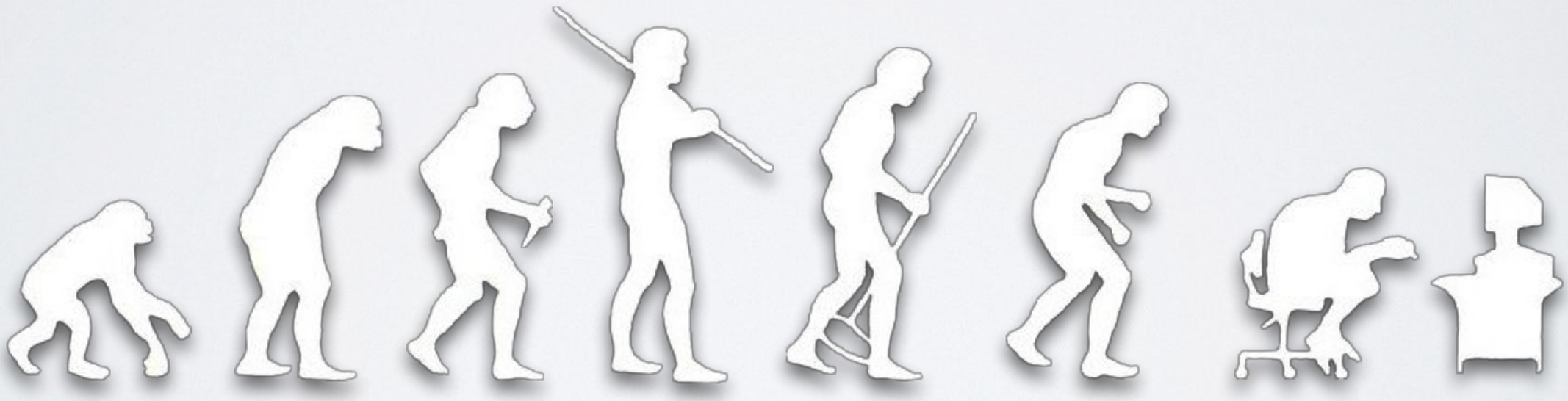
TRAVELER SALESMAN PROBLEM

Group 15

THE ALGORITHM

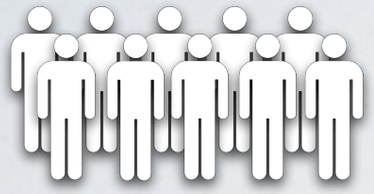
Genetic + 2-Opt



1. Create a new population

First individual is a tour of subsequent cities as they appear in the input cities list.

All the other individuals are random tours.

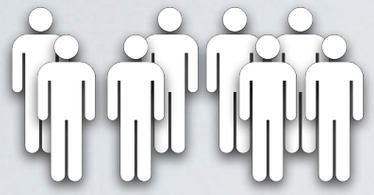


2. Evolution stage

We select the best individual of the population to be saved into the new population.

Then we take random 20% of the population and select the best out of it. This process is executed *two* times and the selected individuals go into the new population.

NEW POPULATION: 



We now have to select two parents to create a child to be inserted into the new population. We generate again a random subgroup of individuals (20%) and choose the best out of it as a parent.
We apply a Crossover function that takes a random segment of the Parent 1 and fills the remaining free slots of the child according to the relative sequence of cities in Parent 2.
We execute 2-Opt algorithm on the child before inserting it into the new population.

Parent 1

Parent 2

Crossover

Child

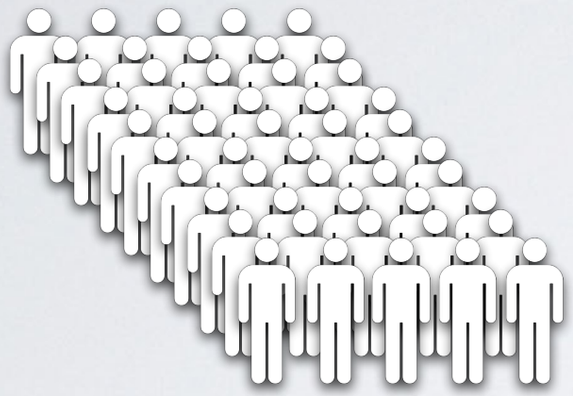
If the final individual is equal to another one already present into the new population, we discard the clone and we repeat the creation process. This is done for a maximum of 15 times for each child creation (in order to avoid slowdown in case of eccessive clones generation). After the 15

