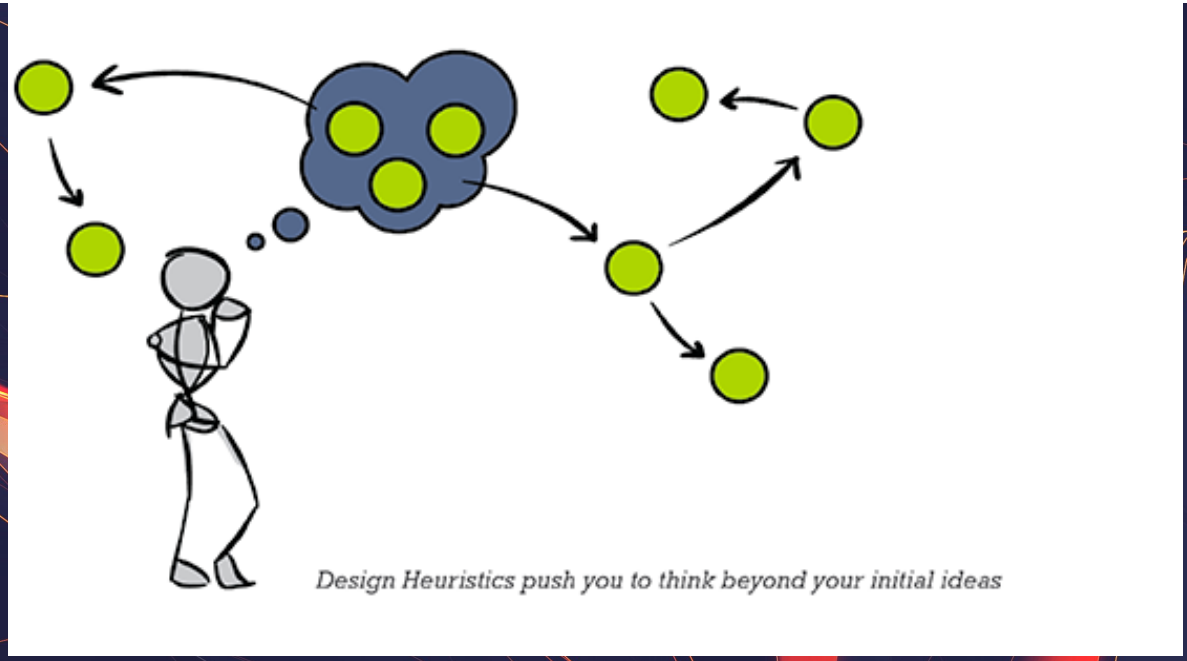



# SINGLE SOLUTION METAHEURISTICS

## BASICS AND LOCAL SEARCH

*Prof. Eduardo Pécora*

# SOLUTION'S REPRESENTATION

*Prof. Eduardo Pécora*

# SOLUTION'S REPRESENTATION

- How should I represent my solution space?
- How should I represent a single solution?
- Is it Binary?
- Mixed?
- Continuos?
- Graph?
- Is there a set of variables more important than others? (Structural variables)

## CONTINUOUS VARIABLES

$$X_i \in \mathbb{R}$$

$$X_i \in [a, b] \in \mathbb{R}$$

$$X_i = \{1.4; 3.14; 3.1; 5.6; 2.8; 768; 99\}$$

## BINARY VARIABLES

$$X_i \in \mathbb{B} = \{0,1\}$$

$$X_i = \{0; 0; 1; 1; 1; 0; 0; 0\}$$

## INTEGER VARIABLES

$$X_i \in \mathbb{Z}$$

$$X_i = \{1; 76; 0; -1; 16; -157; 13; 67\}$$

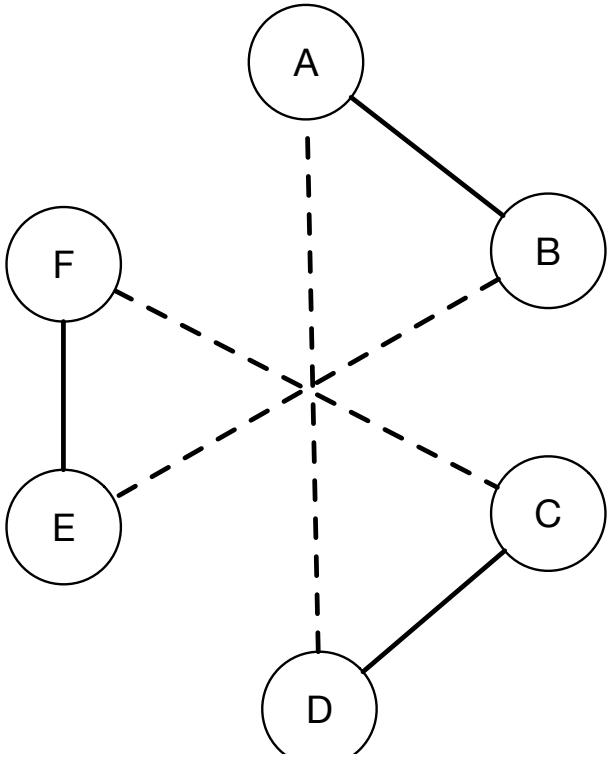
## SOLUTION AS A SEQUENCE

$$S = \{A, B, C, R, Y, T\}$$

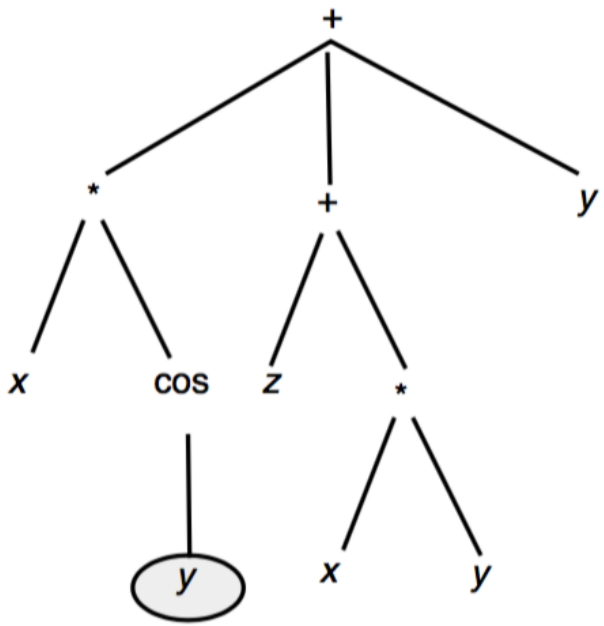
$$S = \{1,2,3,4,5,6\}$$

$$S = \{Curitiba, Carlopolis, Barretos, Manaus\}$$

# SOLUTION AS GRAPH



# SOLUTION AS TREE




$$x \cdot Cos(y) + (z + x \cdot y) + y$$

# SOLUTION'S REPRESENTATION

**YOU MUST CHOSE THE BEST REPRESENTATION FOR THE SOLUTION OF YOUR PROBLEM !**

# S-METAHEURISTICS

*Prof. Eduardo Pécora*


SINGLE SOLUTION
METAHEURISTICS


# NEIGHBOURHOODS

*Prof. Eduardo Pécora*

S

S<->S

Flipped Classroom

# WORK IN PAIRS AND ANSWER THESE QUESTIONS!

- For each problem of Lab 1 you already proposed a heuristic to solve it. Now think back and answer these questions for each Heuristic you have proposed. (S)
  - How did I represent my solution space?
  - How did I represent a single solution?
  - Are they Binary, Integer, Mixed, Continuous, Graph?
- Discuss with a colleague and check your answers! (S <-> S)

