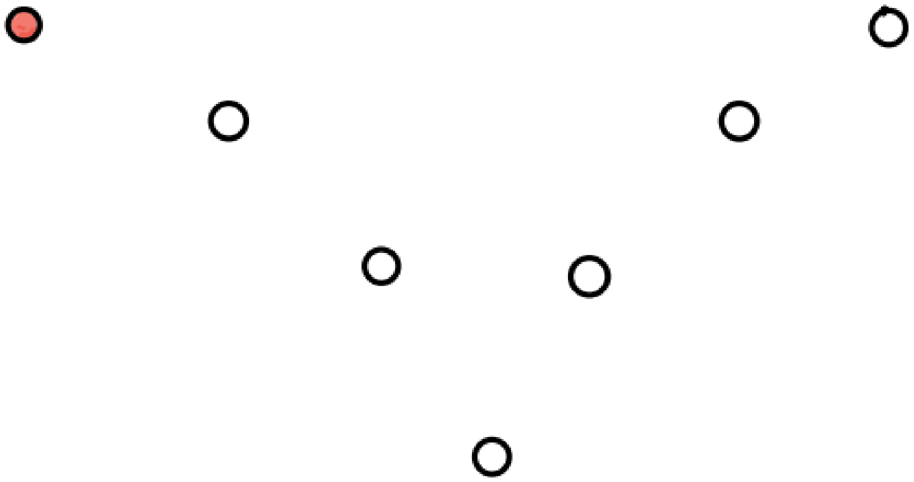
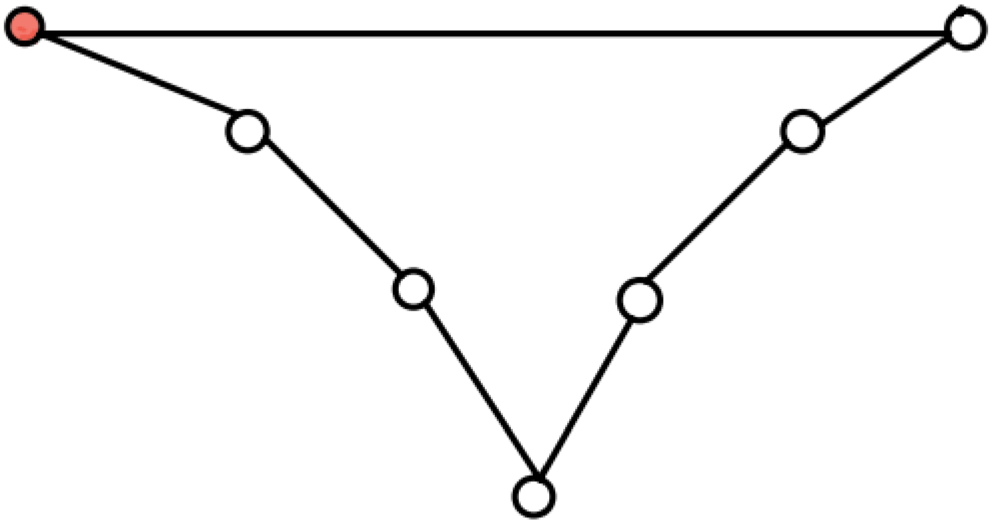
P1

Our goal is to find a greedy solution can;t be improved by inversion-based local search. Suppose processing one time of inversion-based local search is N_1. Therefore we need to find a inversion-based local search that inversing one time cannot improve the result but inversing multiple times can provide a better result. The ideology is small range local search may be blinded by the context.

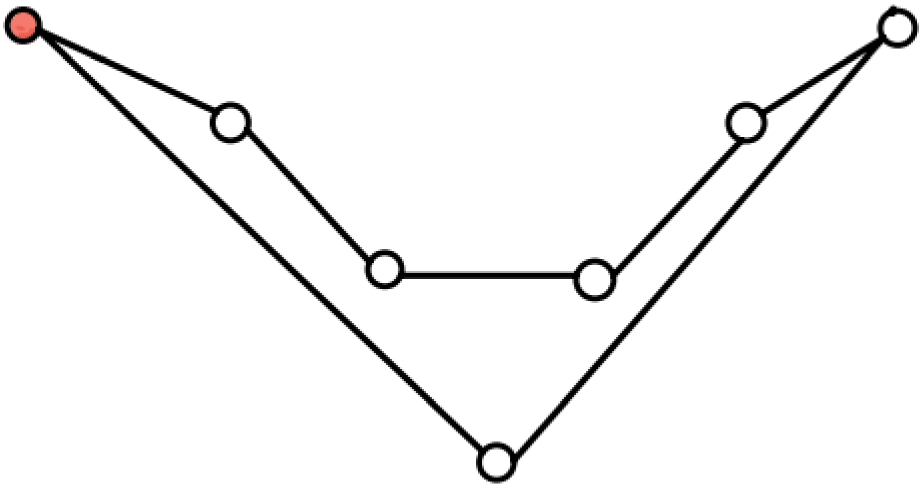
Red point is the initial point .



