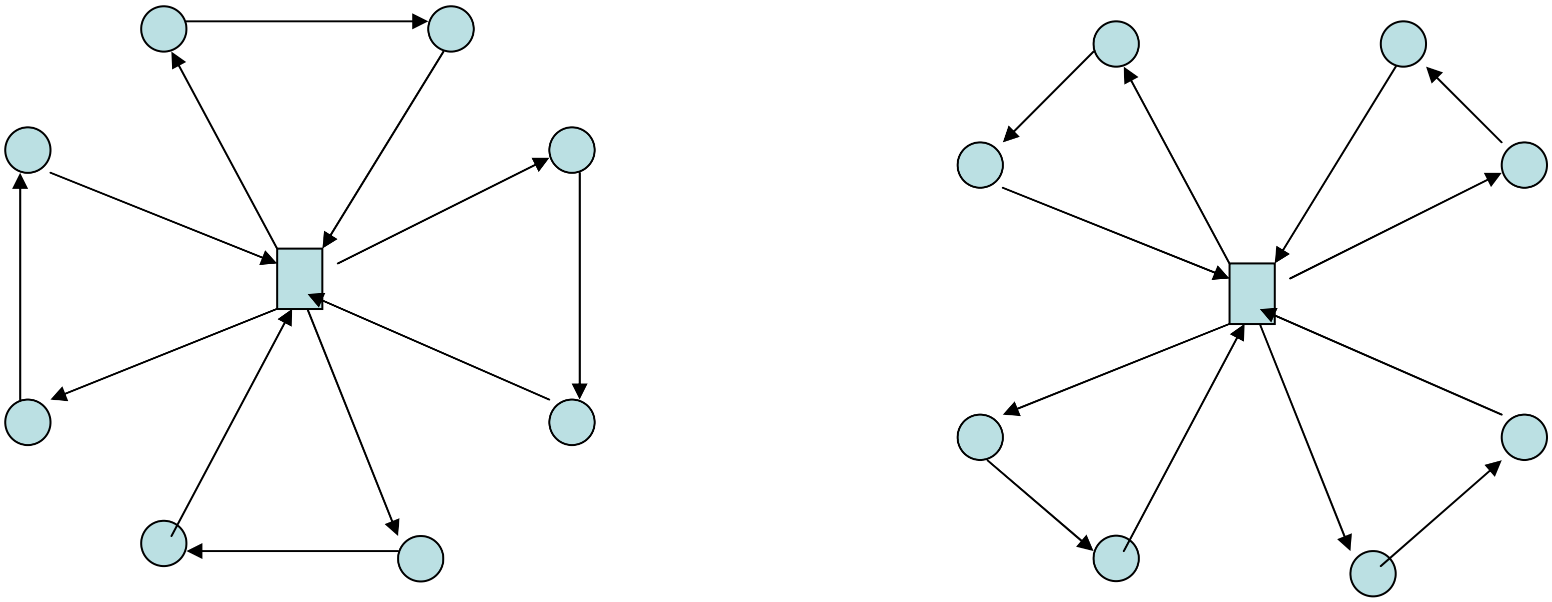
DSS - Distance measures

- Search processes often generate a plethoria of solutions
- When should a new solution be presented to the DM?
- Some distance measure should be used
- Can use Hamming Distance on the 0/1 solution vector - weak

Similarity is NOT Visual Pattern Matching



Similarity is NOT Visual Pattern Matching



People will say the routes are very similar,
Only rotated 45 degrees
⇒ Can not use isotropic measures

Our Distance/Similarity Measures

- The words *similarity* and *distanc*e are complimentary and the literature for computing values for them is intertwined.
- We will use the terms *similarity* and *distance* function in their broad, intuitive sense
- For making measurements, and comparing them, we need a *metric*

Distance metrics

- To have a metric, we need
  1) $d(x,x) = 0$
  2) $d(x,y) > 0$ if $x \neq y$
  3) $d(x,y) = d(y,x)$
  4) $d(x,y) + d(y,z) \geq d(x,z)$
- We are not too concerned with 4) – the triangle inequality
  – Our metric is a semi-metric

Tversky's similarity measure

- Given 2 sets A and B, then

$$J(A,B) \equiv \frac{|A \cap B|}{|A \cap B| + |A - B| + |B - A|}$$

- |A| denotes the cardinality of A
- $1 - J(\bullet)$ is a semi-metric
- We will base our measure on generalizations of this ratio

Other Similarity/Distance Measures

- Number of common edges
  - Similar to Hamming distance
  - This is presumably too weak, ignores structure
  - E.g. TSP, VRP – n edges in the tour, n^2 in total

Richer Models

- Our measures automatically applies to solutions of *richer* VRP models
  - Time Windows
  - Pickup-and-Delivery
  - …
- Our measures are concerned with *solutions*, and not the *constraints*

Computations

- We use a CVRP solver based on Laporte's solver (move a customer to a neighboring tour).
- Testcases are taken from standard benchmarks on the web, supplemented by real-world cases
- The solver collects the best solutions
-

Adding Attributes

- A solution often has attributes associated with its components
- For a VRP, attributes can be associated with
  - Stops
  - Arcs
  - Tours

Attributes for Stops

- Accessability
  - parking
  - manouvring
  - loading/unloading facilities
- Time windows
- Load
  - type of load
  - pickup or delivery
  - amount

Attributes for Arcs

- Length
- Road quality
  - Number of lanes
  - axle pressure
  - slope
  - curves
- Travel time
  - Average
  - variability
- Other travel time variations
  - rush hour
  - ferry routes

Attributes for Tours

- Day/time of tour
- Driver
- Vehicle
- Importance
  - criticality of load

Extended similarity measure

- With attributes we must extend the similarity measure

- Ex: Difference between vector elements x_j and y_j

$$\eta_j\left(x_j, y_j\right) \equiv \min\left(1, \frac{\left|x_j - y_j\right|}{s_j(\bullet)}\right)$$

- $s_j(\bullet)$ is some dispersion measure

Vector distance

- The distance between two vectors (of attributes) *x* and *y* is

$$\delta(w; x, y) \equiv \left(\sum_{j=1}^{p} \eta_j (x_j, y_j) w_j \right) \Bigg/ \sum_{j=1}^{p} w_j$$

Sets of vectors

- Given two sets A and B, the following is a generalization of | A – B |

$$g(w; A, B) \equiv \sum_{k \in (A-B)} \sum_{k' \in B} \delta(w; A_k, B_{k'}) \Big/ |B|$$

