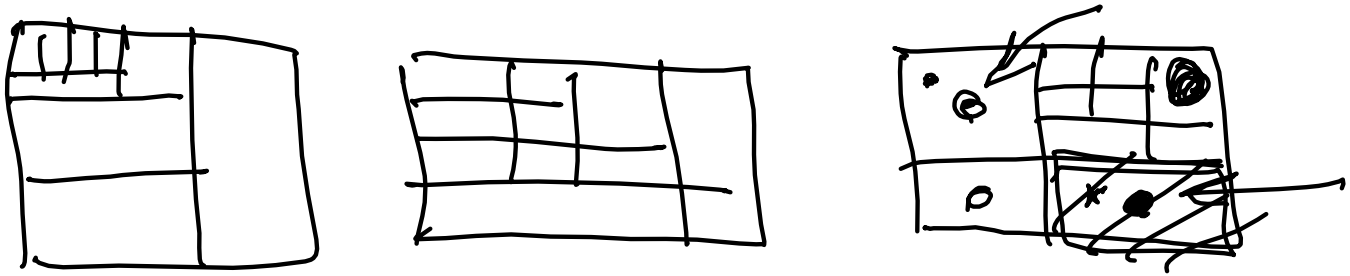
Metaheuristic Optimization: Tabu Search

Adaptive and Cooperative Algorithms (ECE 457A)

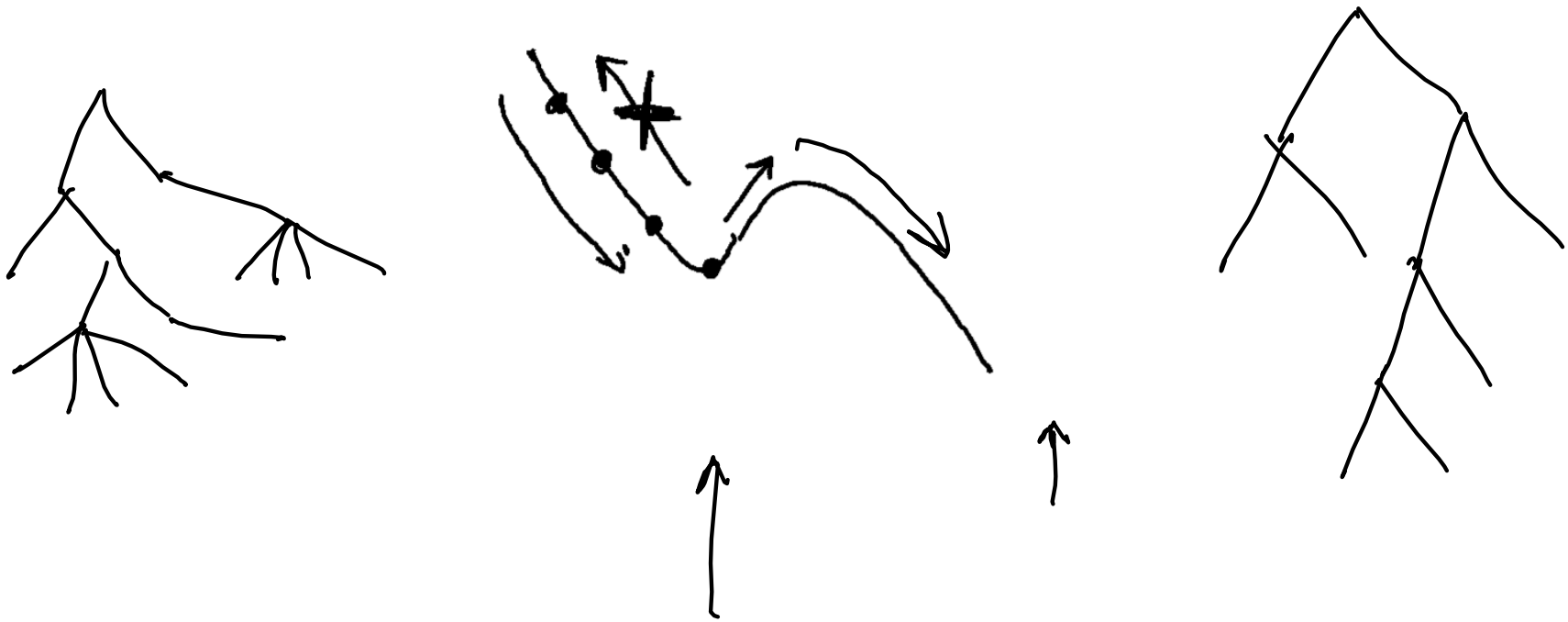
ECE, MME, and MSCI Departments,
University of Waterloo, ON, Canada