# HIGH LEVEL ALGORITHM FOR A S-METAHEURISTICS

**Algorithm 2.1** High-level template of S-metaheuristics.

**Input:** Initial solution $s_0$.
$t = 0$;
**Repeat**
  /* Generate candidate solutions (partial or complete neighborhood) from $s_t$ */
  Generate($C(s_t)$) ;
  /* Select a solution from $C(s)$ to replace the current solution $s_t$ */
  $s_{t+1}$ = Select($C(s_t)$) ;
  $t = t + 1$ ;
**Until** Stopping criteria satisfied
**Output:** Best solution found.

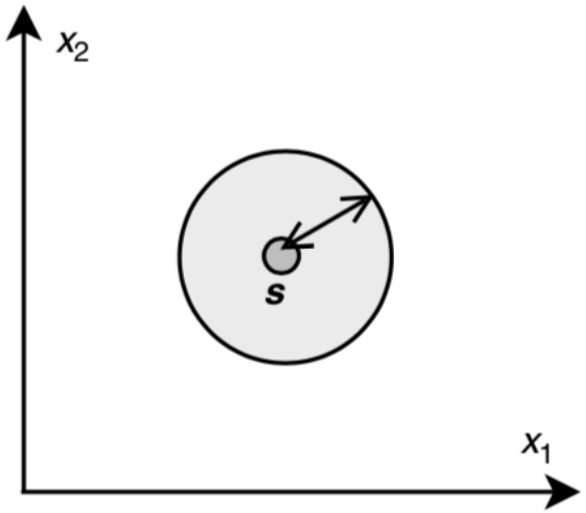
# NEIGHBOURHOOD

**Definition 2.1 Neighborhood.** *A neighborhood function $N$ is a mapping $N : S \rightarrow 2^S$ that assigns to each solution $s$ of $S$ a set of solutions $N(s) \subset S$.*

- The main property that must characterize a neighborhood is *locality*.
- Locality is the effect on the solution when performing the move (perturbation) in the representation.
- When small changes are made in the representa- tion, the solution must reveal small changes. In this case, the neighborhood is said to have a **Strong locality.**
- **Weak locality** is characterized by a large effect on the solution when a small change is made in the representation.
- In the extreme case of weak locality, the search will converge toward a random search in the search space.

# CONTINUOUS NEIGHBOURHOOD

**Definition 2.2** *The neighborhood $N(s)$ of a solution $s$ in a continuous space is the ball with center $s$ and radius eaual to $\epsilon$ with $\epsilon > 0$.*




The circle represents the neighborhood of *s* in a continous problem with two dimensions.

**Figure 2.2**

# DISCRETE NEIGHBOURHOOD

**Definition 2.3** *In a discrete optimization problem, the neighborhood $N(s)$ of a solution $s$ is represented by the set $\{s' / d(s', s) \leq \epsilon\}$, where $d$ represents a given distance that is related to the move operator.*

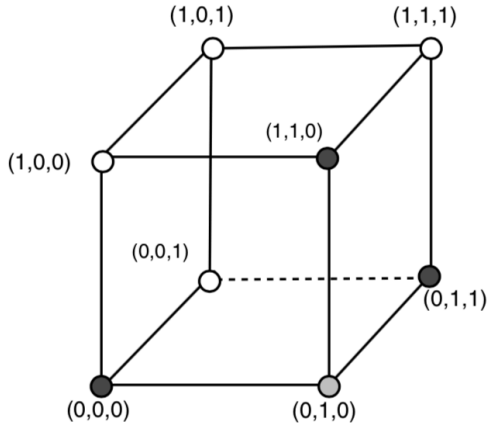



**Figure 2.2**

# DISCRETE NEIGHBOURHOOD

**Definition 2.3** *In a discrete optimization problem, the neighborhood $N(s)$ of a solution $s$ is represented by the set $\{s'/d(s', s) \leq \epsilon\}$, where $d$ represents a given distance that is related to the move operator.*

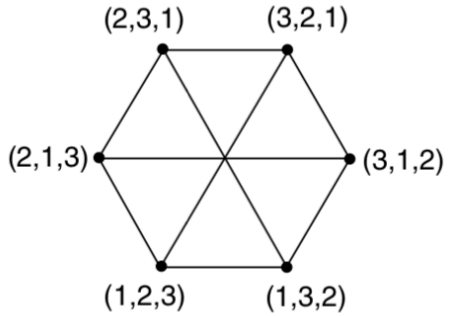



**FIGURE 2.3** An example of neighborhood for a permutation problem of size 3. For instance, the neighbors of the solution (2, 3, 1) are (3, 2, 1), (2, 1, 3), and (1, 3, 2).

