Q1

Assignment 1.

Rongrong Su.
20751807.

1.

(a) state: Represent the missionaries by M. and the cannibals by C. Let the boat be B. Each state can be represented by the item on each side
eg. ~~Side1 {MMCC}. Side2 {M.C.B}.~~ Side 1. {3M. 3C. b}. side 2 {0}

start state ~~Side1 {M.M.M. C.C.C. B}.~~ {3M. 3C. b}

~~goal~~ goal state: ① ~~Side1 {B}. Side2 {M.M.M.C.C.C}~~ {0M. 0C. 1.

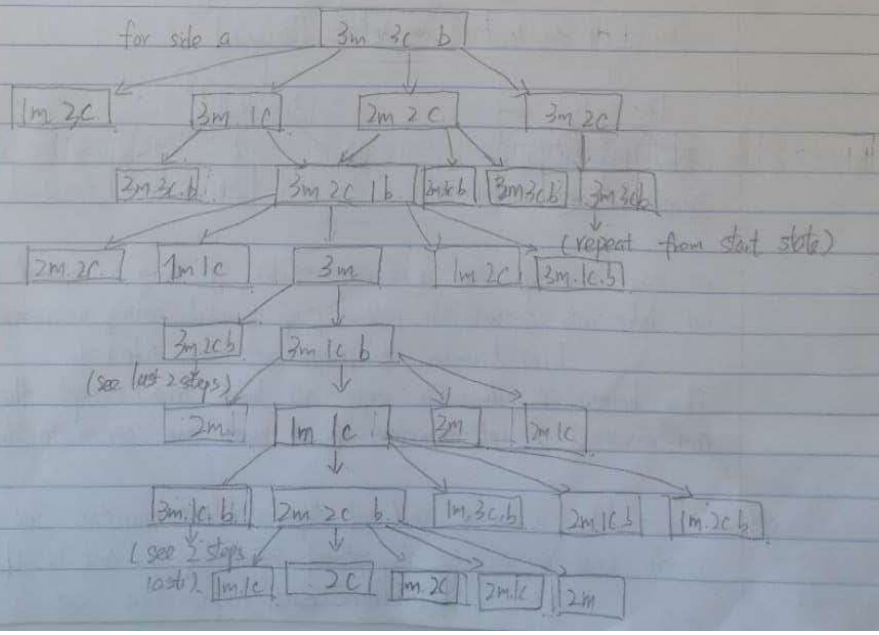
② ~~Side1 {0}. Side 2 {M.M.M.C.C.C.B}~~ {0M. 0C. b}. (Move)