2-Opt

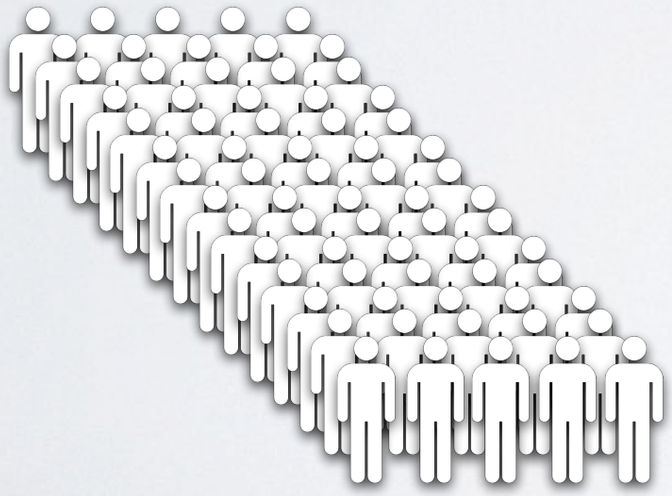
NEW POPULATION:

Mutation

We swap two cities randomly (preserving the inner sub-tour).
For each city of each tour there is a default value of **0.8% of probability** that a mutation occurs.



Every **5 evolutions** we execute a mutation on each child before the execution of the 2-Opt



If our best individual doesn't change within *7 evolutions*, we increase the rate of mutations: once every **2 evolutions** until we find a new best tour

Our work

Tabu

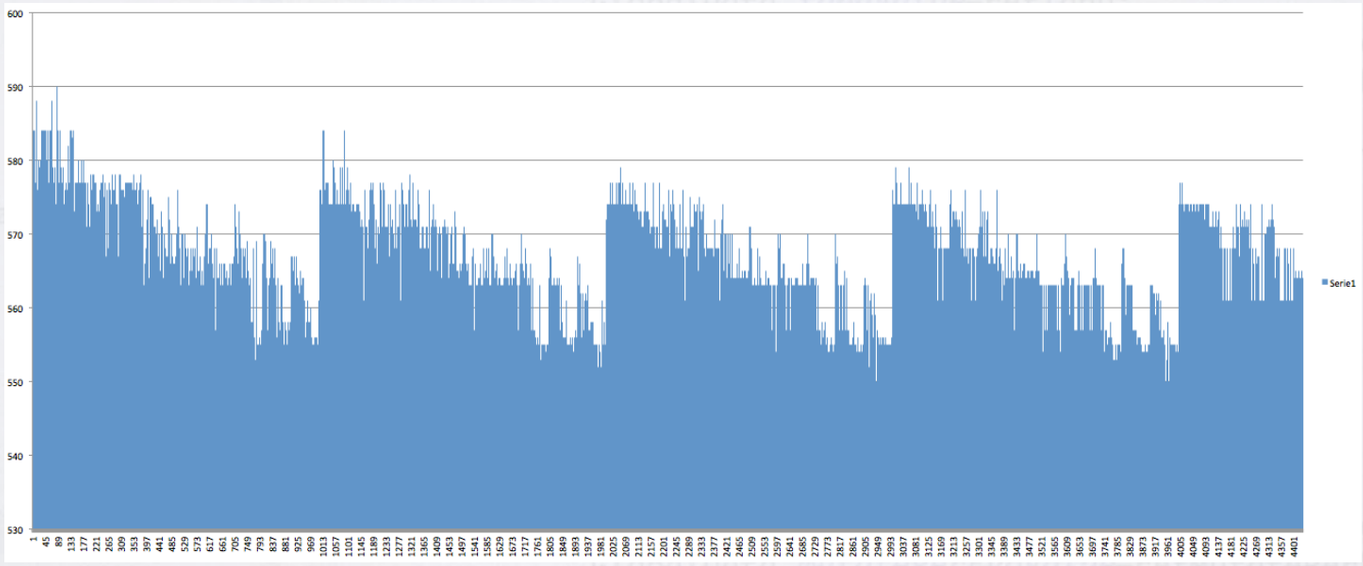
- Improvement of the function *MyGreedyStartSolution*