# SS

# NEIGHBOURHOOD – EXERCISES

1) $S_i \in \mathbb{B}^5$

2) $S_i \in \mathbb{N}^5$

3) $S_i \in \mathbb{R}^4$

4) $S_i \in \mathbb{B}^3 \mathtt{x} \mathbb{R}^2$

5) $S_i \in \mathbb{B}^3 \mathtt{x} \mathbb{Z}^2$

*Write down a neighbourhood for each of these sets.*

*Which is the distance function?*

*You should think about the size and the locality of your* ***Neighbourhood!***

# T <-> C

# NEIGHBOURHOOD – EXERCISES

## *Ordered Based Problems*

$$s_1 = (4, 3, 1, 2)$$

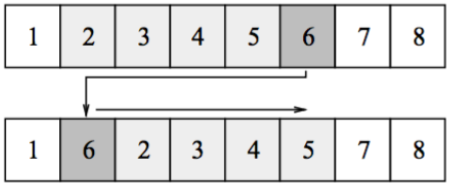



**FIGURE 2.7** Insertion operator.

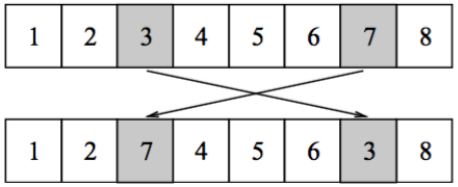



**FIGURE 2.8** Exchange operator.

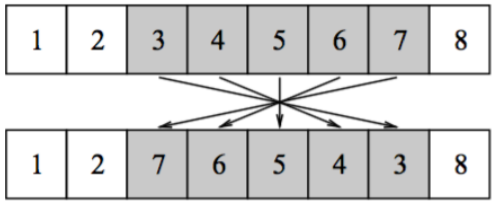



**FIGURE 2.9** Inversion operator.

*Which parameters sets these neighbourhoods uses?*

*Write a pseudo-code for each of these neighbourhoods.*

*Classify them as diversification-intensification!*

*Think about the Sizes?*

*Are you able to find another Neighbourhood?*

# NEIGHBOURHOOD – INSERTION

*Ordered Based Problems*

$$s_1 = (4, 3, 1, 2)$$

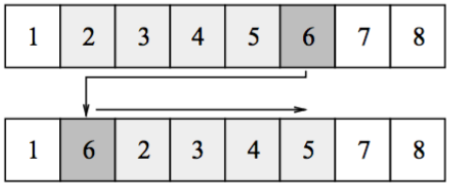



**FIGURE 2.7** Insertion operator.

*Which parameters sets these neighbourhoods uses?*

*Write a pseudo-code for each of these neighbourhoods.*

*Classify them as diversification-intensification!*

*Think about the Sizes?*

# NEIGHBOURHOOD – EXCHANGE

*Ordered Based Problems*

$$s_1 = (4, 3, 1, 2)$$

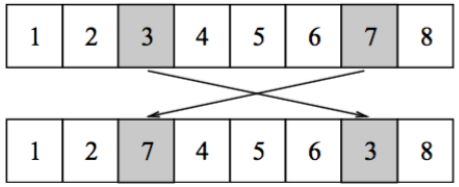



**FIGURE 2.8** Exchange operator.

*Which parameters sets these neighbourhoods uses?*

*Write a pseudo-code for each of these neighbourhoods.*

*Classify them as diversification-intensification!*

*Think about the Sizes?*

# NEIGHBOURHOOD – INVERSION

*Ordered Based Problems*

$$s_1 = (4, 3, 1, 2)$$

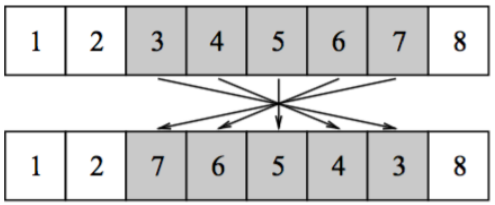



**FIGURE 2.9** Inversion operator.

*Which parameters sets these neighbourhoods uses?*

*Write a pseudo-code for each of these neighbourhoods.*

*Classify them as diversification-intensification!*

*Think about the Sizes?*

# NEIGHBOURHOOD – EXERCISES

*Ordered Based Problems*

$$s_1 = (4, 3, 1, 2)$$

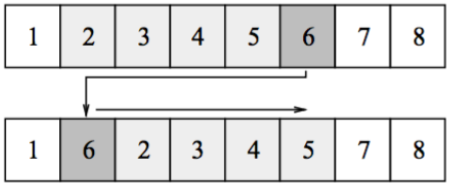



**FIGURE 2.7** Insertion operator.

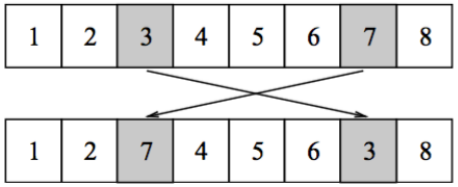



**FIGURE 2.8** Exchange operator.

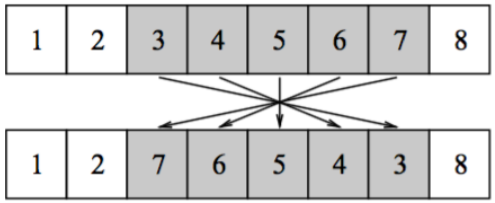



**FIGURE 2.9** Inversion operator.

*Are you able to find another Neighbourhood?*

# ORDERED BASED PROBLEMS: INSERTION

S0 = (1,2,3,4)

| | | | | |
|---|---|---|---|---|
| **Insert in the first position** | (4,1,2,3) | (3,1,2,4) | (2,1,3,4) | |
| **Insert in the second position** | (1,4,2,3) | (1,3,2,4) | | |
| **Insert in the third position** | (1,2,4,3) | | | (2,1,3,4) |
| **Insert in the fourth position** | | | (1,3,2,4) | (2,3,1,4) |
| **Insert in the fifth position** | | (1,2,4,3) | (1,3,4,2) | (2,3,4,1) |

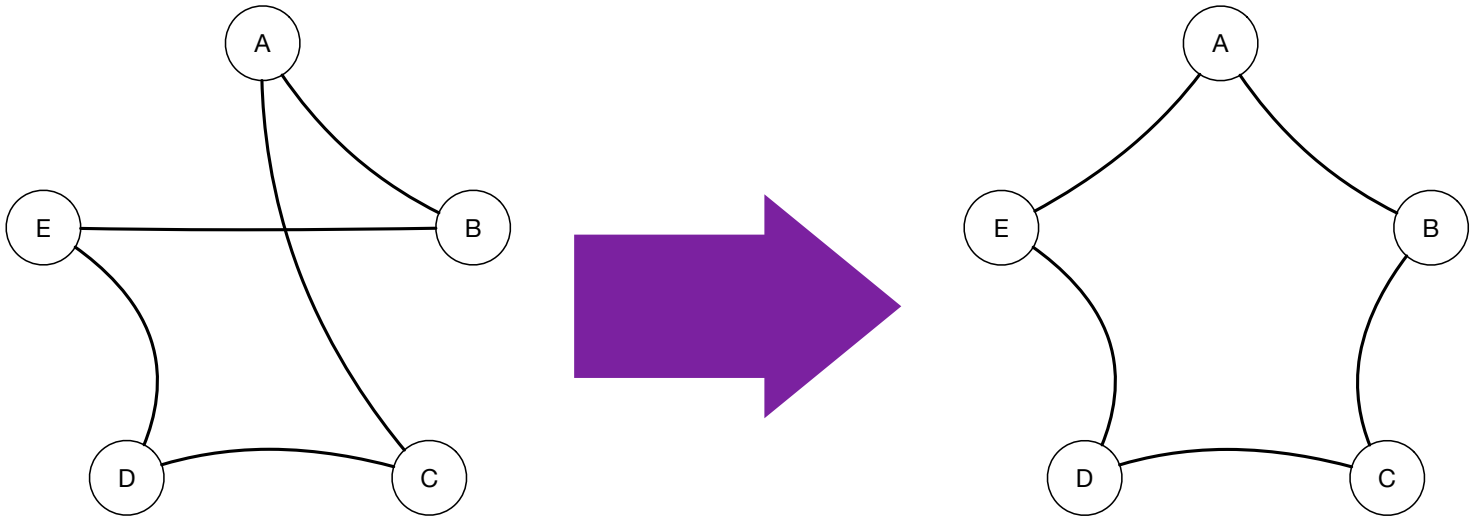
# ORDERED BASED PROBLEMS: INSERTION

(1,2,3,4)

| | | | | |
|---|---|---|---|---|
| **Insert in the first position** | (4,1,2,3) | (3,1,2,4) | (2,1,3,4) | |
| **Insert in the second position** | (1,4,2,3) | (1,3,2,4) | | |
| **Insert in the third position** | (1,2,4,3) | | | (2,1,3,4) |
| **Insert in the fourth position** | | | (1,3,2,4) | (2,3,1,4) |
| **Insert in the fifth position** | | (1,2,4,3) | (1,3,4,2) | (2,3,4,1) |

