

# 109學年度大學部專題競賽



國立清華大學資訊工程學系

## Learning to Route Communication - Limited Vehicle Fleet in Road Networks

徐逢禧, 趙柔茵

### 摘要

在這個大數據時代中，數據可以發展出許多有趣且實用的應用。然而，大量的數據收集往往非常耗時。為了解決這個問題，本專題以Reinforcement Learning為方法，於路網上分析資料特徵並選擇最佳路徑，讓數據收集效率最大化。藉由在模擬環境中進行訓練和測試，來達到比起近似演算法為依據更加有效率的路徑選擇。

### 題目介紹

此題目給定 $G = (V, E)$ ，有 $N$ 個agent起始在不同頂點。目標是每個agent各自找到一條路徑且它們的聯集覆蓋 $E$ ，並使所有路徑加總起來的和 (cost) 盡可能最小化。

期間agent可以根據地圖資訊判斷如何選路徑。agent要獲得關於地圖資訊的方式有兩種，第一種是親自經過該道路蒐集所需資料，第二種是透過與在溝通範圍內 (constraint) 的其他agent進行資訊交換。

### 研究方法與步驟

為了實作並證明出 RL在解決此問題上能勝過一般演算法。我們設計Deterministic algorithm，並以cost低於它為目標，訓練出DQN model，並探討RL能勝出的原因。

### Deterministic Algorithm

Agent永遠挑「被走過最少次」的edge走，如有多條edge走過次數相等便按照「固定順序」來挑選，直到所有的edge都被走過一次為止。

至於edge被走過幾次的資訊，則由agent之間互相溝通而知，有溝通範圍限制的演算法是「Algo\_WC」；沒有溝通範圍限制，也就是有上帝視角的演算法為「Algo」。

D  
Q  
N  
  
M  
O  
D  
E  
L

### Model introduction

總共有11層layer，其中1層input layer、9層hidden layer和1層output layer，除了output layer以外每一層都加上Relu；loss function使用MSE；optimizer使用Adam。

### Input example

假設agent A在 node 1，而node 1有3條edge分別連到node 2, node 3, node 4，而edge(1,2)、edge(1,3)、edge(1,4)的長度分別是(532, 166, 983)、此三條edge有連到的edge數量為(6, 2, 5)、被走過的次數分別是(0, 1, 2)。

Model input :  $[[532, 6, 0], [166, 2, 1], [983, 5, 2]]$ 。

### Output example

Model output :  $[[0.7752], [0.3106], [0.1665]]$ 。agent A 挑數值最高的 edge(1,2)走。

### Reward function

reward function的設計從4個方向著手。

(1) edge被走過的次數 (2) edge的長度 (3) 目前所花的時間 (4) 下一條edge的好壞程度。

透過以上4種方法，我們設計兩個不同面向的reward function，因此最終有兩個不同的DQN model。「Model\_dist」，依照(1)、(2)、(3)的規則做設計；「Model\_newedgecon」，依照(1)、(4)的規則做設計。