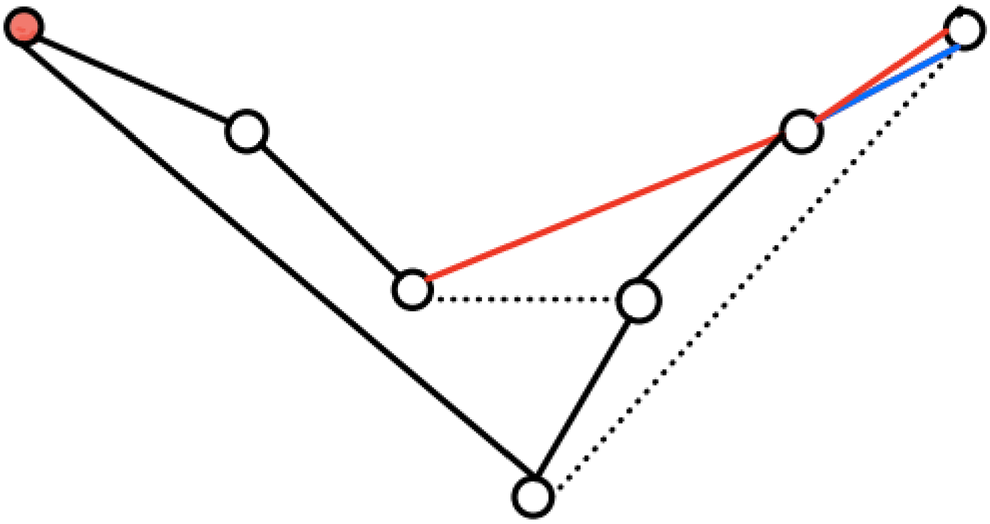
Optimal solution:

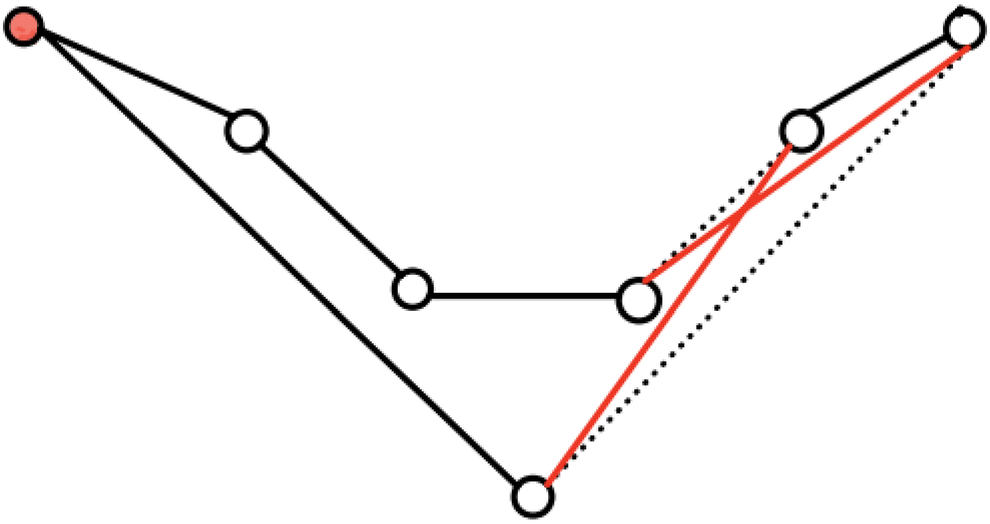


Greedy solution:



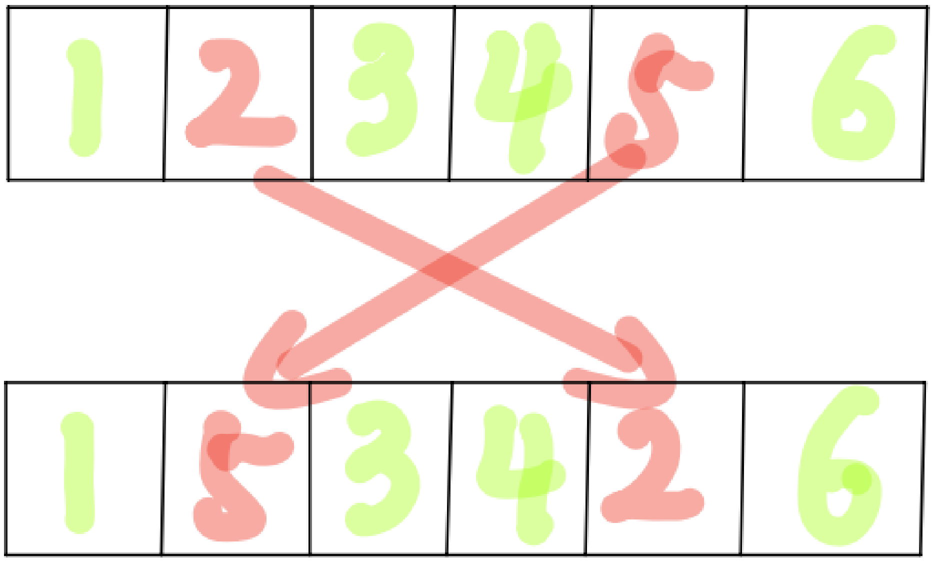
No better solution in one step range:



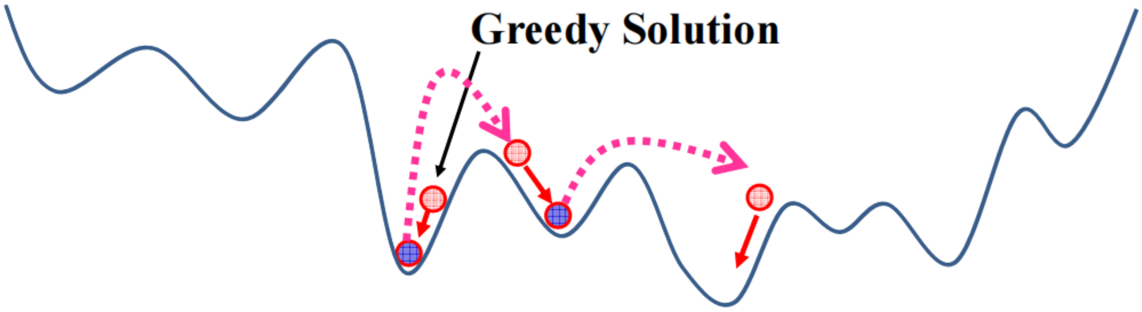


P2

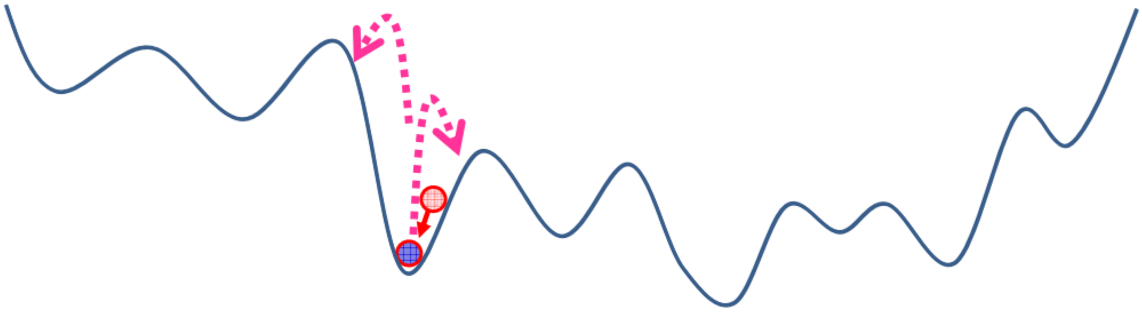
To create an initial solution for a new local research run, we can swap the positions of two points that were originally 2 or more apart, see the figure below.