Course Instructor: Benyamin Ghojogh
Fall 2023

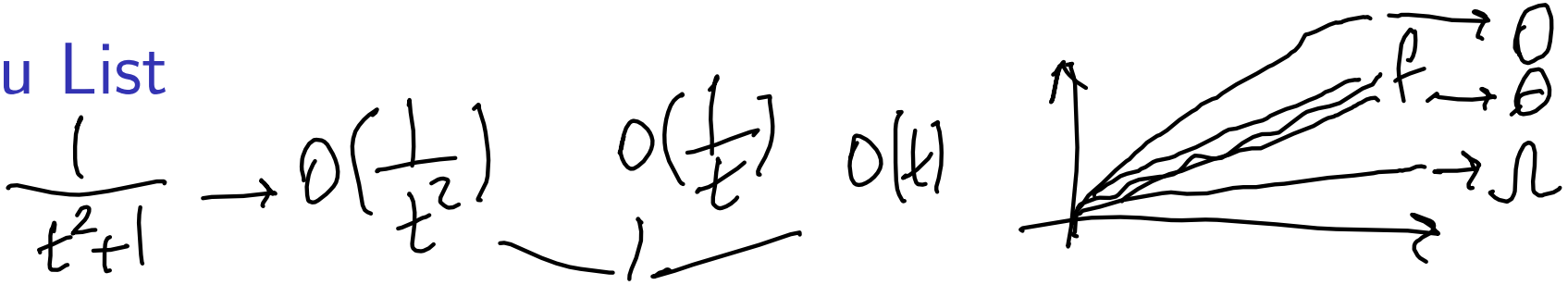
Introduction



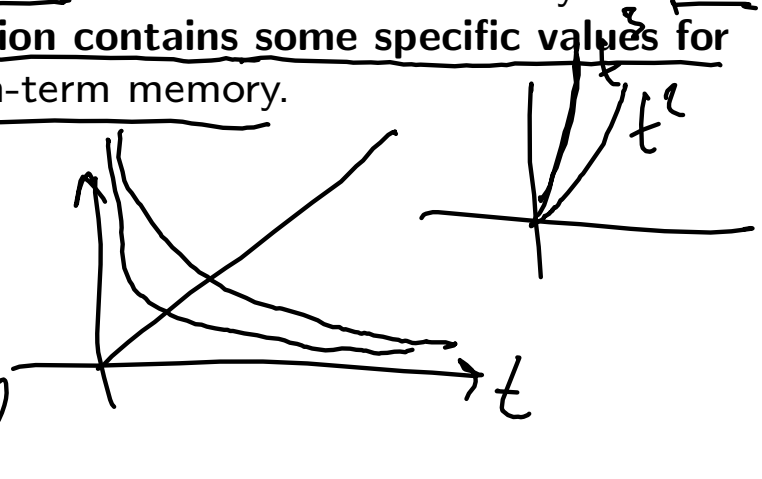
- Tabu search is a **meta-algorithm** which can be used with other metaheuristic optimization algorithms, such as local search or PSO.
- It was proposed in 1986 [1] and formalized in 1989 [2, 3].
- Its idea is simple; it keeps **a record of the previously searched candidate solutions** so that the algorithm does not check them again redundantly.
- In other words, those already searched candidate solutions are **taboo (tabu)** to be searched again.