Tabu + Genetic

- Merging Tabu and Genetic;
- Various techniques tested to handle evolutions and parents selection;
- Crossover functions;

2-OPT + Genetic

- Combine genetic, tabù and 2-opt;
- Deep studying of 2-opt and 3-opt with many different combinations and implementation techniques.
- Management of clones;
- Mutation function parameters;
- Statistics through recursive functions;



Our parameters and results

Population and Evolutions dynamic parameters have been chosen to better perform the algorithm according to the instance dimension and also to be consistent with the time constraints imposed by the project guideline

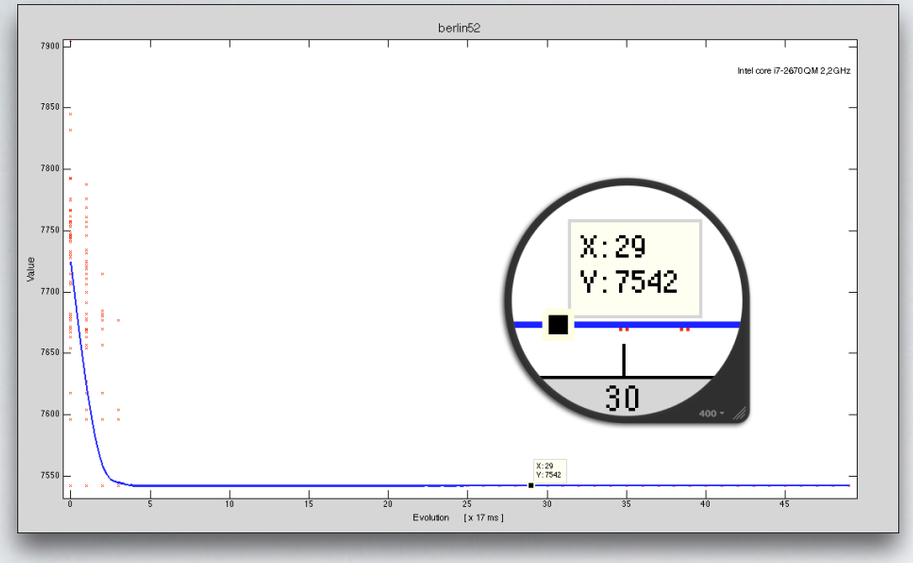
- **Population** size: **50** (0-500 cities), **30** (>500 cities);
- **Evolutions**: **30** (0-160 cities), **150** (160-500 cities), **75** (>500 cities);
- **Mutate rate** (standard): **5** evolutions; mutate rate (fast): **2** evolutions;
- Number of not improving evolutions (**maxAge**) to fasten the mutate rate: **7**;
- **Mutate probability**: **0.8%**;
- **Tournament Size**: **20%** (of the total population size);
- Maximum number of **clone reprocessing** per child: **15**;
- Total number of **iterations** per instance: **10**;

Instance	Mean sol	Min sol	Max sol	Time Best	Time mean	Best known
berlin52	7542	7542	7542	17 ms	510 ms	7542
eil51	426,5	426	427	28 ms	420 ms	426
eli76	538,1	538	540	99 ms	990 ms	538
pr152	73736,4	73682	73818	465 ms	4,65 s	73682
pr1002	266100	263800	268700	4,9 m	4,9 m	259045
rat195	2334	2323	2348	10.3 s	39,7 s	2323