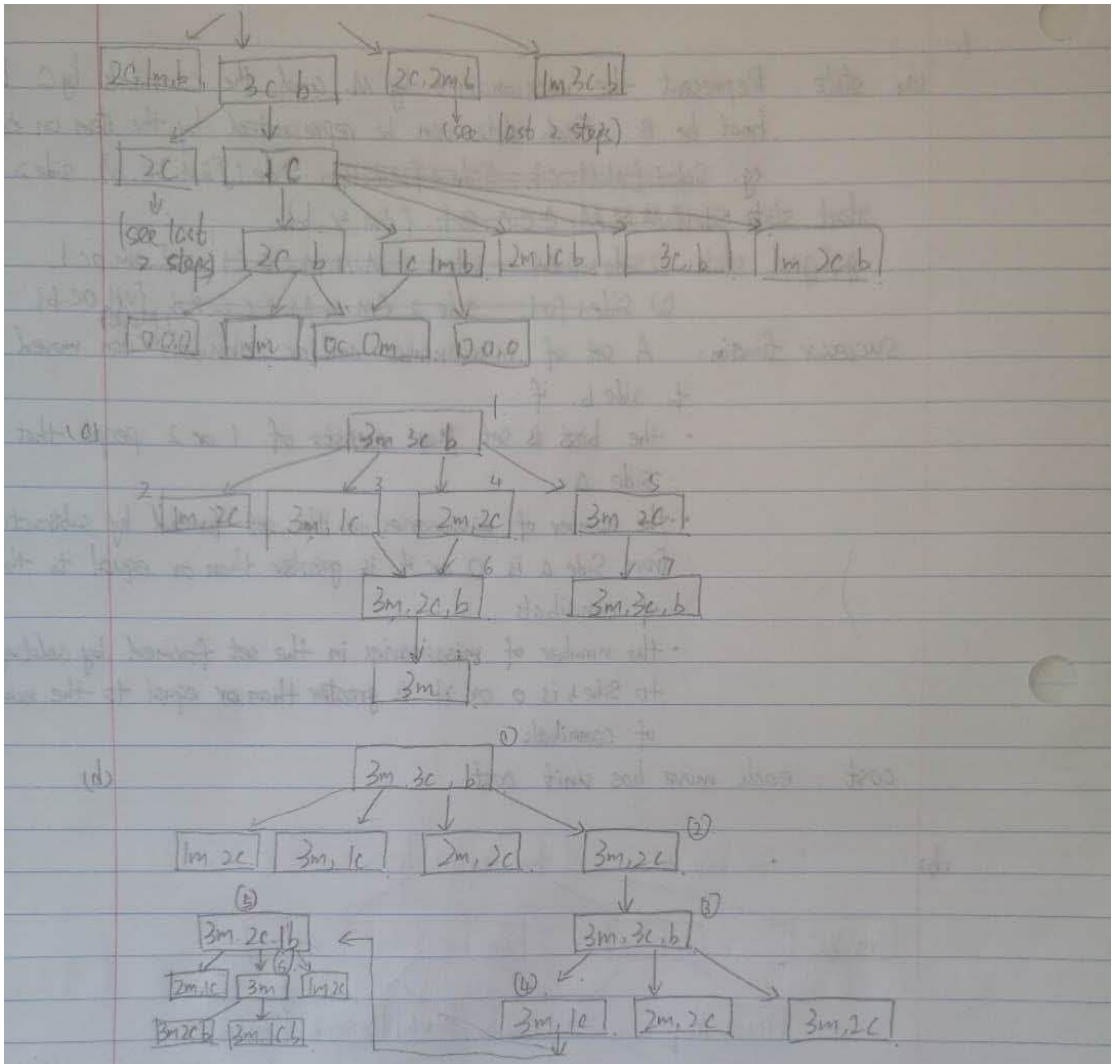
Successor function: A set of missionaries and/or cannibals can moved from side a to side b. if:

- the ~~boat~~ set Move consists of 1 or 2 people that are on side a.
- the number of missionaries in the set formed by subtracting Move from Side a is 0 or it is greater than or equal to the number of cannibals
- the number of missionaries in the set formed by adding Move to Side b is 0 or it is greater than or equal to the number of cannibals

cost: each move has unit cost.

(b)





(e) we can come up with a heristic h(x) by solving a relaxed problem, i.e. we do not take into account the possibility of cannibals eating missionaries

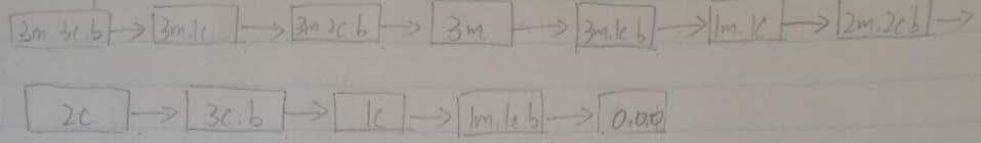
$h(x) = (\text{number of people on initial side}) - 1.$

This heristic is admissible, since all boat trip (except the last one) can result in a net transfer of at most one person to the destination side

(f) Since the tree is small and all steps cost are equivelant, we can use BFS search since it produce an optimal solution, The cost is 11.

And there exists loop in this case

the path is



Q2

Q2.

(a) state : sequence of cities visited
start state: A
goal state: every city is visited once and the path at last goes back to A. with the lowest total cost.
successor function: travel from one city to another.

(b) heuristic function: distance to the nearest unvisited city from the current city + estimated distance to travel all the unvisited cities (MST heuristic) + nearest distance from an unvisited city to the start city.

(c) As shown below.