Tabu List

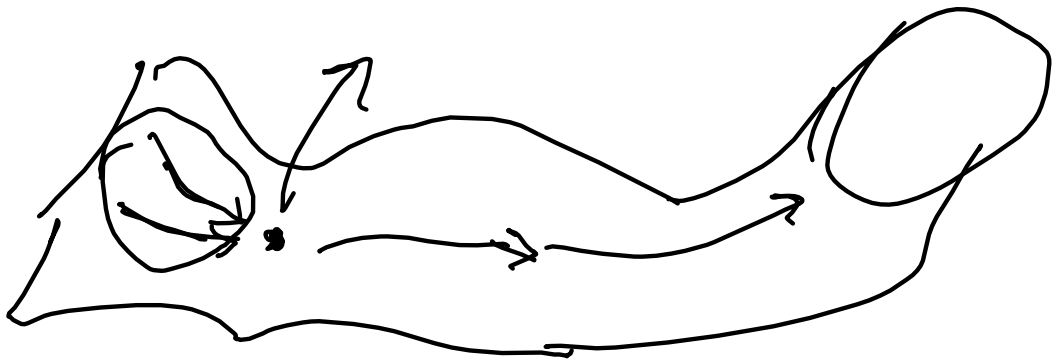


- We have a tabu list which is a memory of previous candidate solutions which we have searched for.
- Tabu search uses space/memory as a trade-off with not checking redundant solutions.
- This list has a maximum length not to spend too much memory. If the length of the tabu list is ℓ, the tabu list contains the last ℓ candidate solutions and does not repeat them.
- We can have several tabu lists, with different lengths, named **short-term memory**, **medium-term memory**, and **long-term memory**. These memories have different lengths where the list of the long-term memory has more length.
- Based on the qualities of candidate solutions and/or **pre-defined rules**, we can put any visited candidate solution in one or several of these lists. The rules are defined by the user and the problem. For example, if **a candidate solution contains some specific values for some specific features**, we put them in the medium-term memory.