Sets of vectors

- Given two sets A and B, the following is a generalization of $|A \cap B|$

$$h(w; A, B) \equiv \left(|A| - g(w; A, B) + |B| - g(w; B, A) \right) / 2$$

Tversky's dissimilarity measure between two sets of vectors

$$d(w;A,B) \equiv 1 - \frac{h(w;A,B)}{h(w;A,B) + g(w;A,B) + g(w;B,A)}$$

Note

- Our distance measure is valid only for feasible solutions
- Can use it for infeasible solutions anyway – the user (implementor) knows the nature of the infeasibility

Computational tests

- Real world data – transportation of livestock to a slaughterhouse in Norway. One week horizon
- Attributes:
  - Stops
    - Order number
    - Animal type
    - Size of order
  - Arcs
    - Identity
    - Length
  - Tours
    - Vehicle
    - Weekday
    - *Criticality* – Is it necesary for immediate production ?

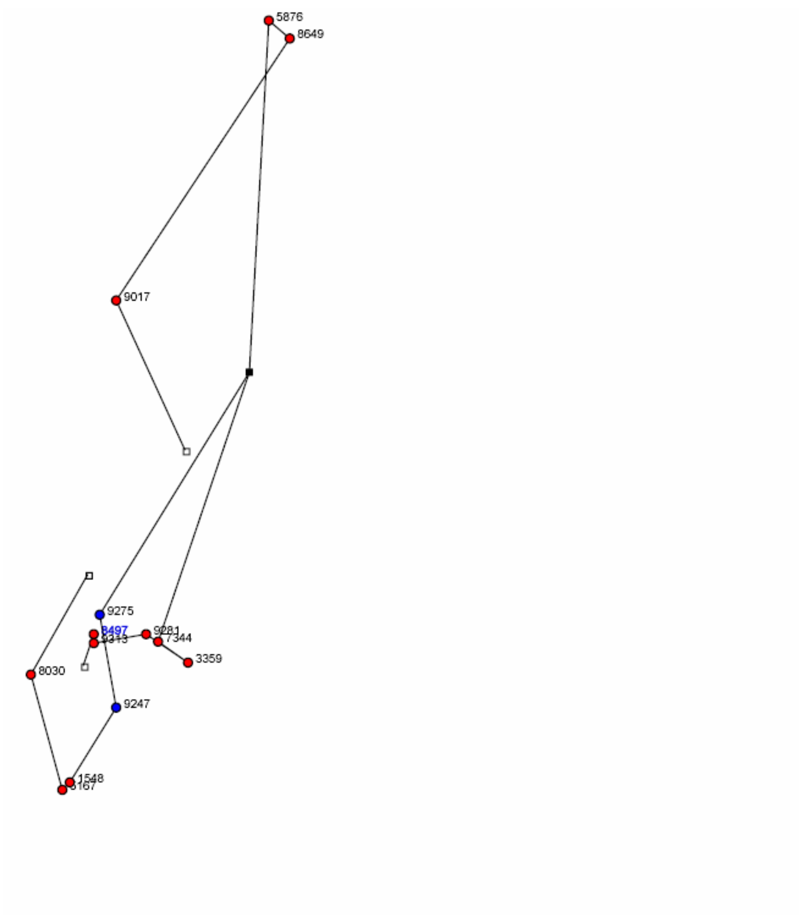
Abandon usual practice

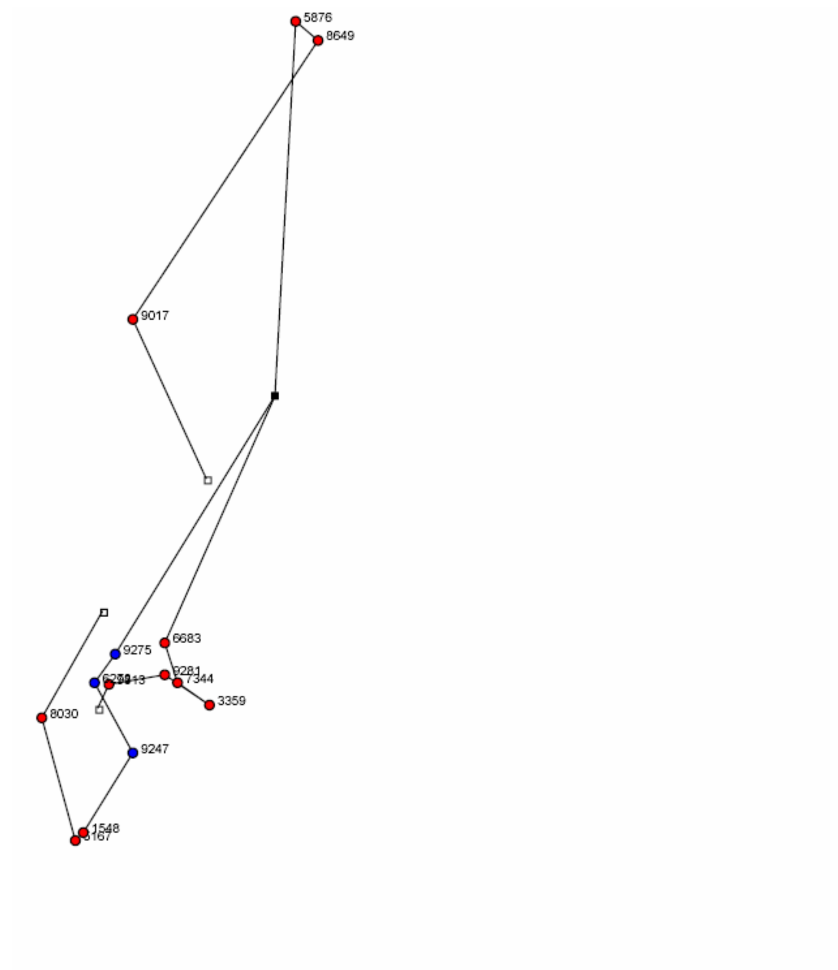
- Usually collect the K best (K =5) solutions
- With a local search basis, many of these will be very similar (or marginally different), being collected on a descent phase.