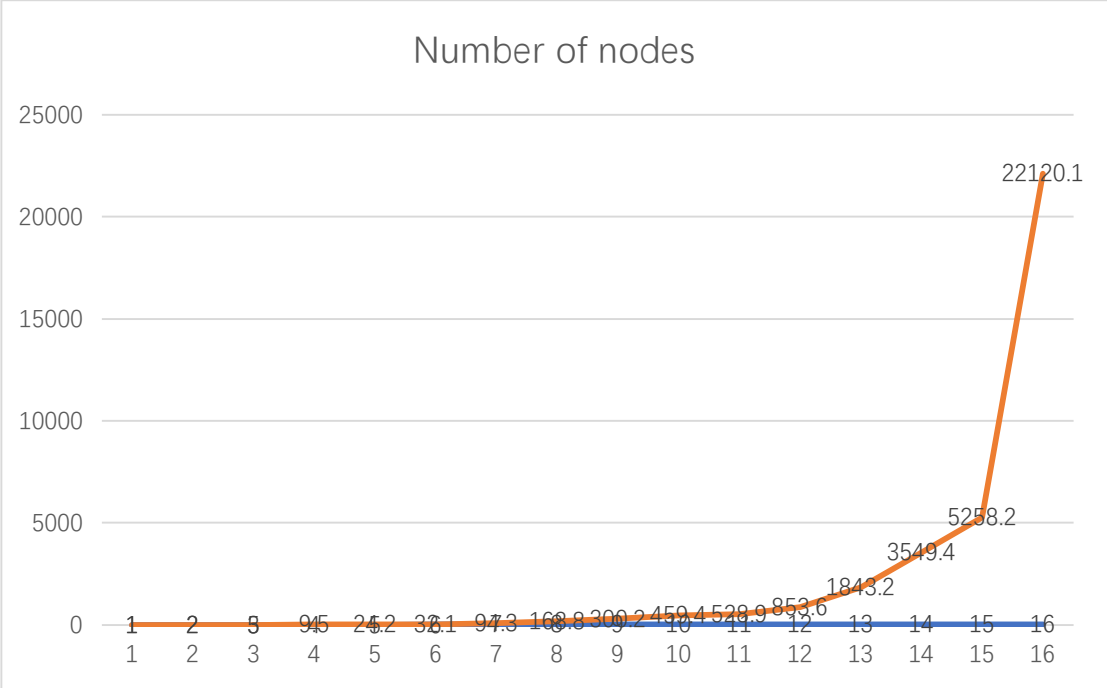
(d) As showed in picture. I fit the line with polynomial.
The function is $y = 0.1818x^6 - 8.704x^5 + 144.78x^4 + 545.9x^3 - 9337.4x + 11886$
when $x = 36$ the estimated number of estimated nodes is around 8800000

(e) As shown below

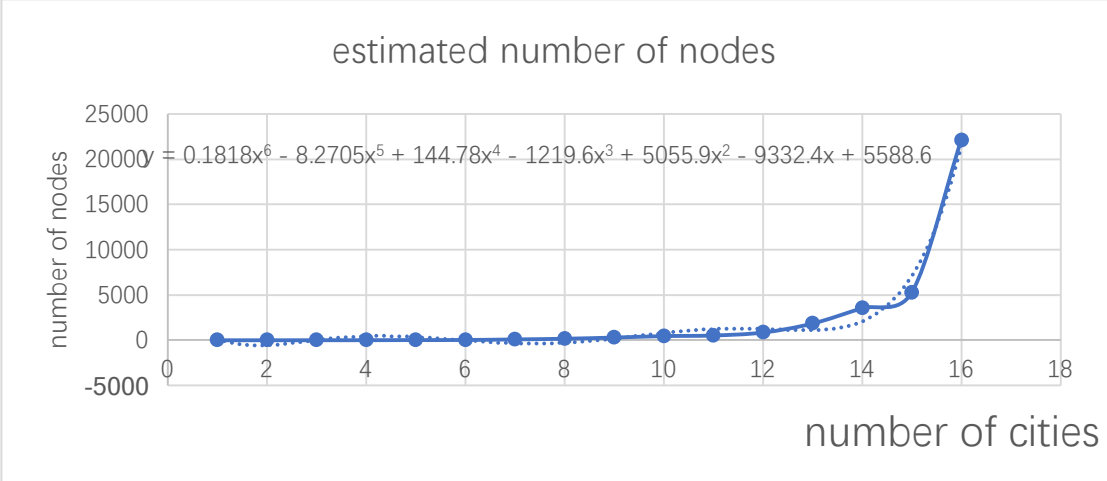
(f) As shown in picture. I fit the line with polynomial it fits well when n = 1 to 16
The function is $y = 1131.4x^6 - 6.51 \times 10^5 x^5 + 9.2546x^4 - 7E+06x^3 + 3E+x^2 - 6E+07x + 2E+07$
when $x = 36$ the estimated number will greater than 3E+11 and close to 10E+12 which is a huge number