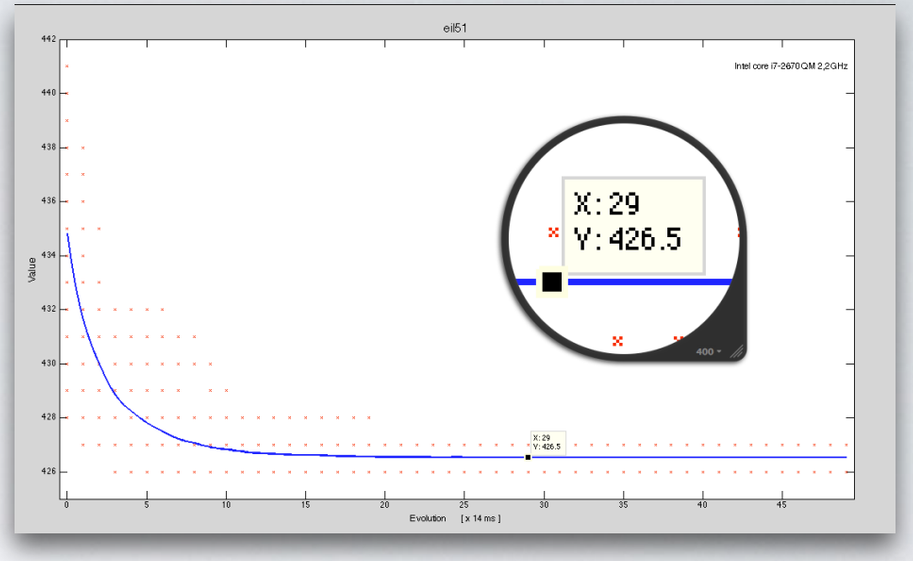
Mean		
Istanze	**Best (%)**	**Time [ms]**
berlin52	0,00	510
eil51	0,12	420
eli76	0,02	990
pr152	0,07	4650
pr1002	2,72	294000
rat195	0,47	39700
Mean	**0,57**	**56711,7**

Best		
Istanze	**Best (%)**	**Time [ms]**
berlin52	0,00	17
eil51	0,00	28
eil76	0,00	99
pr152	0,00	465
pr1002	1,84	294000
rat195	0,00	10300
Mean	**0,31**	**50818,2**