# MORE EXAMPLES

# NODE SWAP



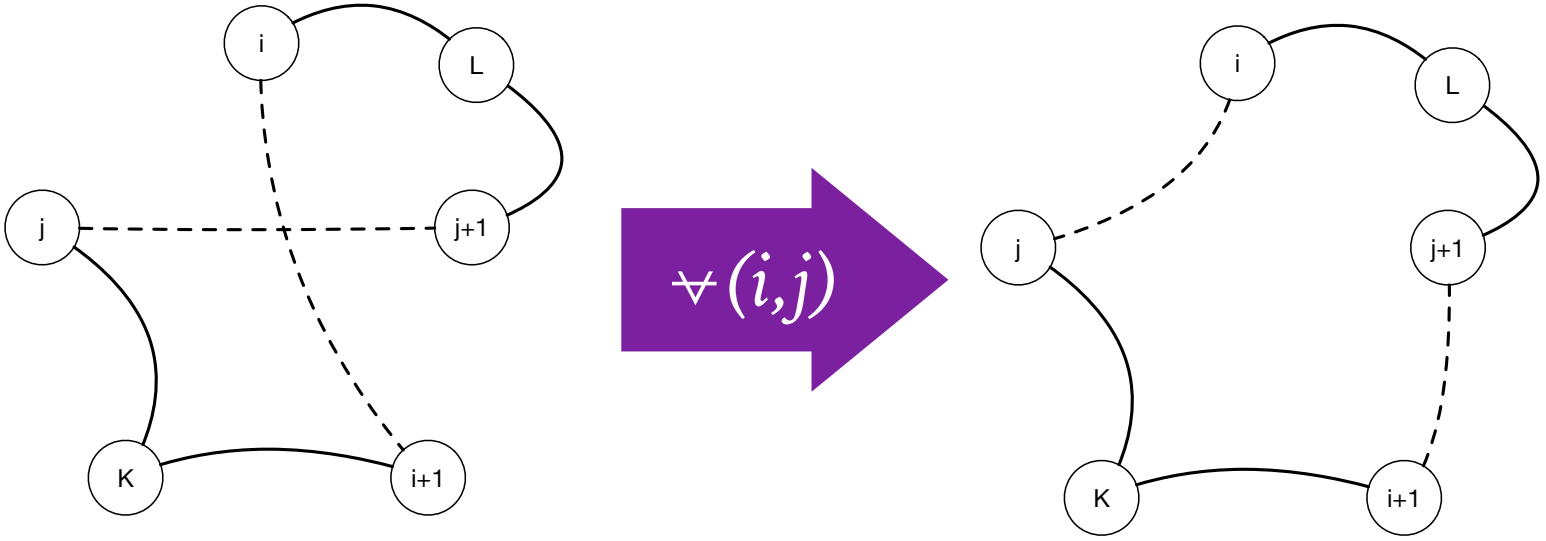

$(E,B,A,C,D)$ $(E,A,B,C,D)$

*Classify it as diversification-intensification!*

*Size?*

*Parameters?*

# 2-OPT



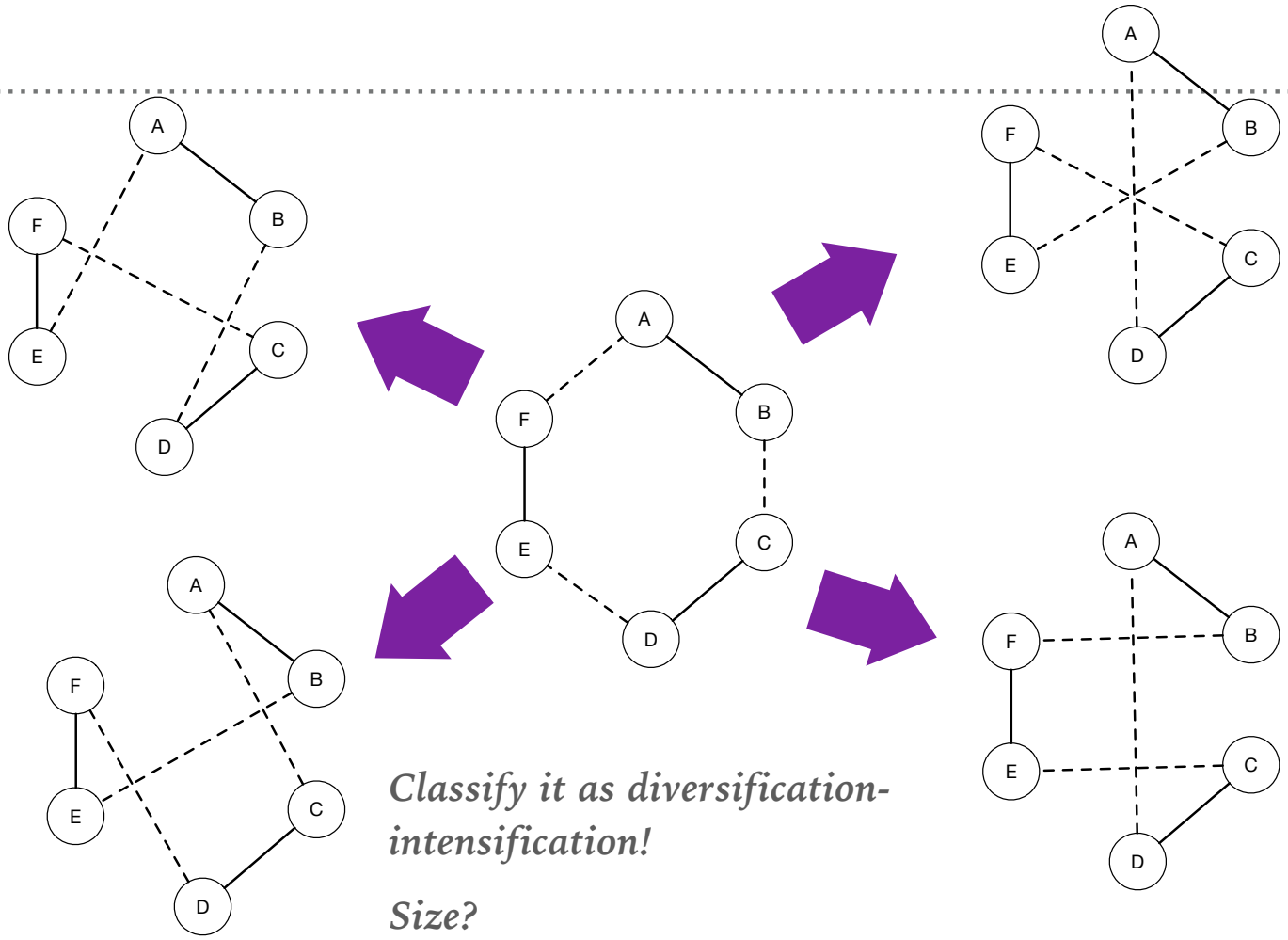

$(i, i+1)$ $(j, j+1)$ $(i, j)$ $(i+1, j+1)$

*Classify it as diversification-intensification!*

*Size?*

*Parameters?*

# 3-OPT

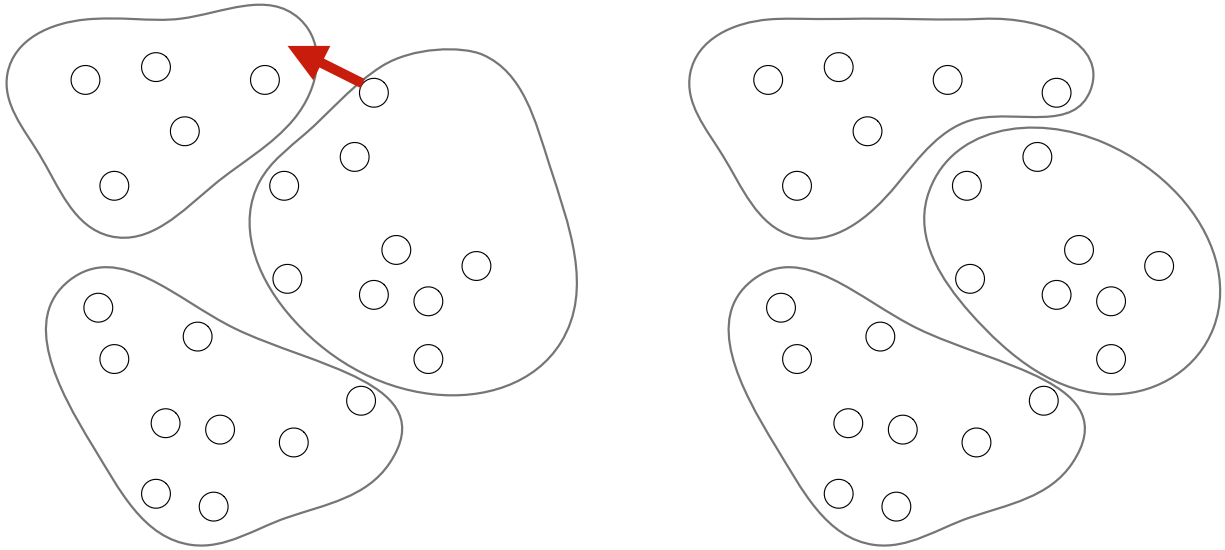



*Classify it as diversification-intensification!*

*Size?*

*Parameters?*

# CLUSTER BASED PROBLEMS – INSERTION

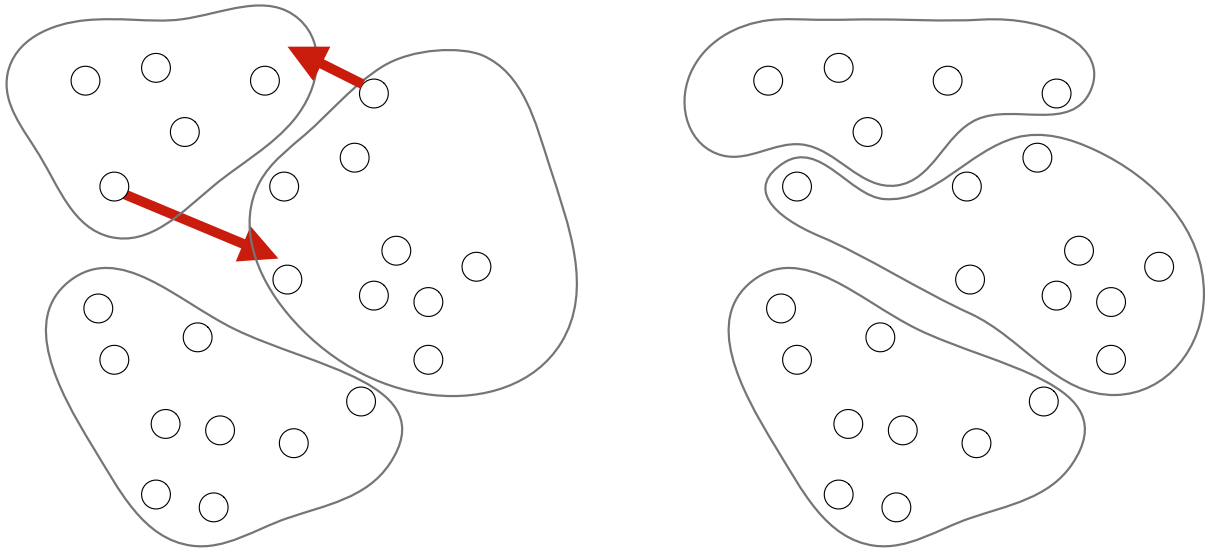


*Classify it as diversification-intensification!*

*Size?*

*Parameters?*

# CLUSTER BASED PROBLEMS – SWAP

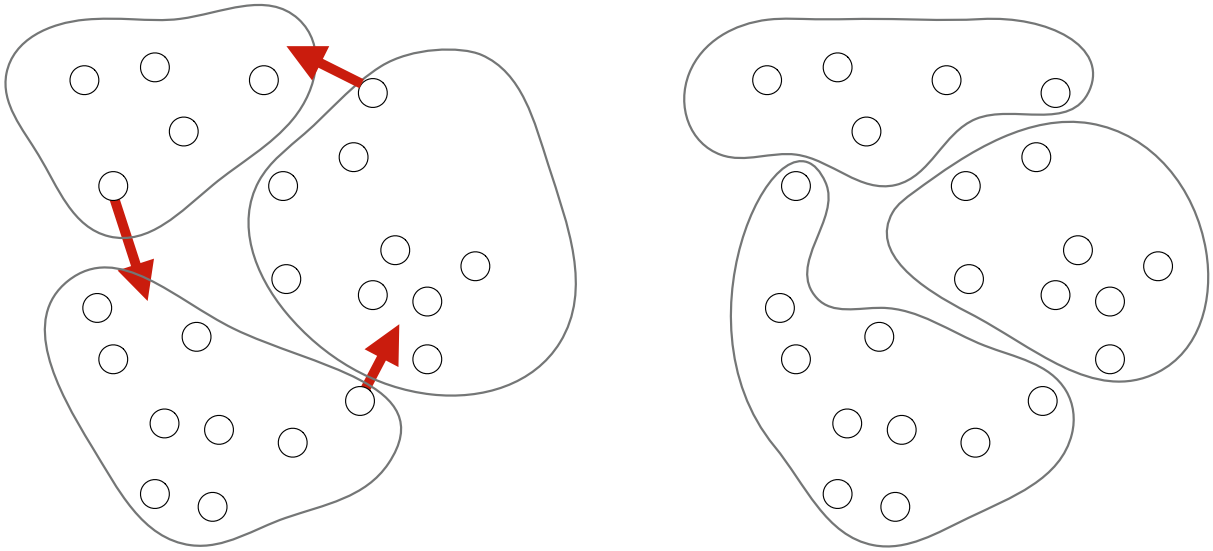