(g) first MST heuristic is admissimable. we can solve 36-cities problem in 2.8×10^7 exploring nodes. MST algorithm in fact, provides a smart way to find the smallest path, using $AB + BC \geq AC$. Actually $h(n)$ is always lower than the cost of moving from n to the goal, then A^* is guaranteed to find a shortest path. The lower $h(n)$ is, the more nodes A^* expands, making it slower. (As we show in question(c)(e)) And it is very possible that with an improper h function (like h=0) we can not get the solution of a problem.

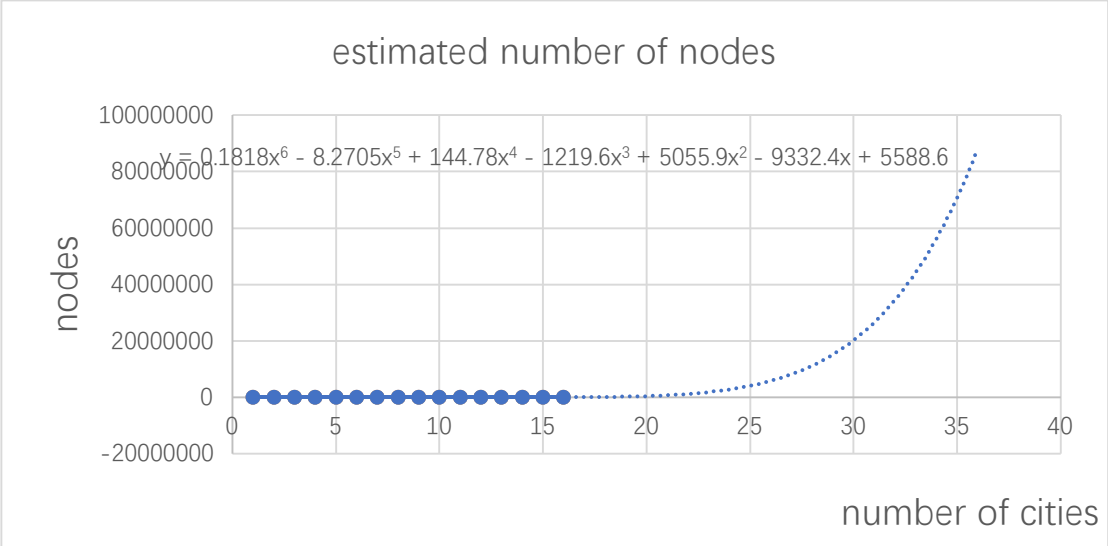