Collection Strategy

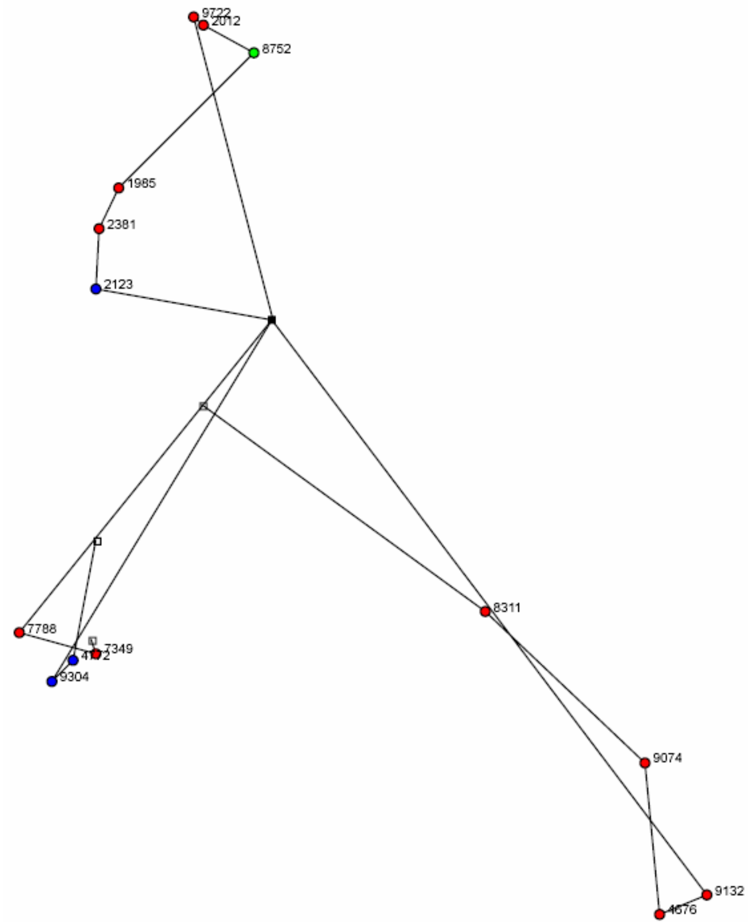
- Only collect solutions that are sufficiently different
- If a new better, close, solution is found, throw out the previous one.
- Need a threshold for goodness, and for distance
- Required goodness (or quality) should diminish with distance
- The user (DM) should decide on K, the number of solutions to keep.
- These diverse solutions could be used for later intensification
  - Possibly user initiated