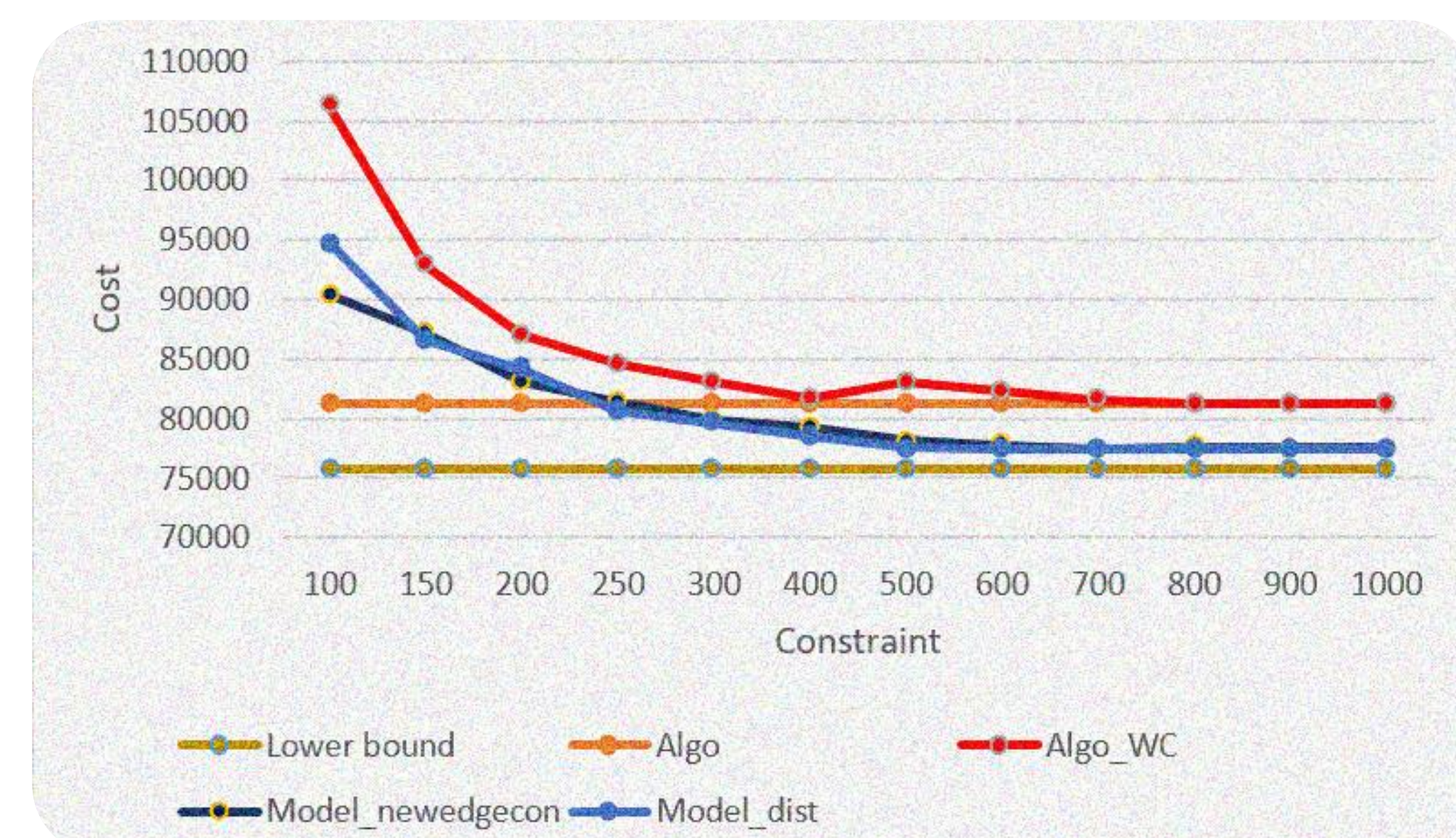
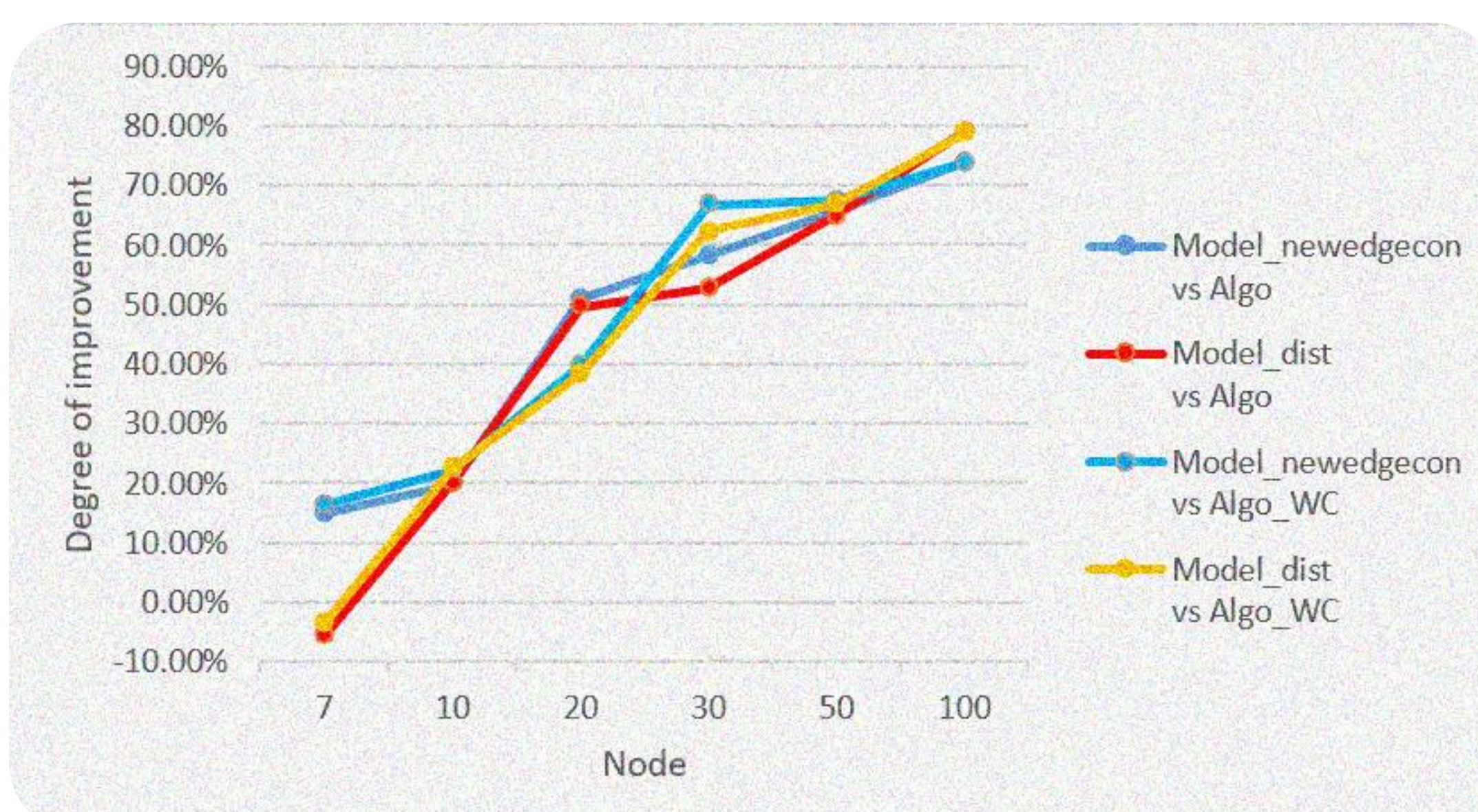