Number of nodes for TSP with h function



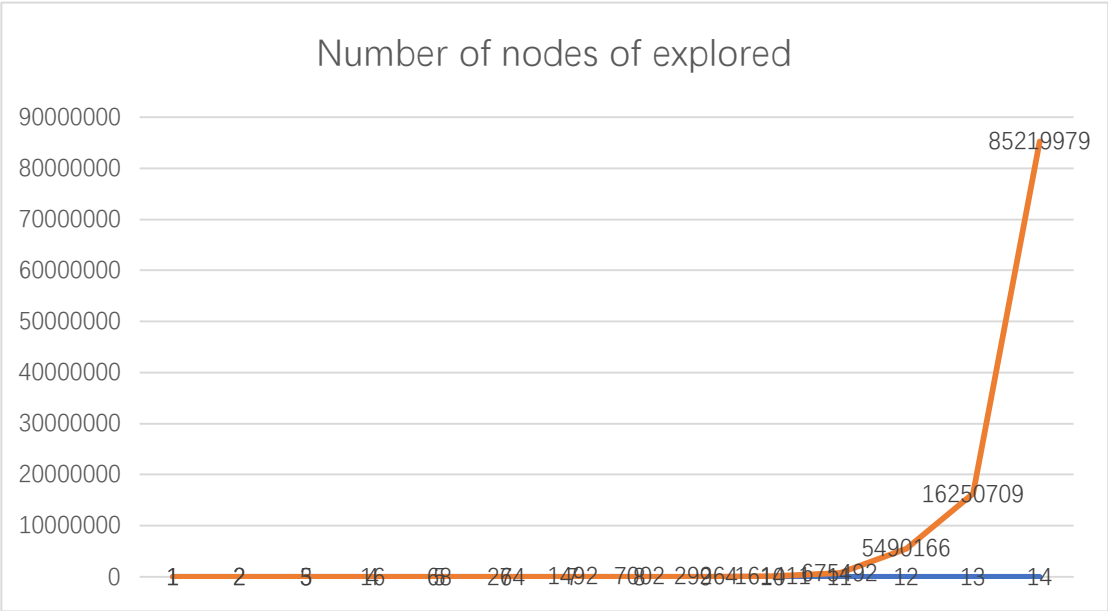
Fitted line and estimated function



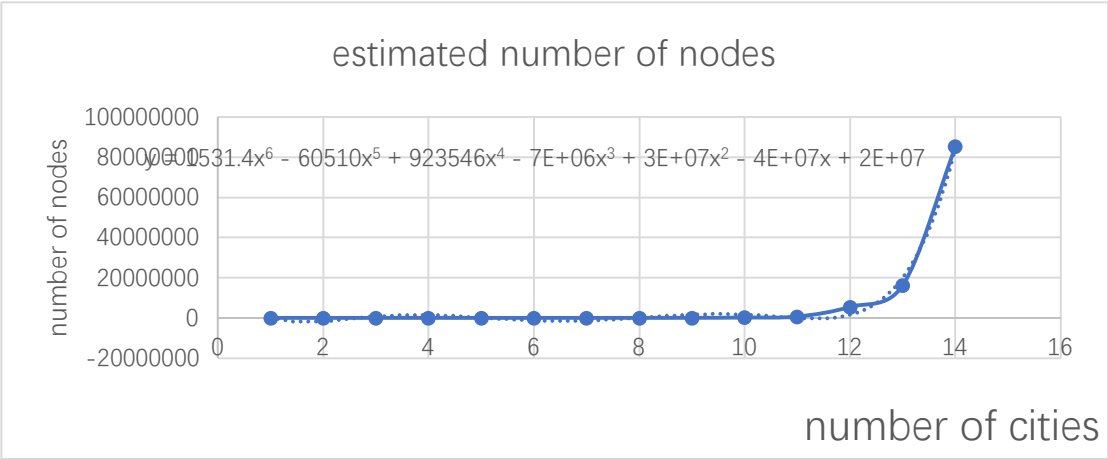
Estimates number when n=36



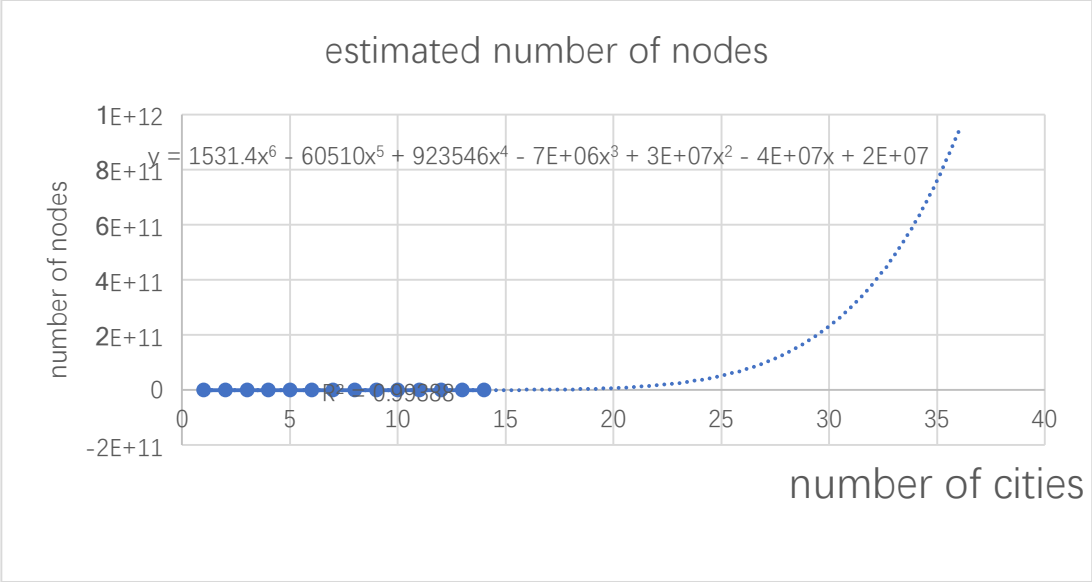
Number of nodes for TSP with h=0



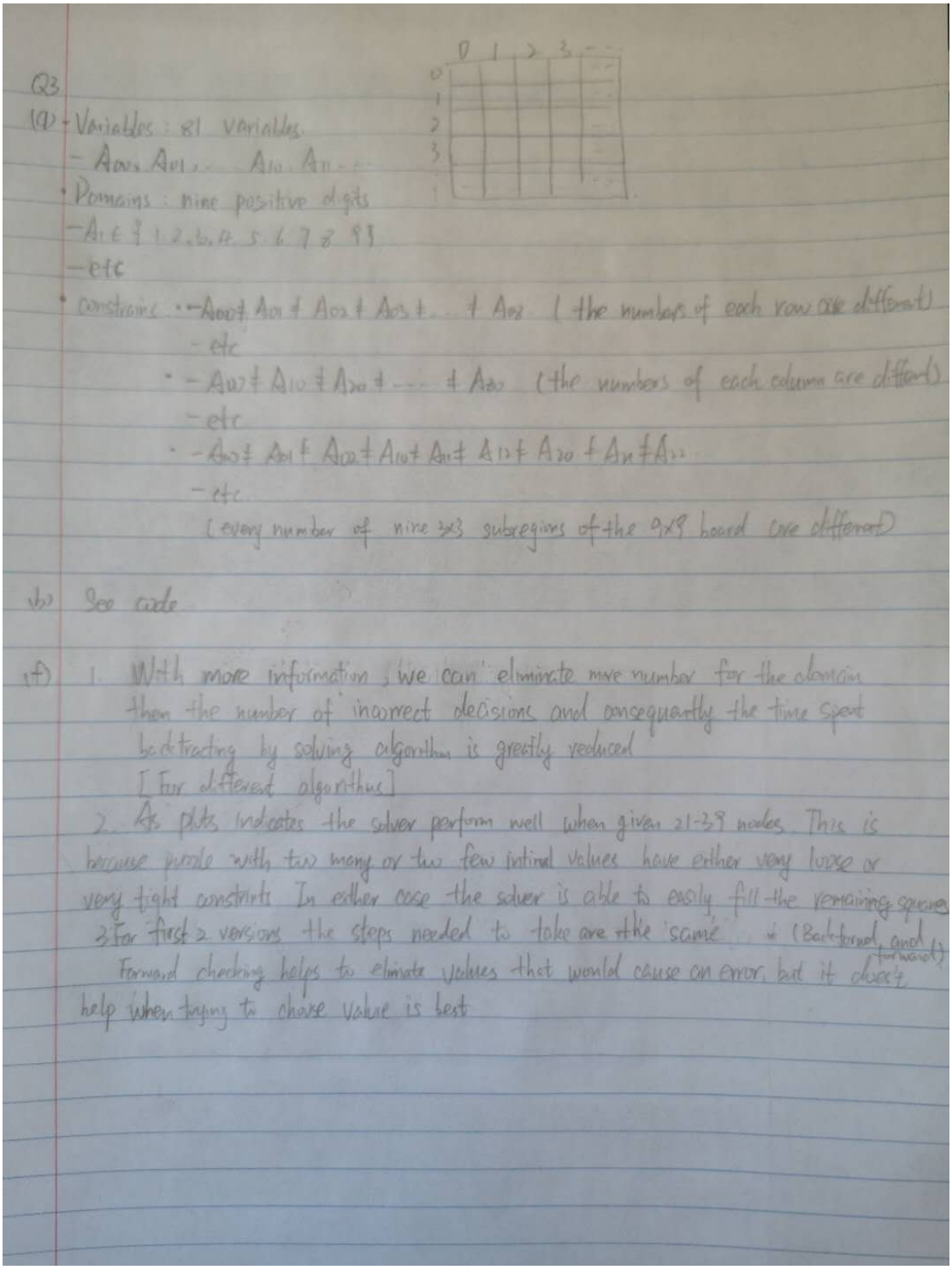
Fitted line and estimated function



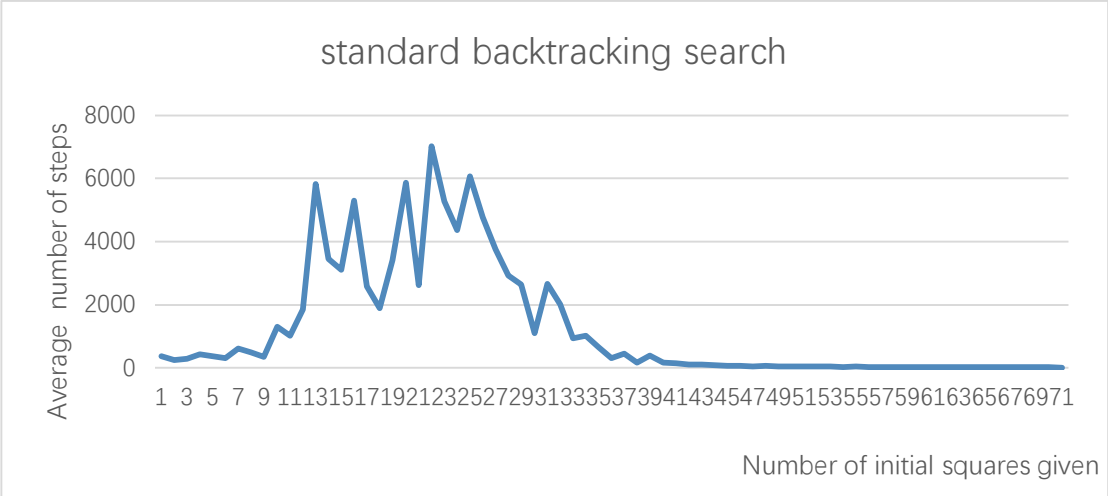
Estimates number when n=36



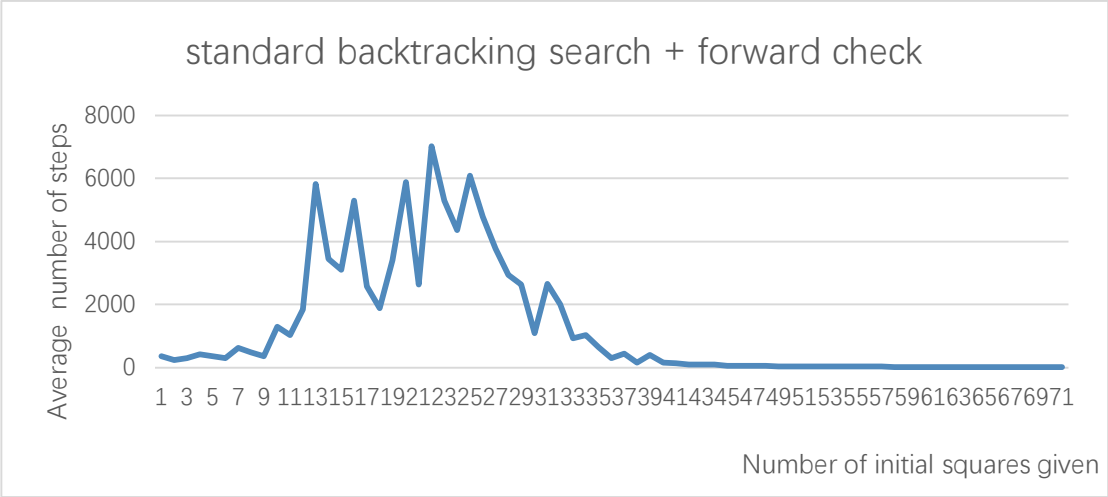
Q3


Q3
(a) - Variables : 81 variables.
- A_{00}, A_{01}, ... A_{10}, A_{11}, ...
• Domains : nine positive digits