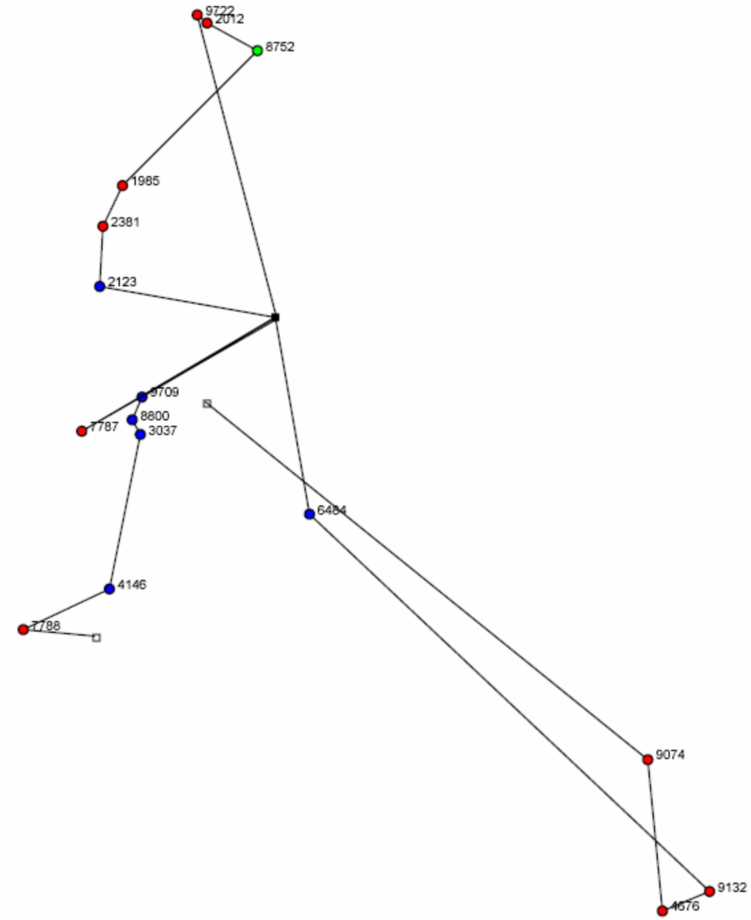
Using the difference measure – sunday



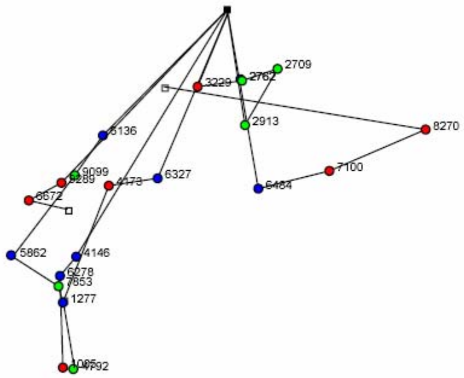


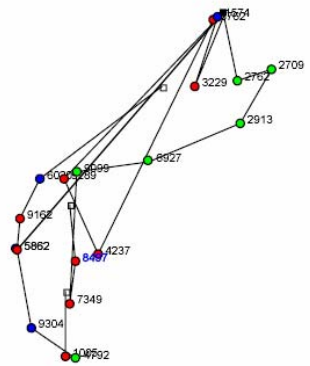
Using the difference measure – monday



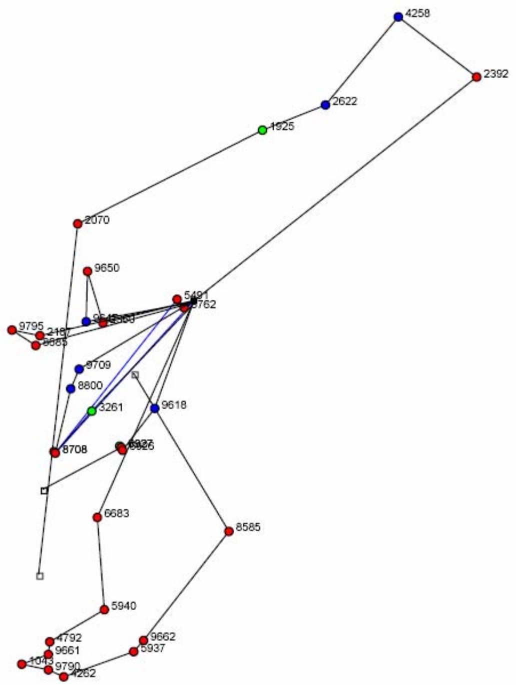


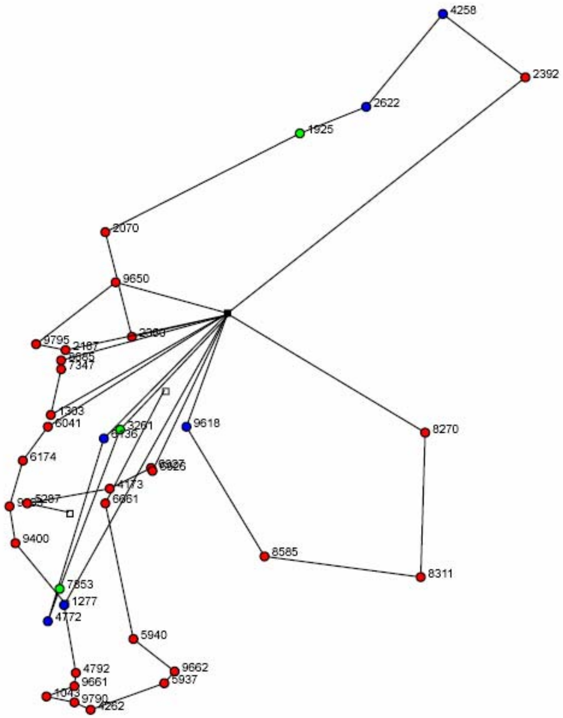
Using the difference measure – tuesday



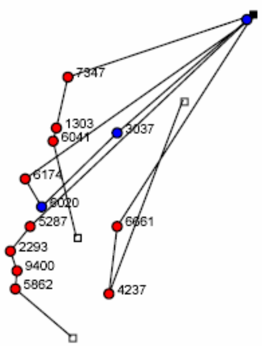


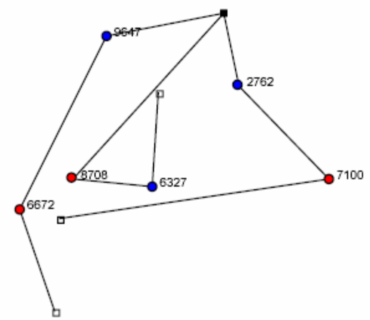
Using the difference measure – wednesday



