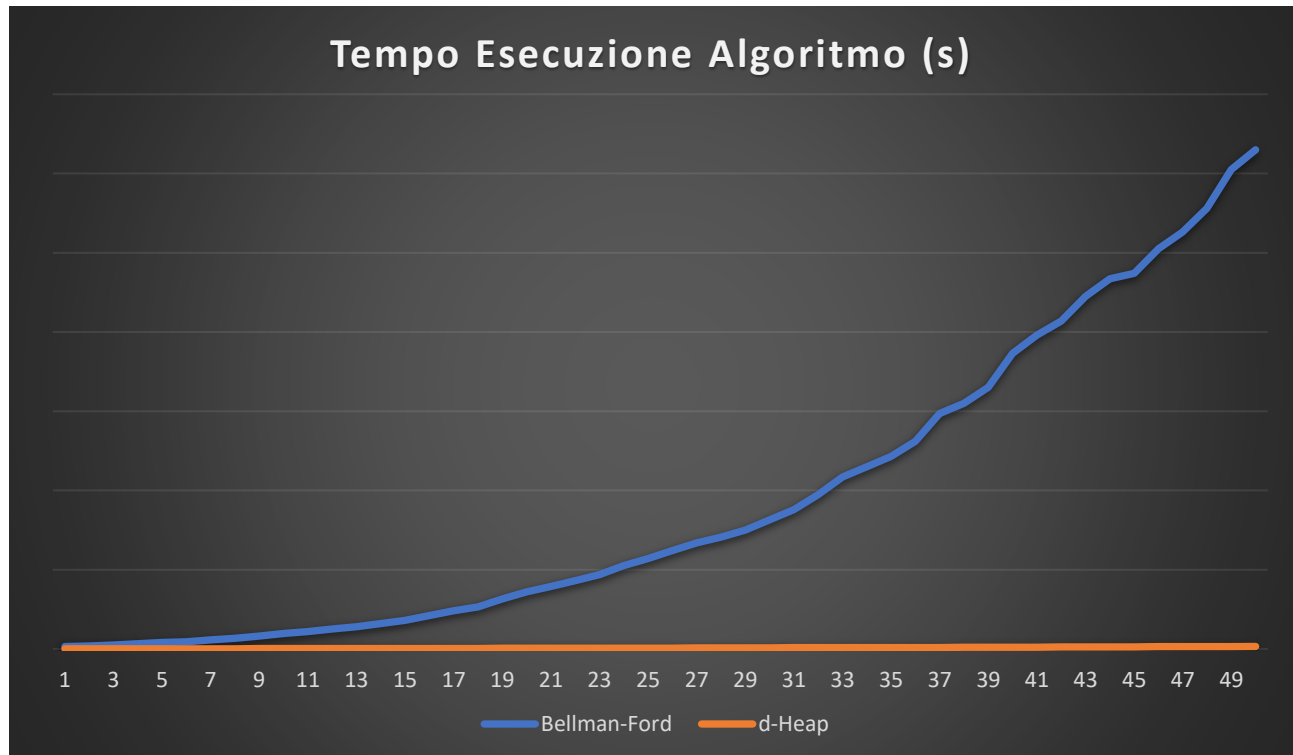


[xxxx] Engineering Of Algorithms Project

Oxxxxxx – Valerio Pinsone

Per la rappresentazione del grafo è stata utilizzata una matrice di adiacenza per garantire tempo di accesso costante ad ogni cella ed una rappresentazione intuitiva.

Tempo di esecuzione dell'algoritmo al variare lineare dell'input (step di 10):



Come si nota dal grafico, l'algoritmo che non utilizza l'ordinamento topologico tramite coda di priorità ha un drastico calo di efficienza all'aumentare dell'input. La complessità di Bellman-Ford è infatti $O(|V| |E|)$.

Non sono state riscontrate differenze significative nei tempi di esecuzione dell'algoritmo al variare di d utilizzando d-Heap come coda di priorità

Tabella con alcuni risultati sperimentali:

Numero Vertici Grafo	Tempo Exec (s): Bellman-Ford	Tempo Exec (s): d-Heap
130	0.3	0.016
140	0.37	0.016
150	0.5	0.016
160	0.67	0.02
170	0.83	0.025
180	0.9	0.025
190	1.13	0.026
200	1.36	0.026
210	1.62	0.031

220	1.93	0.031
230	2.17	0.031
240	2.5	0.037
250	2.8	0.037
260	3.2	0.046
270	3.6	0.047
280	4.2	0.05
290	4.8	0.06
300	5.3	0.062
310	6.3	0.07
320	7.2	0.07
330	7.9	0.078
340	8.6	0.08
350	9.4	0.084
360	10.5	0.09
370	11.4	0.1
380	12.4	0.1
390	13.4	0.11
400	14.1	0.12
410	15	0.14
420	16.3	0.14
430	17.6	0.15
440	19.5	0.16
450	21.7	0.16
460	23	0.17
470	24.3	0.175
480	26.2	0.18
490	29.7	0.188
500	31	0.19
510	33	0.21
520	37.3	0.22
530	39.6	0.23
540	41.4	0.25
550	44.5	0.26
560	46.7	0.265
570	47.4	0.27
580	50.5	0.28
590	52.6	0.285
600	55.6	0.29
610	60.5	0.3
620	63	0.31

L'ordinamento topologico (e quindi la sua variante con coda di priorità) permette all'algoritmo di esplorare i nodi nell'ordine corretto garantendo di effettuare le operazioni di rilassamento coerentemente alla gerarchia dei nodi. Questo si traduce in una maggiore efficienza dell'algoritmo e quindi minor tempo necessario alla risoluzione del problema.