- $A_{11} \in \{1, 2, 3, 4, 5, 6, 7, 8, 9\}$
- etc
• constrains : - $A_{00} \neq A_{01} \neq A_{02} \neq A_{03} \neq \ldots \neq A_{08}$ (the numbers of each row are different)
- etc
• - $A_{00} \neq A_{10} \neq A_{20} \neq \ldots \neq A_{80}$ (the numbers of each column are different)
- etc
• - $A_{00} \neq A_{01} \neq A_{02} \neq A_{10} \neq A_{11} \neq A_{12} \neq A_{20} \neq A_{21} \neq A_{22}$
- etc
(every number of nine 3x3 subregions of the 9x9 board are different)
(b) See code
(f) 1. With more information, we can eliminate more number for the domain then the number of incorrect decisions and consequently the time spent backtracting by solving algorithm is greatly reduced
[For different algorithms]
2. As plts indicates the solver perform well when given 21-38 nodes. This is because puzzle with too many or too few initial values have either very loose or very tight constraints. In either case the solver is able to easily fill the remaining squares
3 For first 2 versions the steps needed to take are the same. (Backtrack and forward)
Forward checking helps to eliminate values that would cause an error, but it doesn't help when trying to choose value is best

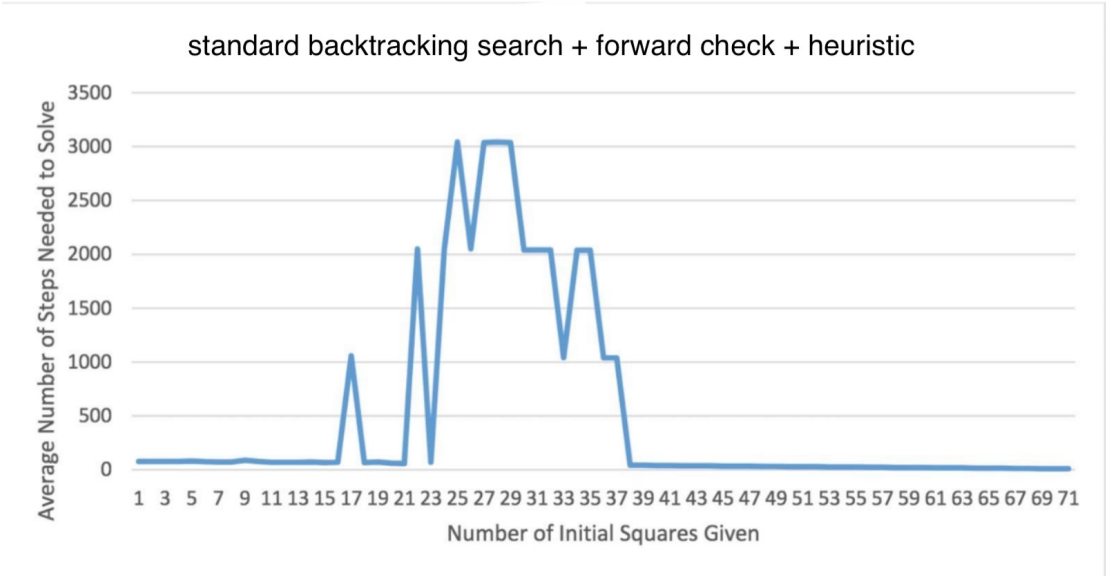
Number of assigned number of standard backtracking search.



Number of assigned number of standard backtracking search + forward check.



Number of assigned number of standard backtracking search + forward check + heuristic.


standard backtracking search + forward check + heuristic
Average Number of Steps Needed to Solve
3500
3000
2500
2000
1500
1000
500
0
1 3 5 7 9 11 13 15 17 19 21 23 25 27 29 31 33 35 37 39 41 43 45 47 49 51 53 55 57 59 61 63 65 67 69 71
Number of Initial Squares Given