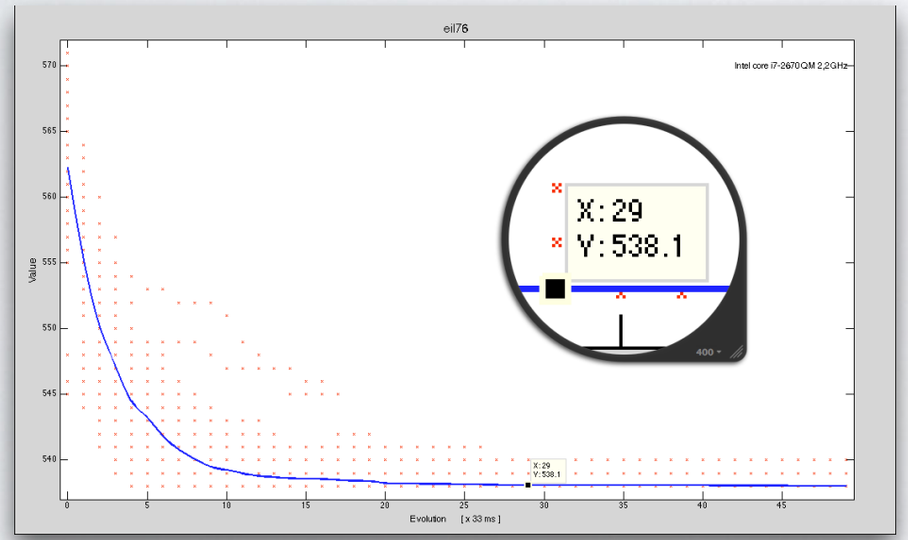
berlin52



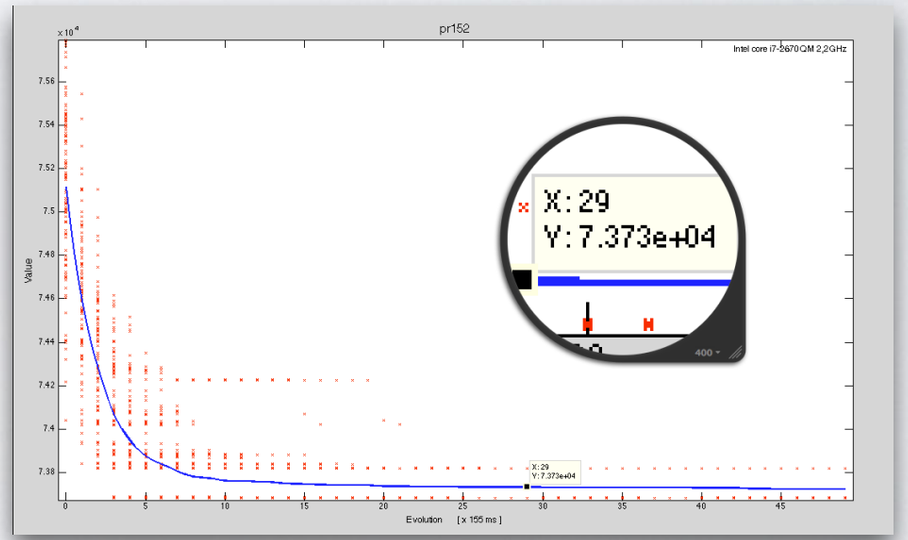
eil51



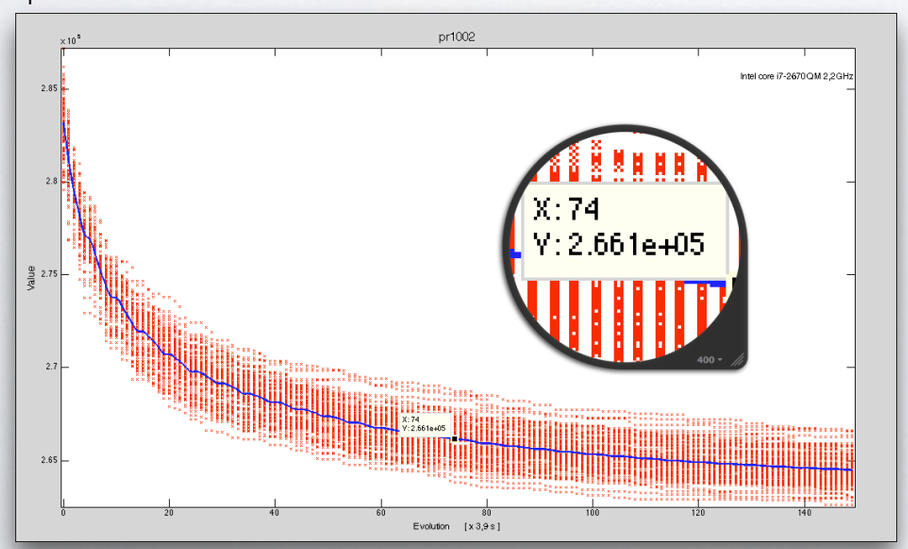
eil76



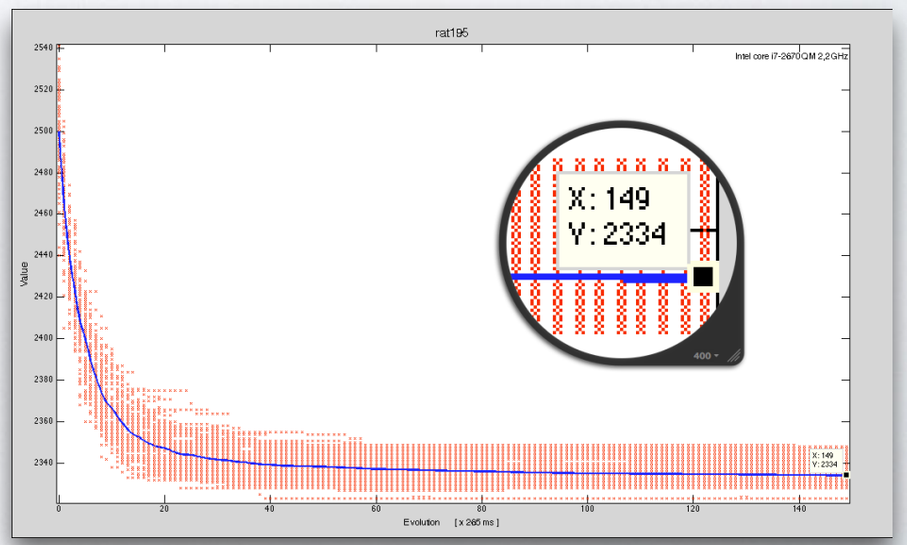
pr152



pr1002



rat195



THANK YOU FOR YOUR ATTENTION