*Classify it as diversification-intensification!*

*Size?*

*Parameters?*

# CLUSTER BASED PROBLEMS – CYCLIC



*Classify it as diversification-intensification!*

*Size?*

*Parameters?*

# VEHICLE ROUTING PROBLEM: EJECTION CHAIN

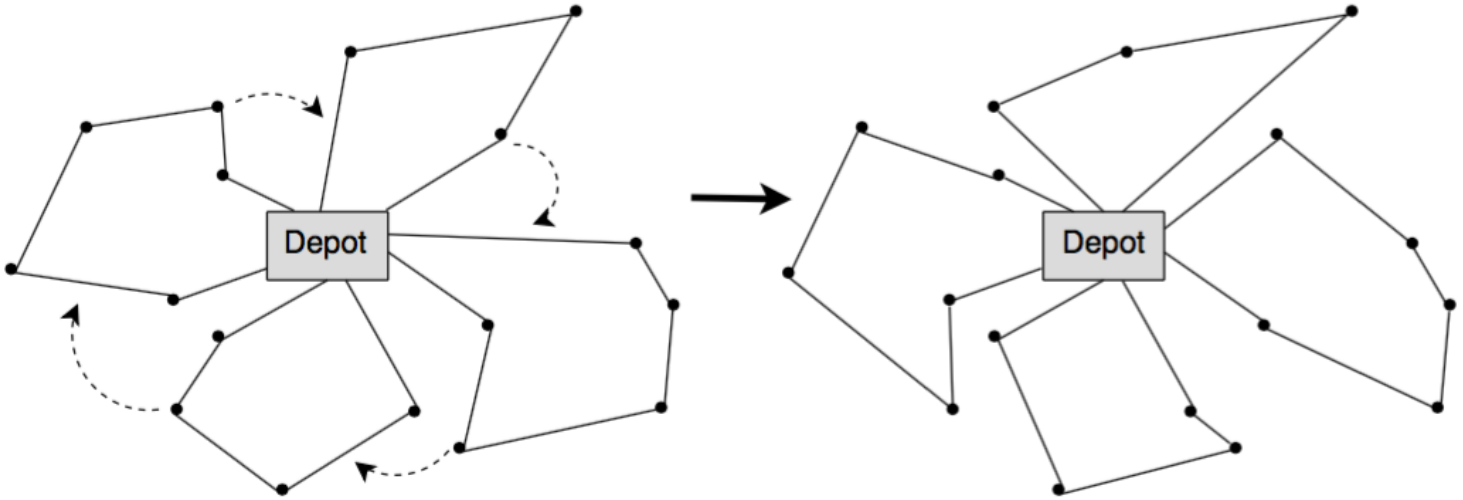



**FIGURE 2.12** A four-level ejection chain for vehicle routing problems. Here, the ejection chain is based on a multinode insertion process.

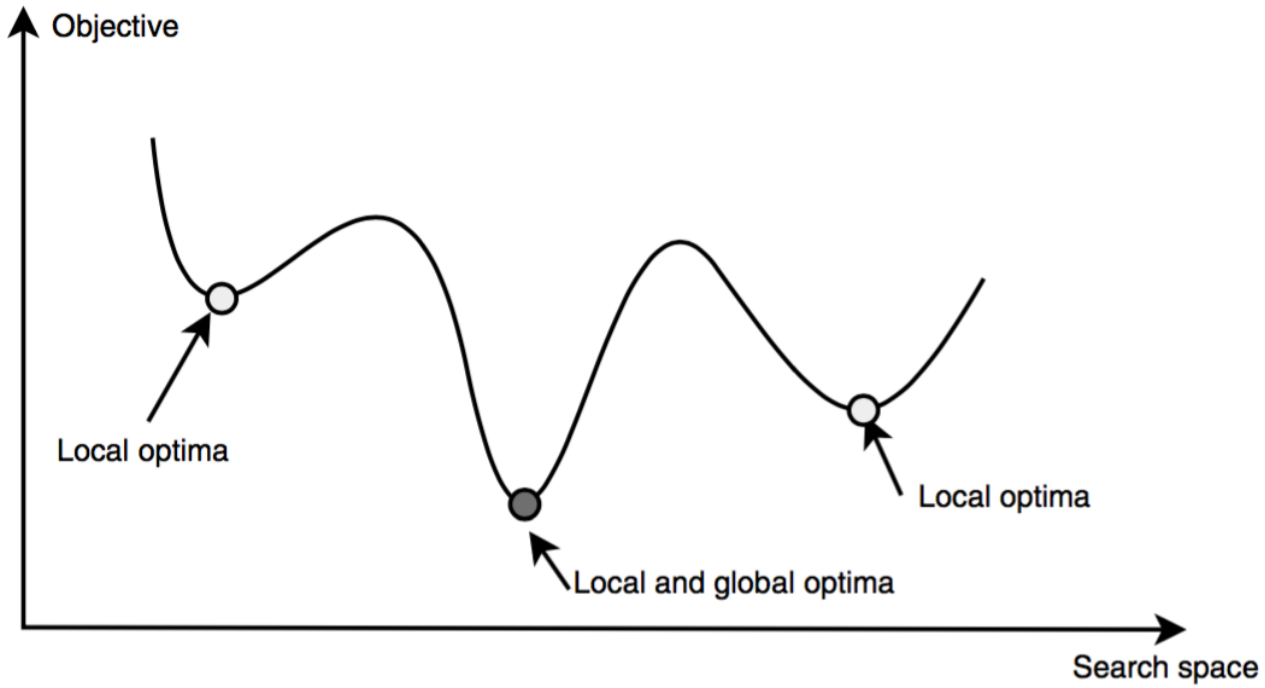
*Classify it as diversification-intensification!*

*Size?*

*Parameters?*

# GLOBAL AND LOCAL OPTIMA POINTS

**Definition 2.4 Local optimum.