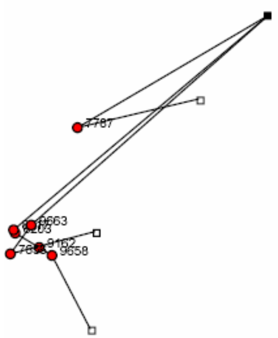


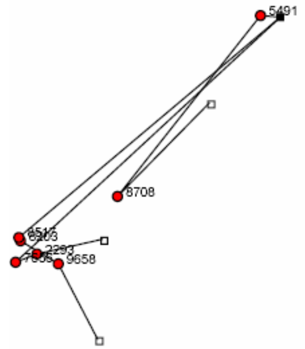
Using the difference measure – Thursday



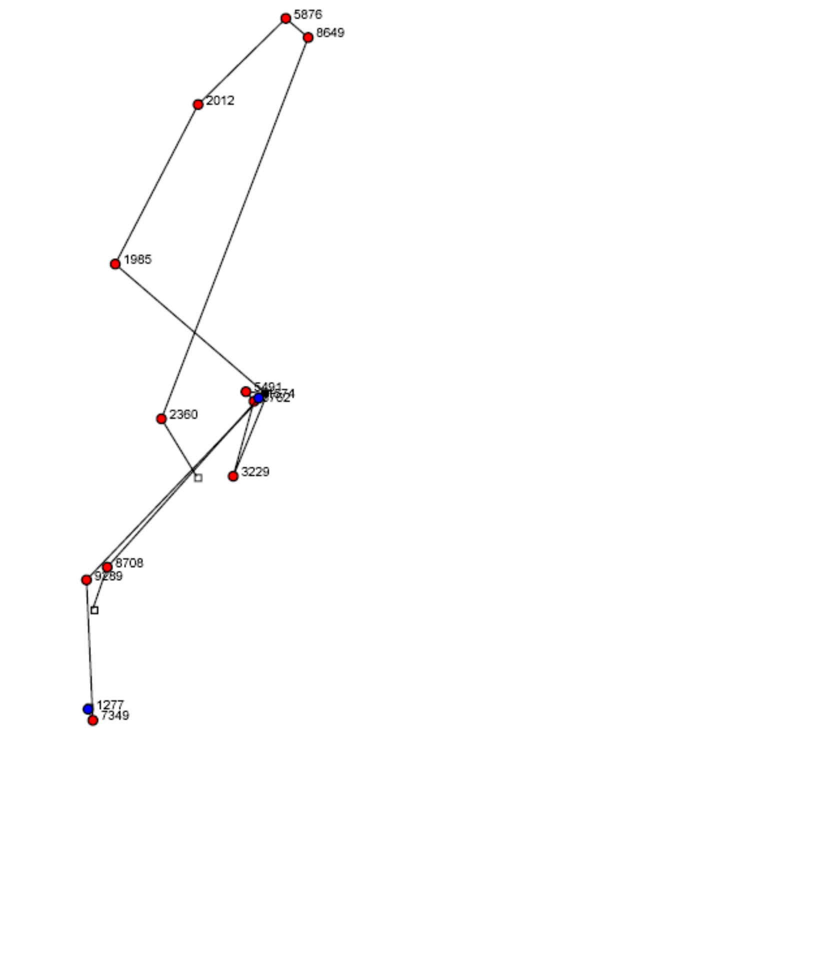


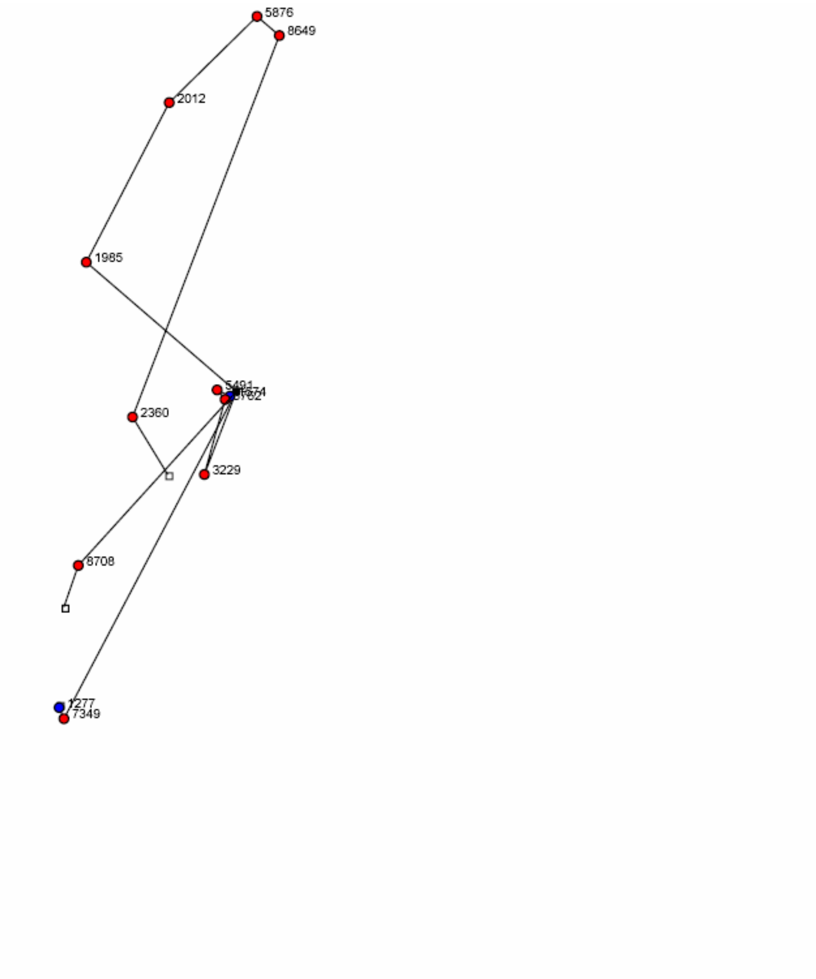
Using the difference measure – Friday



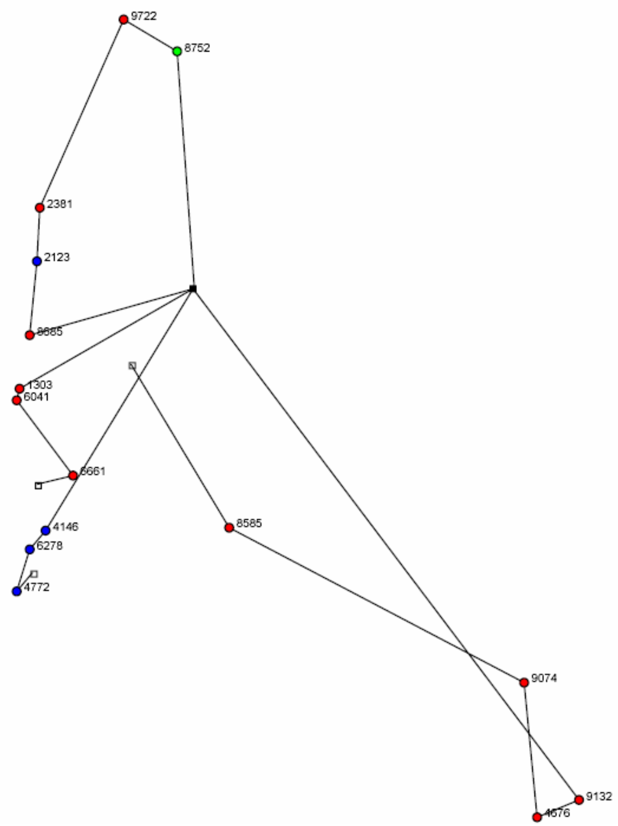


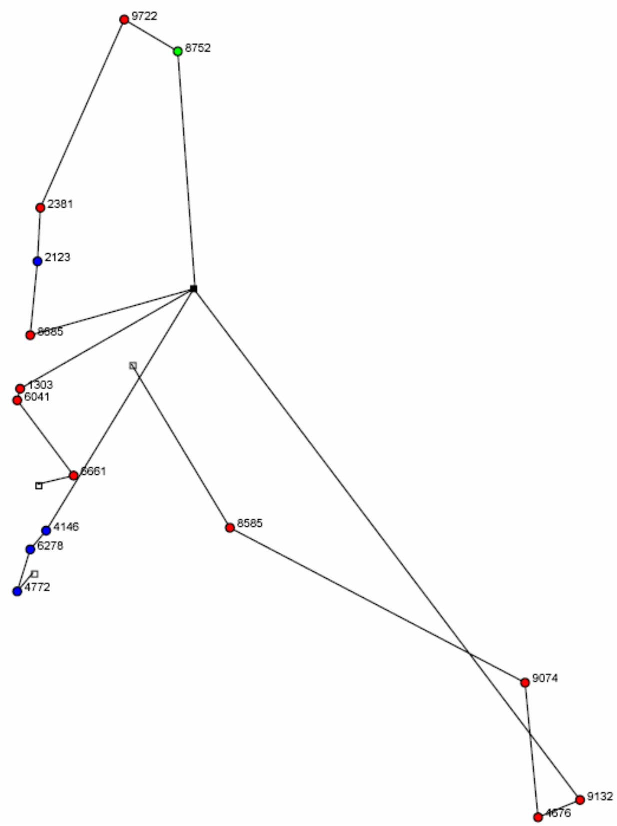
Not using the difference measure – Sunday





Not using the difference measure – Monday





Ship routing example

- Example from tramp shipping
- No depot
- Mandatory and optional cargoes
- Rolling horizon planning
  - Add one, or a few, new orders to the current plan
  - The new plan should be similar to the current plan
    - at least in the near future
    - larger differences far into the future is less important

Similarity Measure Needed !