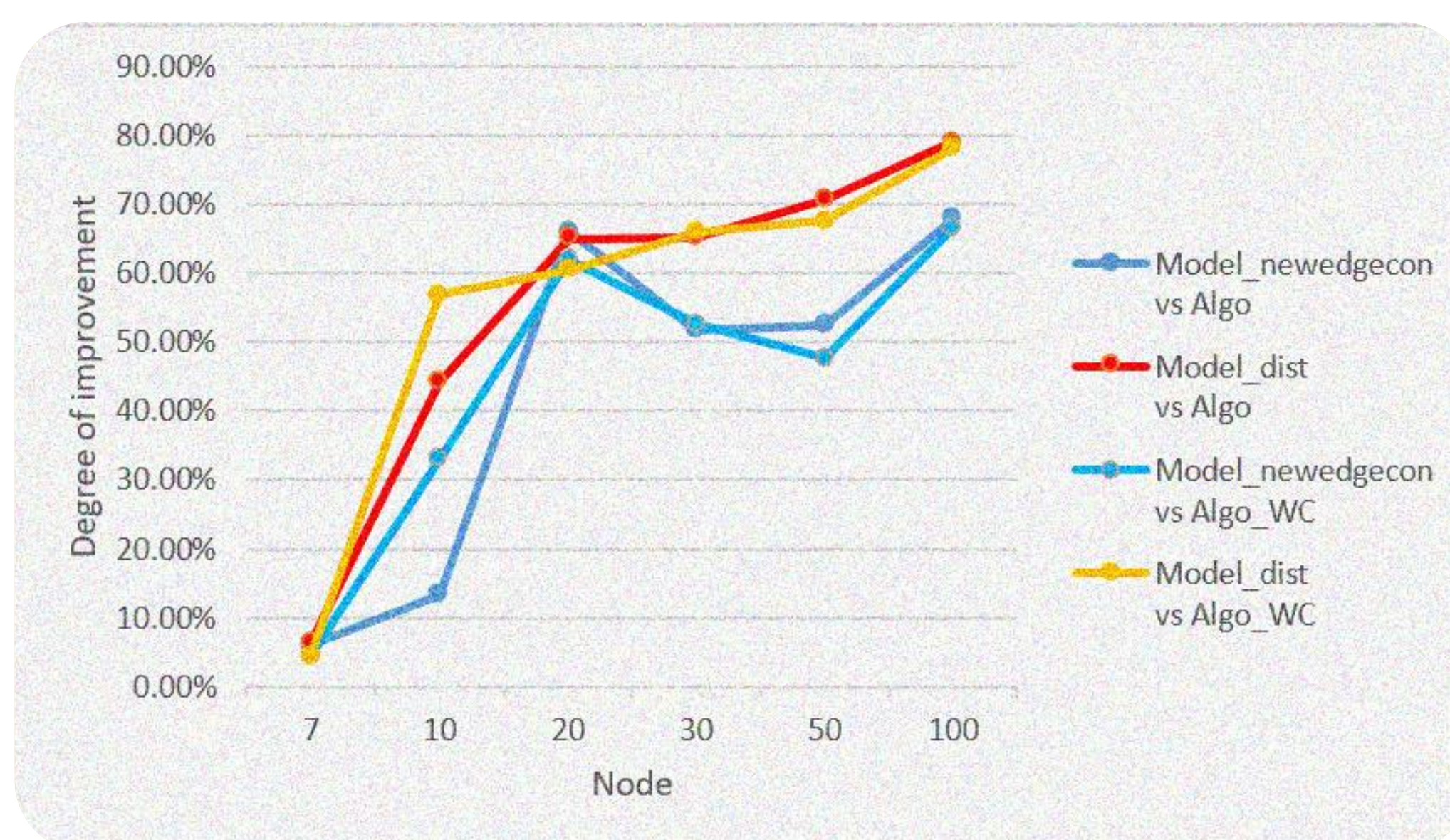
### 效能評估與成果