** *Relatively to a given neighboring function $N$, a solution $s \in S$ is a local optimum if it has a better quality than all its neighbors; that is, $f(s) \leq f(s')^2$ for all $s' \in N(s)$ (Fig. 2.4).*



# CONNECTIVITY

***Initial solution*** $s_0 = 10011$

***N2 - Swap (i, i+1)*** $f(s_0) = 2200$

*01011*

*10011*

*10101*

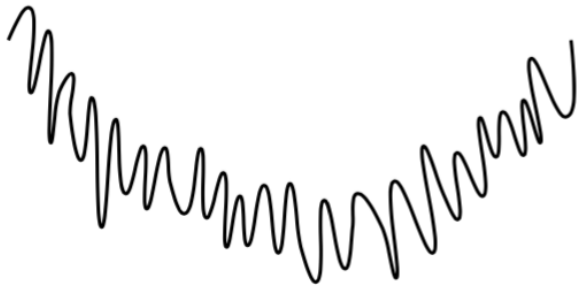
*10011*


***Neighbourhood!!***

1) *Repeating Solutions*
2) *There are solutions which will never be reached*

# LANDSCAPE – MINIMIZATION PROBLEM

**TABLE 2.1 Some Representations of Landscapes in One Dimension Using the Geographical Metaphor**

| | |
|---|---|
| Flat, plain |  |
| Basin, valley |  |
| Rugged plain |  |
| Rugged valley |  |