- Need some way to both
  - generate new plans similar to the current plan
  - discriminate between such plans
- This clearly implies a tradeoff between reluctance to change and possible profit loss
- Important to include *nearness in time* in the measure

Persistance Penalty Function

- Difference penalty between plans A (new) and B (current)
- U_i^{AB} is 1 of cargo i is transported on different ships
- P_1 is the *cargo-ship* penalty
- P_2 is the *cargo-time* penalty

$$P(A) = \sum_{i \in \mathrm{N}_P} P_{1i} \cdot U_i^{AB} + \sum_{i \in \mathrm{N}} P_{2i} \left| T_i^A - T_i^B \right|$$

Distance Measure

- Proportion of cargoes *i* that have changed ship between plans A and B

$$D_{AB} = \frac{1}{N} \cdot \sum_{i \in \mathrm{N}_P} U_i^{AB}$$

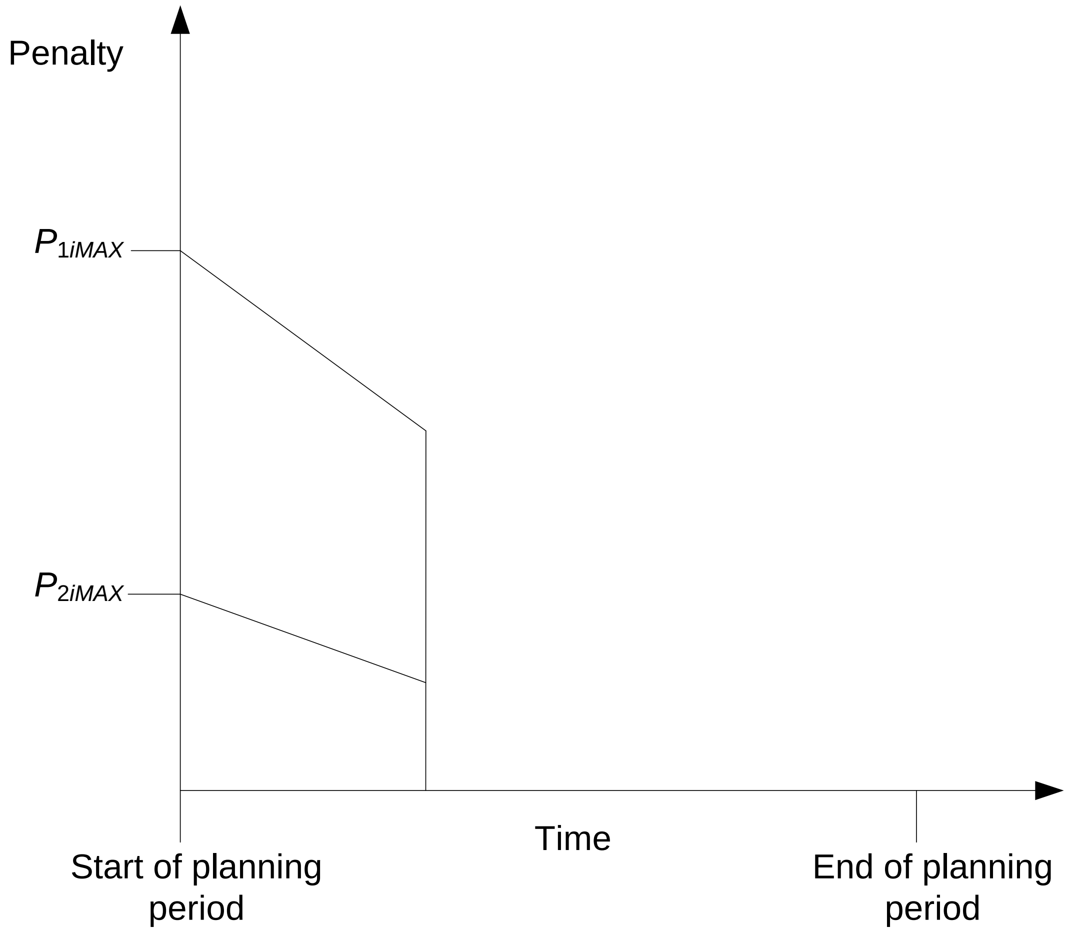
Solution Process

- The solver has a constructive and an iterative phase
- The *Persistence Penalty Function* is included in the Move Evaluation Function when *solving*
- The *Distance* measure is used on two occasions:
  - First it is used to identify a diverse set of start solutions
  - Secondly it is used to identify the set of solutions to present to the user