以下將Algo、Algo\_WC、Model\_dist、Model\_newedgecon，與 Lower bound 進行比較。(Lower bound：走完地圖所需最低的cost，算法是所有edge的長度加總，除以agent的數量。)

Map information						
	Map01	Map02	Map03	Map04	Map05	Map06
Number of nodes	7	10	20	30	50	100
Number of edges	18	42	187	432	1222	4947
Constraint	500					
Cost						
Lower bound	3220.8	7270.8	33046.9	75349.9	215751.5	866940.4
Algo	4084.2	8530.2	37035	81414	228357.9	898777.8
Algo_WC	4069.8	8899.2	36574.2	81538.2	227181.6	897509.7
Model_newedgecon	4032	8361.9	34393.5	78280.2	221739.3	877149.9
Model_dist	4031.1	7972.2	34441.2	77458.5	219452.4	873639.9

Map information						
	Map01	Map02	Map03	Map04	Map05	Map06
Number of nodes	7	10	20	30	50	100
Number of edges	10	22	95	217	612	2475
Constraint	500					
Cost						
Lower bound	1659.2	3888	16222.9	37200.8	108911.8	434736.3
Algo	2498.4	5220	19533.6	41709.6	117707.4	460876.5
Algo_WC	2514.6	5261.4	18927	42847.2	118215.9	460686.6
Model_newedgecon	2374.2	4956.3	17849.7	39081.6	111931.2	441531.9
Model_dist	2545.2	4951.8	17894.7	39329.1	111989.7	440228.7

Map information						
	Map01	Map02	Map03	Map04	Map05	Map06
Number of nodes	30					
Number of edges	432					
Constraint	100	150	200	250	300	400
Map information						
	Map07	Map08	Map09	Map10	Map11	Map12
Number of nodes	30					
Number of edges	432					
Constraint	500	600	700	800	900	1000



### 結論

只要具有足夠的溝通距離，我們的兩個DQN model都可以在不同類型的地圖上勝過演算法，且省下的cost甚至能接近80%！