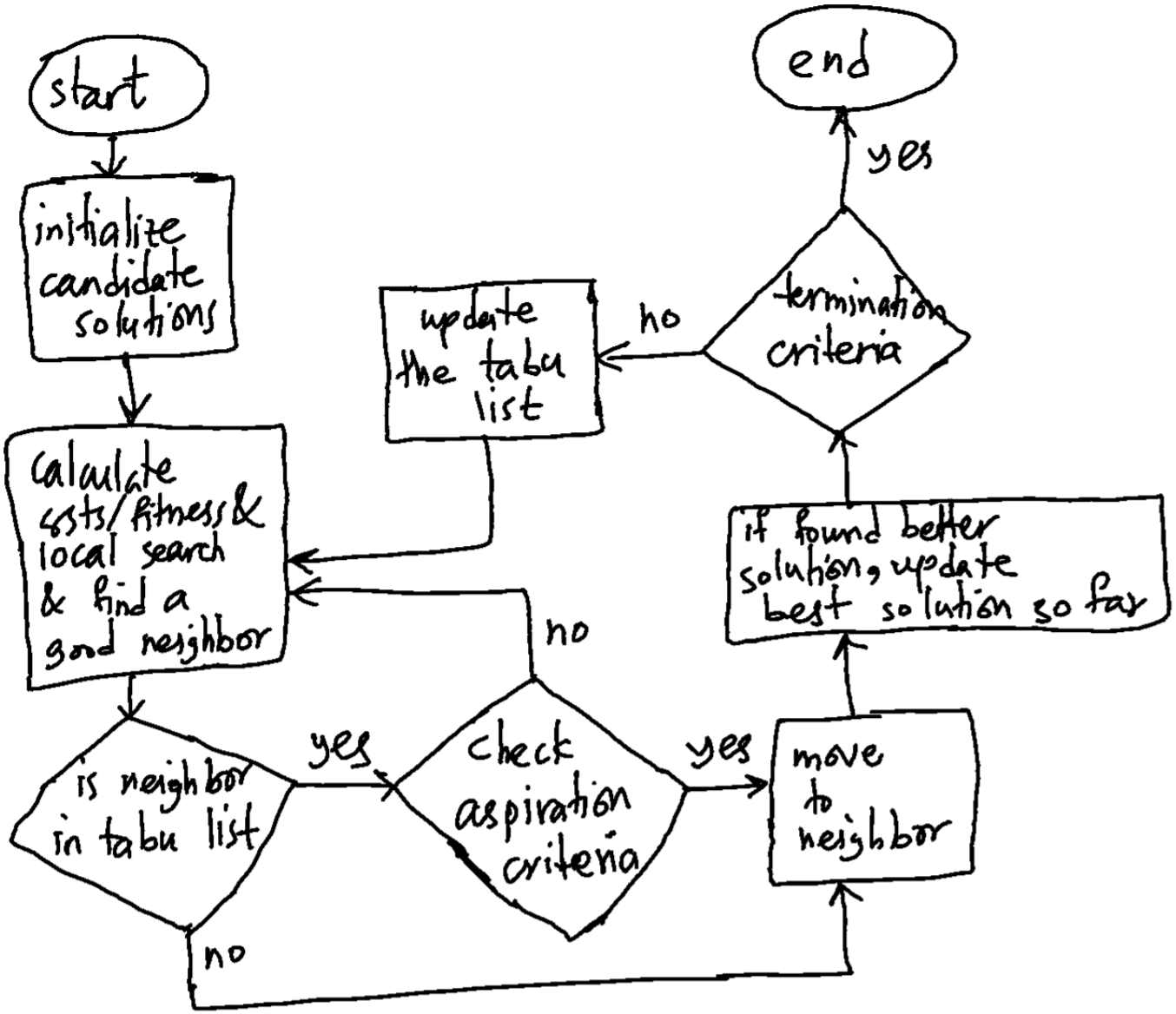


Aspiration Criteria

- When using tabu search, we can have **aspiration criteria** which are rules which allow a tabu move (searching an already checked candidate solution) to be accepted **if it satisfies some condition**.
- For example, a common aspiration criterion is to accept a tabu move **if its cost function is the best solution found so far.**
- In other words, during optimization, if we **do not find good solutions for a while**, it means that we have got stuck in some bad regions of the optimization landscape. In these cases, we can go and **continue searching from one of the tabu candidate solution which had the best solution so far**. From that point, we continue the search but in another direction.



Tabu Search Flowchart



Tabu Search Algorithm

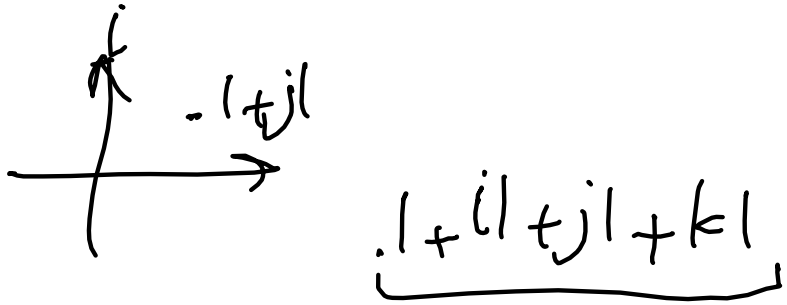
Algorithm Tabu Search

Initialize the candidate solutions $\{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n\}$ and tabu list $\mathcal{T} \leftarrow \varnothing$
while *not converged* **do**
    $\boldsymbol{x}^\dagger \leftarrow$ Perform metaheuristic optimization such as local search
    if $\boldsymbol{x}^\dagger \in \mathcal{T}$ **then**
        if *aspiration criteria are satisfied* **then**
            $\boldsymbol{x} \leftarrow \boldsymbol{x}^\dagger$
        else
            continue the loop (go to next loop)
    else
        $\boldsymbol{x} \leftarrow \boldsymbol{x}^\dagger$
    if $f(\boldsymbol{x}) <$ *best cost so far* **then**
        best cost so far $\leftarrow f(\boldsymbol{x})$
        best solution so far $\leftarrow \boldsymbol{x}$
    if *termination criteria are satisfied* **then**
        Terminate the loop
    else
        $\mathcal{T} \leftarrow \mathcal{T} \cup \boldsymbol{x}$
Return the best solution so far

Acknowledgment

- Some slides of this slide deck are inspired by teachings of Prof. Saeed Sharifian at the Amirkabir University of Technology, Department of Electrical Engineering.
- Some papers and books about Tabu search: [4, 5, 6]

References



[1] F. Glover, "Future paths for integer programming and links to artificial intelligence," *Computers & operations research*, vol. 13, no. 5, pp. 533–549, 1986.

[2] F. Glover, "Tabu search–part i," *ORSA Journal on computing*, vol. 1, no. 3, pp. 190–206, 1989.

[3] F. Glover, "Tabu search–part ii," *ORSA Journal on computing*, vol. 2, no. 1, pp. 4–32, 1990.

[4] F. Glover, "Tabu search: A tutorial," *Interfaces*, vol. 20, no. 4, pp. 74–94, 1990.

[5] F. Glover and M. Laguna, *Tabu search*.
Springer, 1998.

[6] M. Gendreau and J.-Y. Potvin, "Tabu search," *Search methodologies: introductory tutorials in optimization and decision support techniques*, pp. 165–186, 2005.