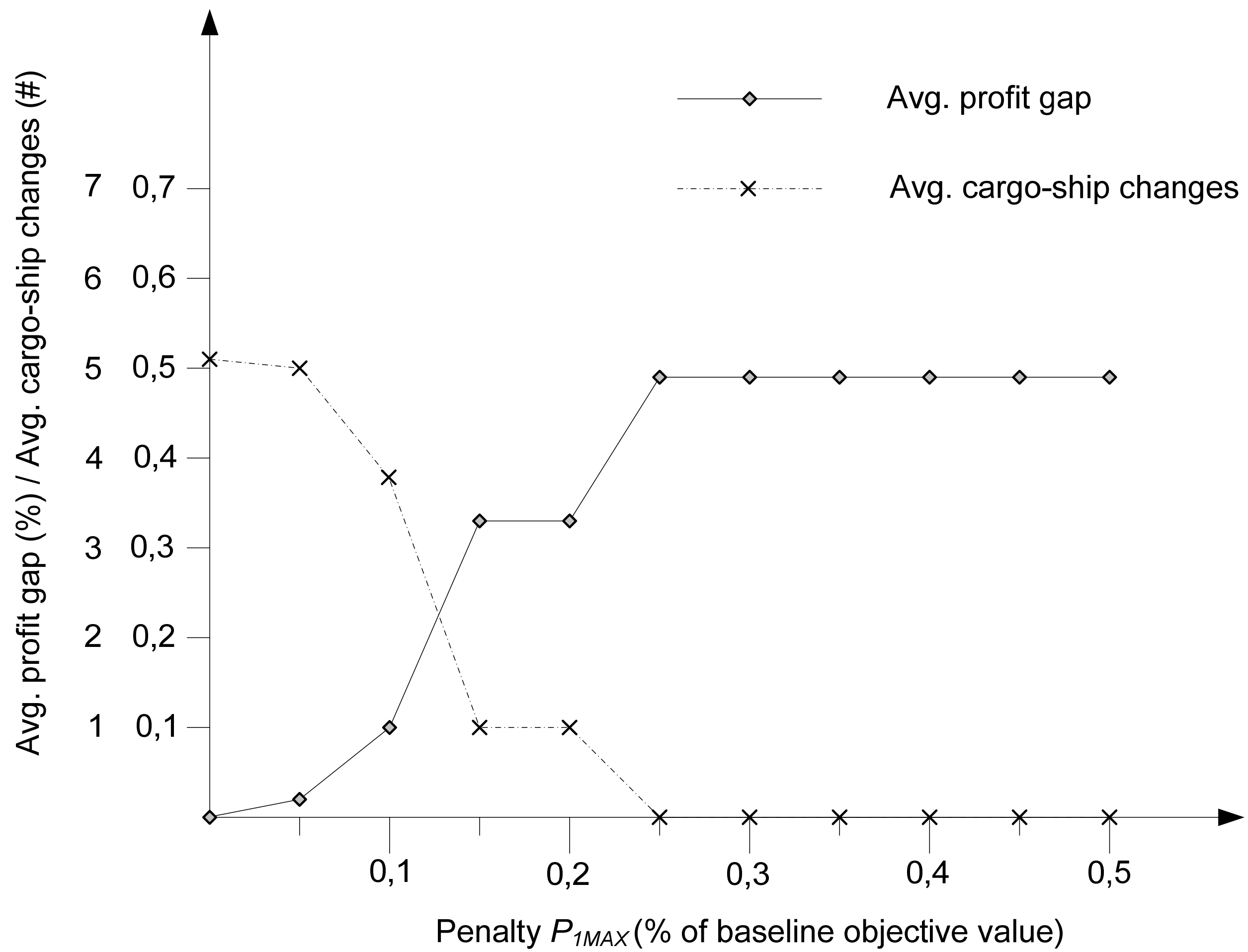
Test Cases

Case	1	2	3	4
Planning horizon [days]	30	30	90	150
# cargoes	24	31	17	40
# optional spot cargoes	4	4	4	6
# ships	7	13	5	6

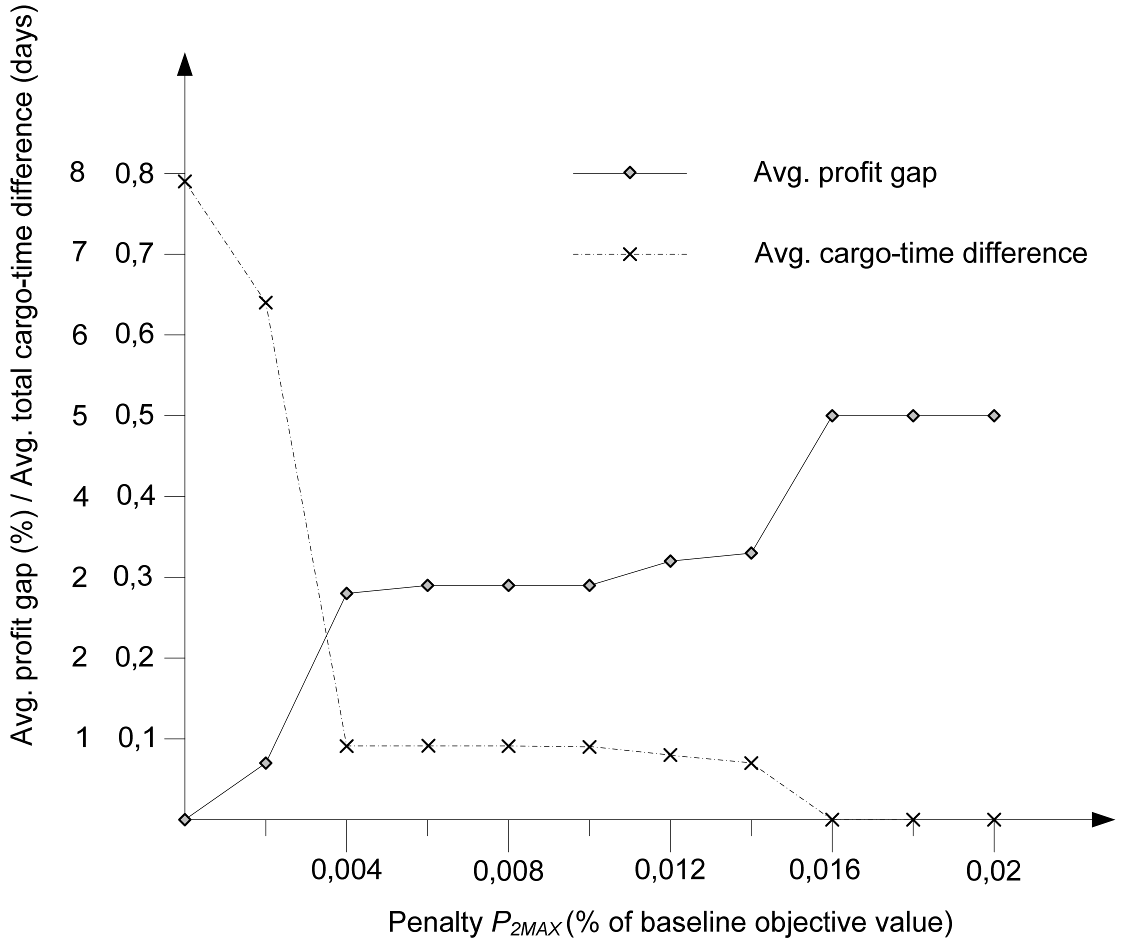
Persistence Penalty as a function of time