**Which one is the most difficult to optimize?**

# TASK 2.0
# TO WRAP UP
# TO SUMMARIZE
# TO SUM UP

# WRITE THE DEFINITIONS:

- Combinatorial Optimization
- Heuristic Method
- Exact Method
- Global Optimum
- Neighbourhood
- Connectivity and Locality of a Neighbourhood
- Local Optimum
- Any other concept you remembered and the professor didn't

# LOCAL SEARCH

*Prof. Eduardo Pécora*

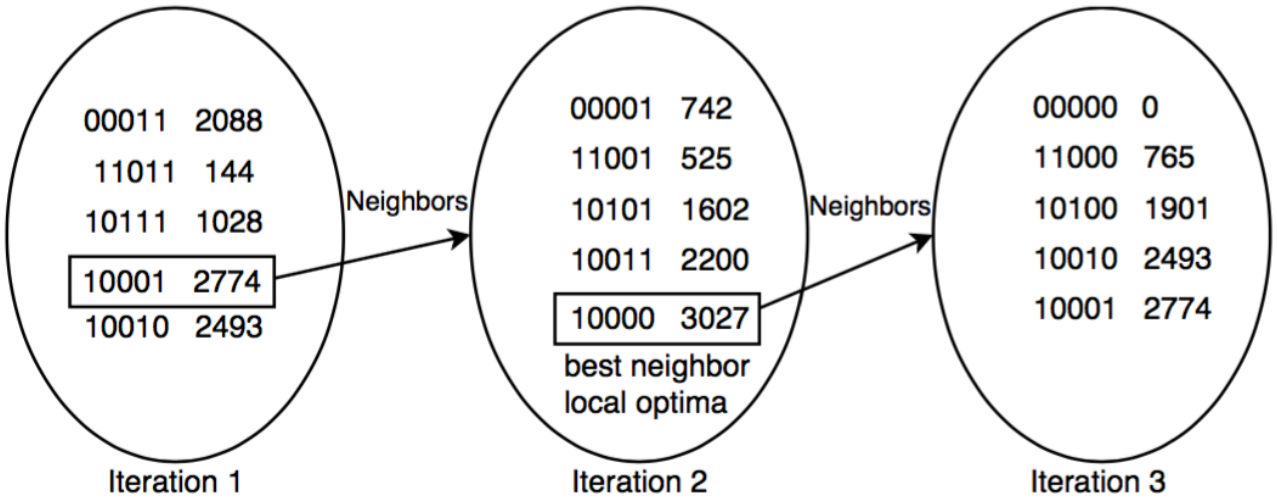
# LOCAL SEARCH

***Initial solution***

$$s_0 = 10011$$

***N1 - Change only one value***

$$f(s_0) = 2200$$



# LOCAL SEARCH ALGORITHM

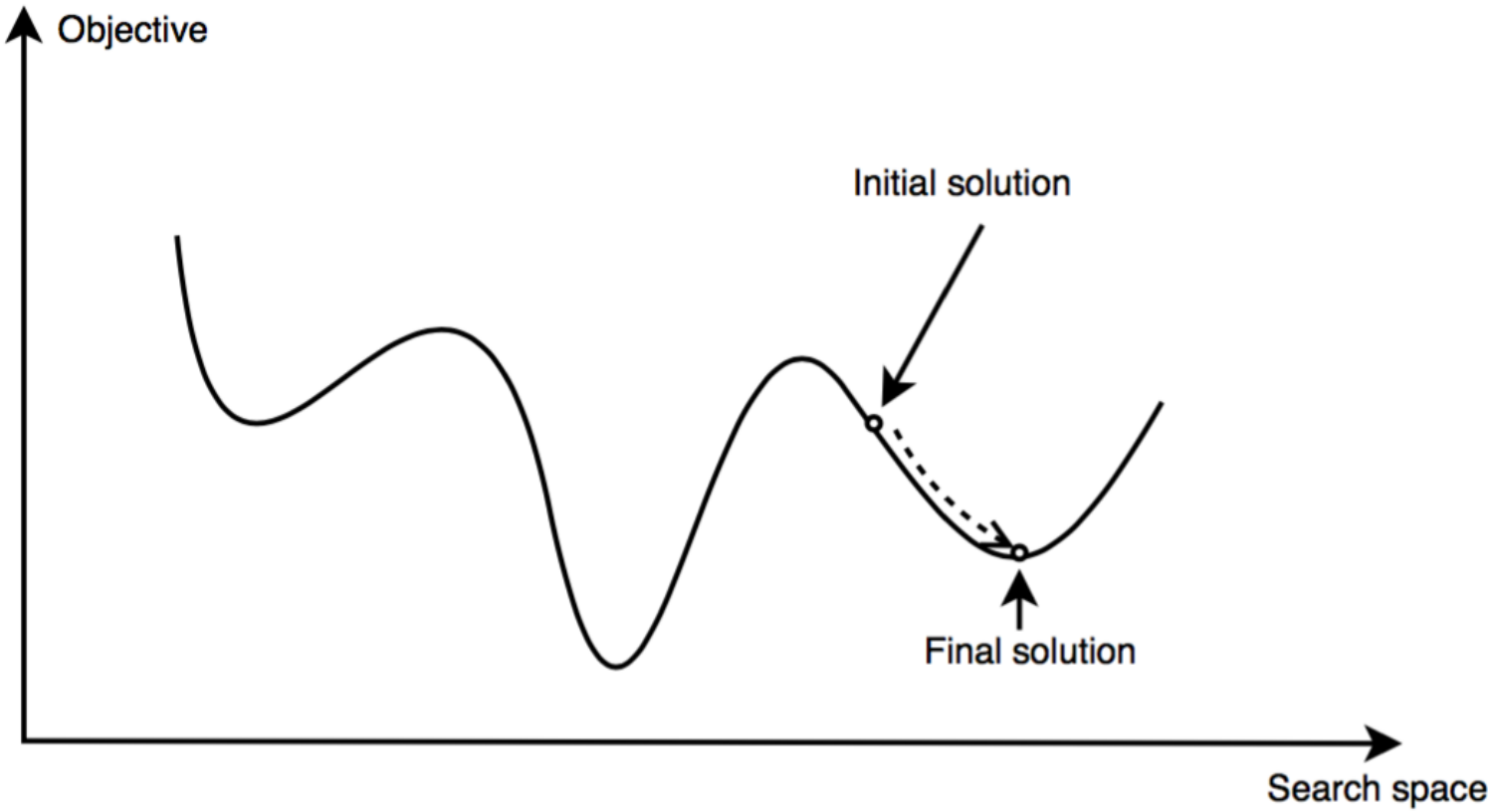
**Algorithm 2.2** Template of a local search algorithm.

$s = s_0$ ; /* Generate an initial solution $s_0$ */
**While** not Termination_Criterion **Do**
  Generate ($N(s)$) ; /* Generation of candidate neighbors */

  **Evaluate all Neighbour** $s' \in N(s)$

  **If** there is no better neighbor **Then** Stop ;
  $s = s'$ ; /* Select a better neighbor $s' \in N(s)$ */
**Endwhile**
**Output** Final solution found (local optima).