A new solution form the final solution



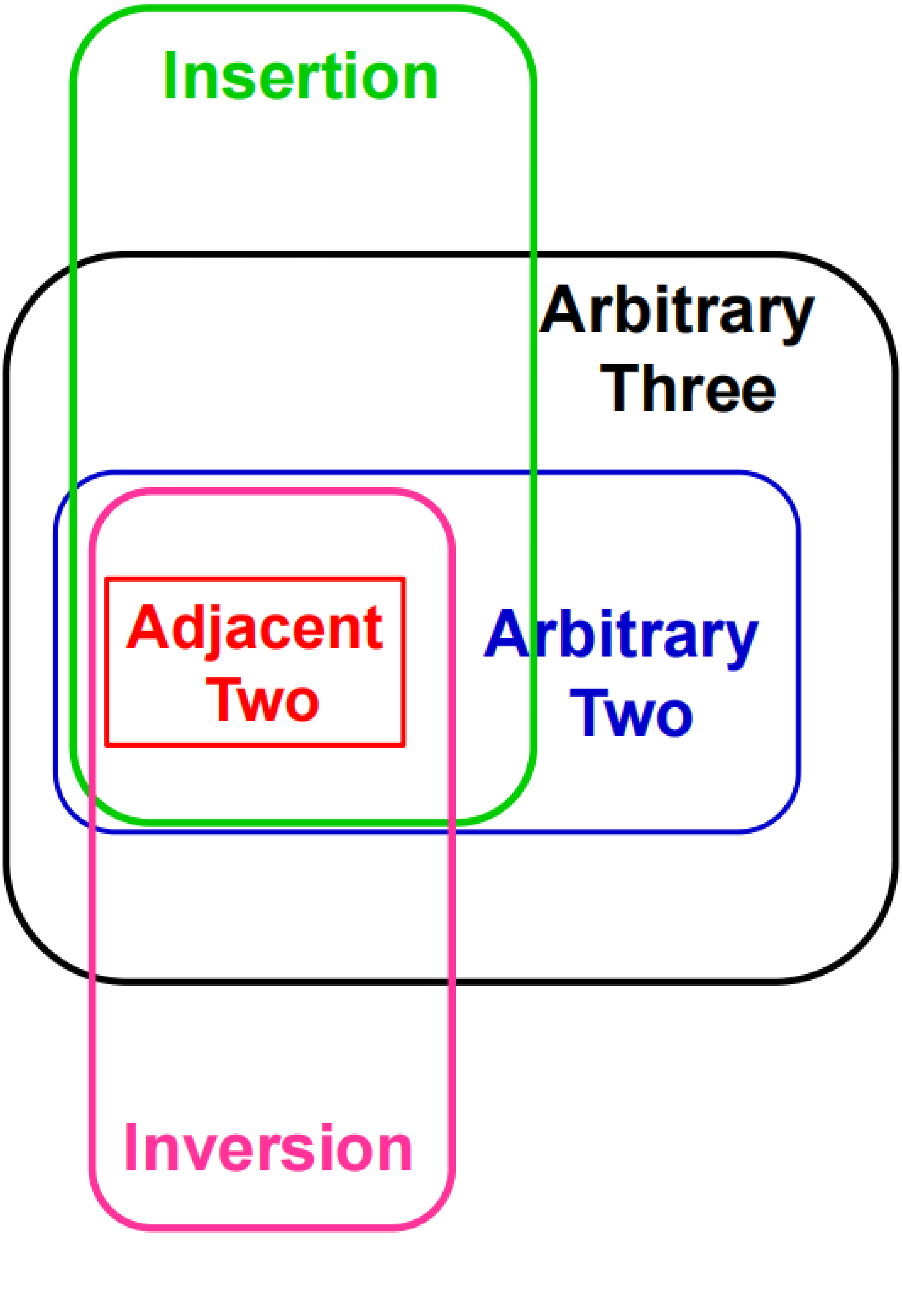
The following undesirable situation can happen.



The new initial solution can be far enough away from the last final solution since we need to exchange two edges for at least 5 times to restore the original state, i.e., the inversion-based distance between the new initial solution and the last final solution is at least 5. That's enough for us to jump out of the local optimal "trap"

P3

There are multilple neighborhoods we can choose. Adjacent Two-city Change (adj), Arbitrary Two-City Change (arb2), Insertion(ins),Inversion(inv), Arbltrary Three-City Change(arb3) are the common methods constructing neighborhood for TSP problem. We can arrange the neighborhoods by {adj, arb2, arb3, ins, inv}.


Insertion
Arbitrary Three
Adjacent Two
Arbitrary Two
Inversion