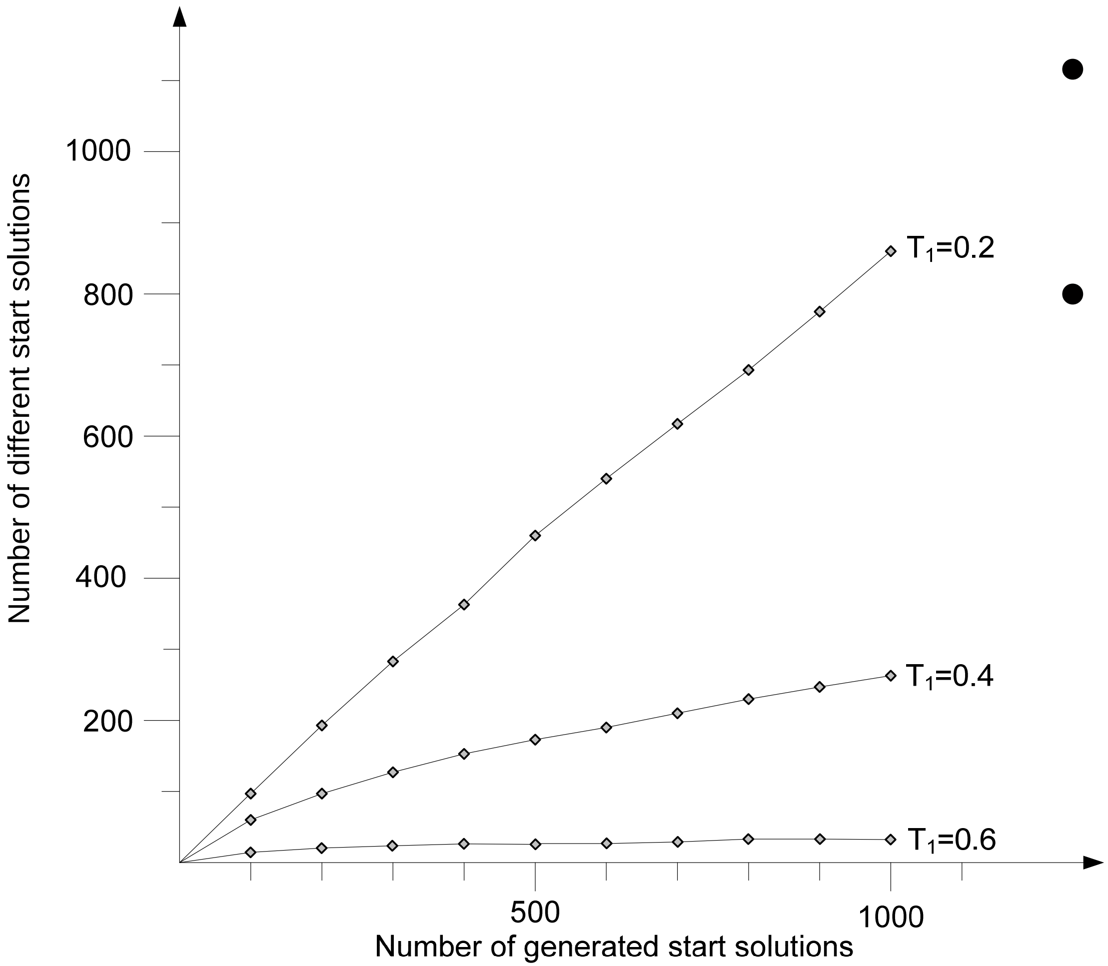
Profit/Change Tradeoff – P_1 - case 2



Profit/Change Tradeoff – P_2 - case 2

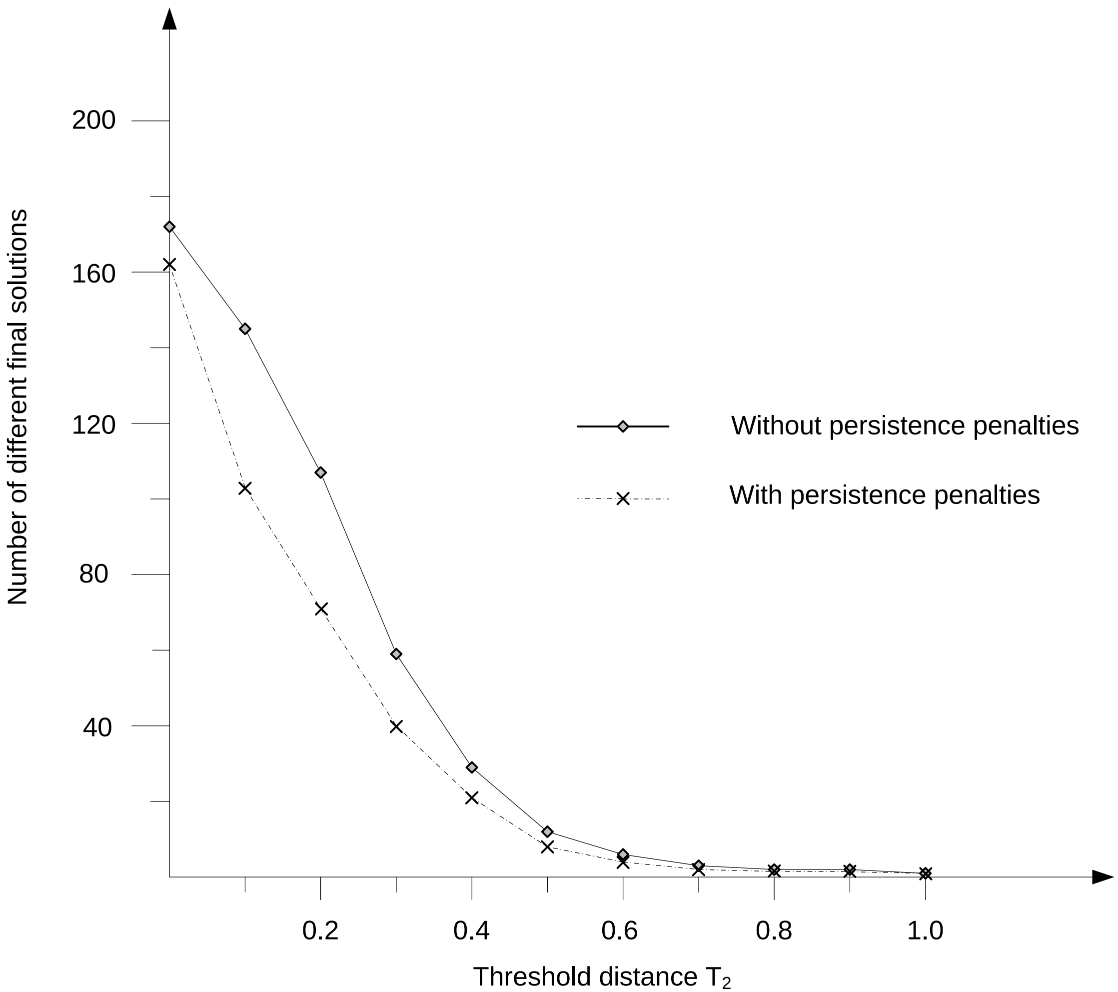


Different Start Solutions – case 2



- Only distance measure used
- T_1 is distance threshold

Different Final Solutions – case 2



Conclusions

- Using solution attributes to distinguish between solutions seems to be good.
- Measures based on these values can be used in different settings
- This is much closer to what a DM wants (or how she works) than just looking at the objective function value.