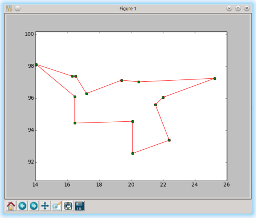
B4M36UIR & BE4M36UIR

Task08 - Multi-goal path planning and data collection path planning - TSP-like formulations

Deadline	1. December 2018, 23:59 PST
Points	4
Label in BRUTE	Task08
Files to submit	archive with `som_solver.py`
Resources	Task08 resource package

Assignment

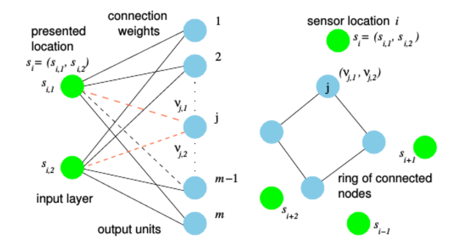
Download provided codes and run the main script `eval_som.py`



Slide 15 from the Lecture08:

- Sensor locations $S = \{s_1, \ldots, s_n\}$, $s_1 \in \mathbb{R}^2$; Neurons $\mathcal{N} = (\nu_1, \ldots, \nu_m)$, $\nu_i \in \mathbb{R}^2$, $m = 2.5n$
- Learning gain σ; epoch counter i; gain decreasing rate $\alpha = 0.1$; learning rate $\mu = 0.6$

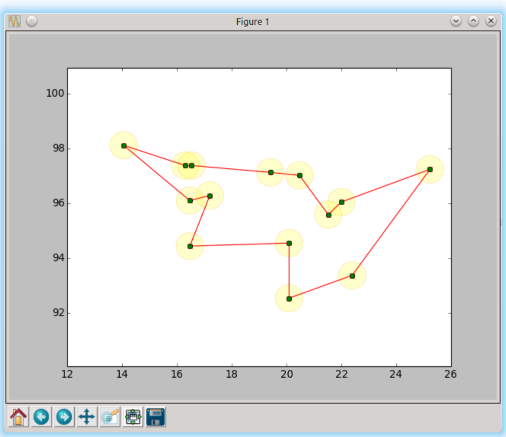
1. $\mathcal{N} \leftarrow$ init ring of neurons as a small ring around some $s_i \in S$, e.g., a circle with radius 0.5
2. $i \leftarrow 0$; $\sigma \leftarrow 12.41n + 0.06$;
3. $I \leftarrow \emptyset$ *//clear inhibited neurons*
4. **foreach** $s \in \Pi(S)$ (a permutation of S)
   4.1 $\nu^* \leftarrow \text{argmin}_{\nu \in \mathcal{N} \setminus I} ||(\nu, s)||$
   4.2 **foreach** ν in d neighborhood of ν^*
   $$\nu \leftarrow \nu + \mu f(\sigma, d)(s - \nu)$$
   $$f(\sigma, d) = \begin{cases} e^{-\frac{d^2}{\sigma^2}} & \text{for } d < 0.2m, \\ 0 & \text{otherwise,} \end{cases}$$
   4.3 $I \leftarrow I \bigcup \{\nu^*\}$ // inhibit the winner
5. $\sigma \leftarrow (1 - \alpha)\sigma$; $i \leftarrow i + 1$;
6. `If` (**termination condition** is not satisfied) `Goto` Step 3; `Otherwise` retrieve solution

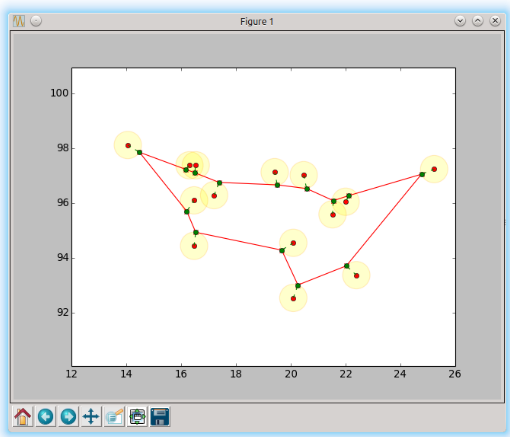


Termination condition can be

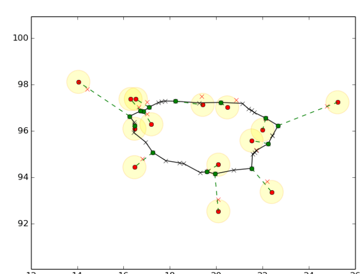
- Maximal number of learning epochs $i \leq i_{max}$, e.g., $i_{max} = 120$
- Winner neurons are negligibly close to sensor locations, e.g., less than 0.001

Utilize 'alternate goal' concept for solving TSP with neighborhoods (TSPN). In each epoch, the neurons are adapted towards the goals which inhibits them. But, in the TSPN, the neurons are adapted to the closes point in the specific goal neighborhood. Therefore, this concept enables to find shorter solutions, see the right image and the following GIF with SOM evolution.





Click on the following image to see the SOM evolution in GIF.



Tasks (4b)

1. Familiarize yourself with a concept of self-organizing maps for TSP-like problems.
2. Implement method `select_winner(…)` which select the closes not inhibited neuron to the target. (1b)
3. Adapt neighborhood of the winner neuron towards the target in the method `learn_epoch(…)` . (1b)
4. Extend the provided solver for the TSP with neighborhoods and modify function `update_goal_potition(…)` to utilize the concept of `alternate goal` . (2b)

courses/b4m36uir/hw/task08.txt · Last modified: 2018/11/19 13:01 by vanapet1