# LOCAL SEARCH



# SELECTION OF THE NEIGHBOUR

- Best improvement (steepest decent)
- First improvement
- Random improvement

# SS

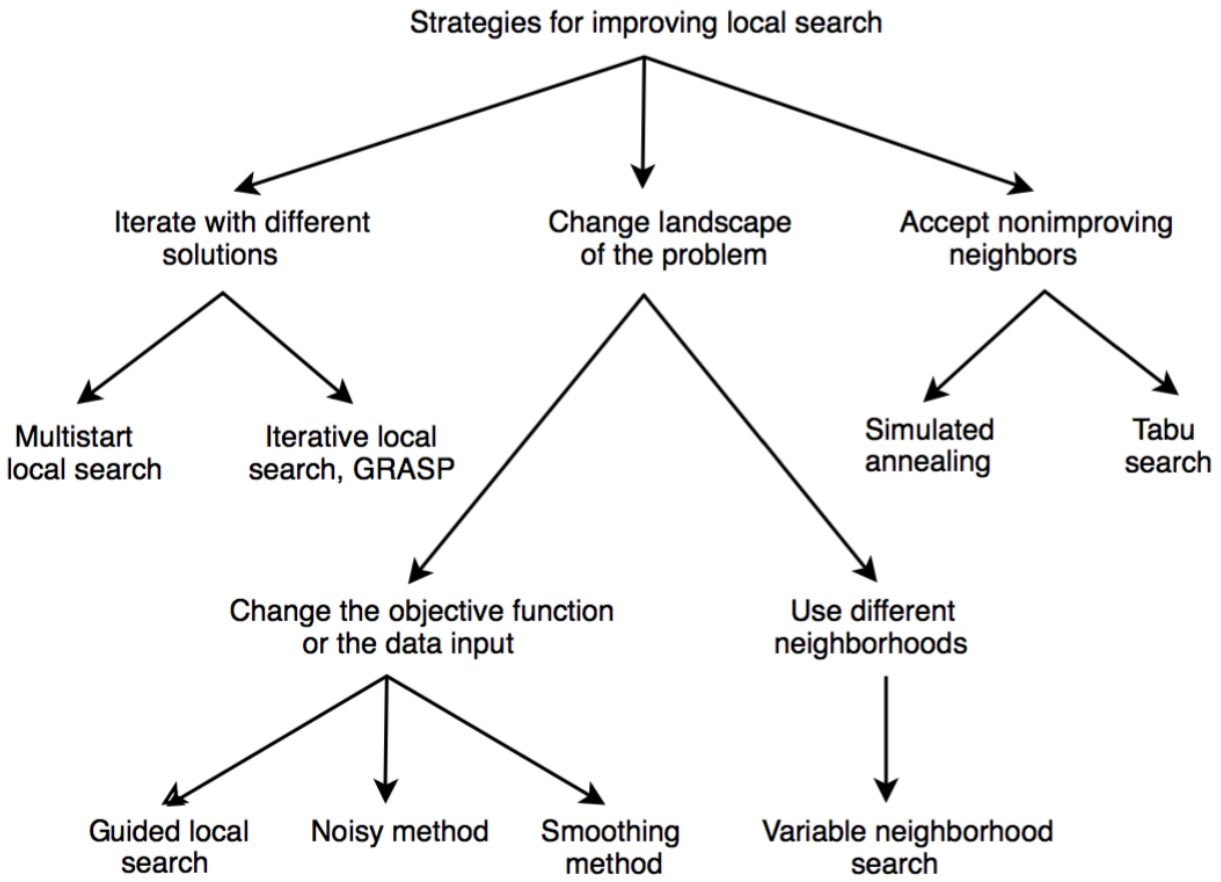
# LOCAL SEARCH – EXERCISES

- Using the Excel Worksheet KnapsackLocalSearch develop a local Search Algorithm
  - Define your Neighbourhood
  - Use the any Solution as S0
  - Perform the local search until the stop criteria

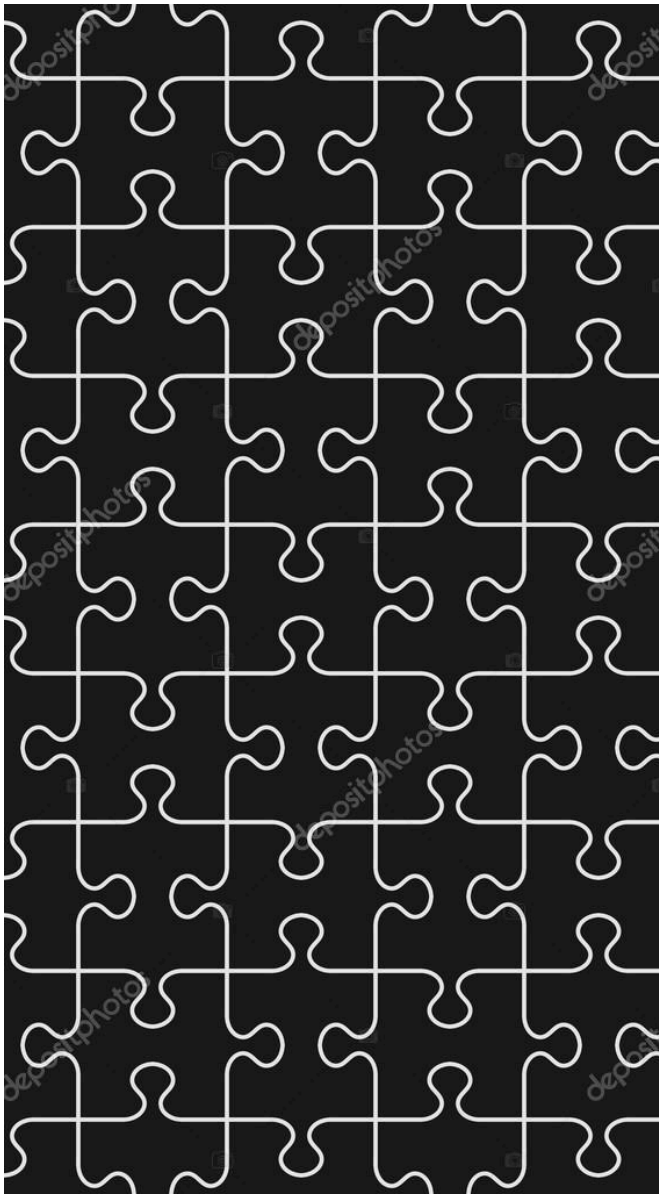
# LOCAL SEARCH – EXERCISES

- Now I have no way out? I am stuck here!
- How can I continue my mission: to search for THE optimum Solution?
- Proposals Please …

# ESCAPING FROM LOCAL OPTIMA



# (S)
# HOME WORK



# LAB 2

**For the knapsack problem**

Program a local Search algorithm and solve the instances given by the professor

You should provide the best solution and its objective function for each instance given.

**(Deliver in Class 4).**