Bavo Goosens r0297884 2e bach inf

Verslag Practicum 4

Gegevensstructuren en algoritmen

Inhoud

Gegevens:

N = 1000:
Tour distance = 26297,03988329931
Nearest insertion: 0,021 s
Tour distance = 15658,676372245684
Smallest increase: 0,039 s
Tour distance = 18985,53796154957
MST insertion: 1,438 s

N = 2000:
Tour distance = 37648,31473461854
Nearest insertion: 0,034 s
Tour distance = 22231,517761867846
Smallest increase: 0,086 s
Tour distance = 27018,56682419262
MST insertion: 11,007 s

N = 4000:
Tour distance = 53453,748307458874
Nearest insertion: 0,087 s
Tour distance = 31072,686570495483
Smallest increase: 0,281 s
Tour distance = 37864,91377536393
MST insertion: 89,84 s

N = 8000:
Tour distance = 75352,82641597513
Nearest insertion: 0,299 s
Tour distance = 43920,386101119286
Smallest increase: 1,066 s

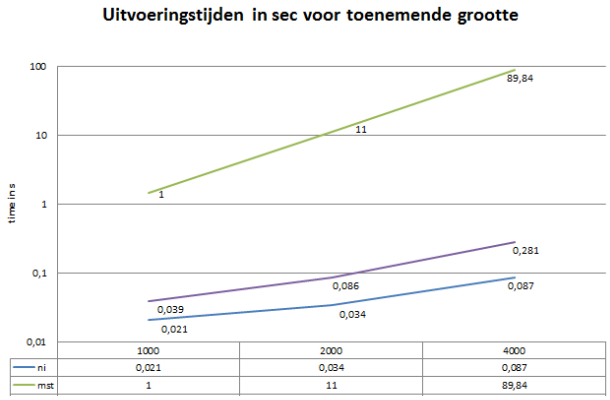
N = 16000:
Tour distance = 105056,98330292058
Nearest insertion: 1,122 seconds
Tour distance = 61874,15134395292
Smallest increase: 4,205 s

N = 32000:
Tour distance = 148900,37834350808
Nearest insertion: 4,311 s
Tour distance = 87667,62448333761
Smallest increase: 16,817 s

N = 64000:
Tour distance = 210151,4443922731
Nearest insertion: 18,755 s
Tour distance = 123954,30811017669
Smallest increase: 73,388 s

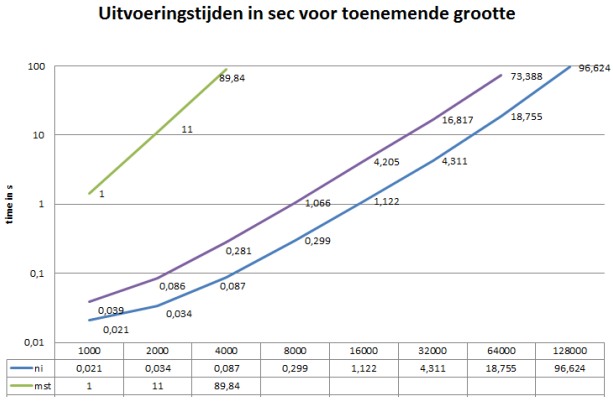
N = 128000:
Tour distance = 298323,77445611893
Nearest insertion: 96,624 s
Tour distance = 175060,13402392497
Smallest increase: 375,632 s

Grafieken:



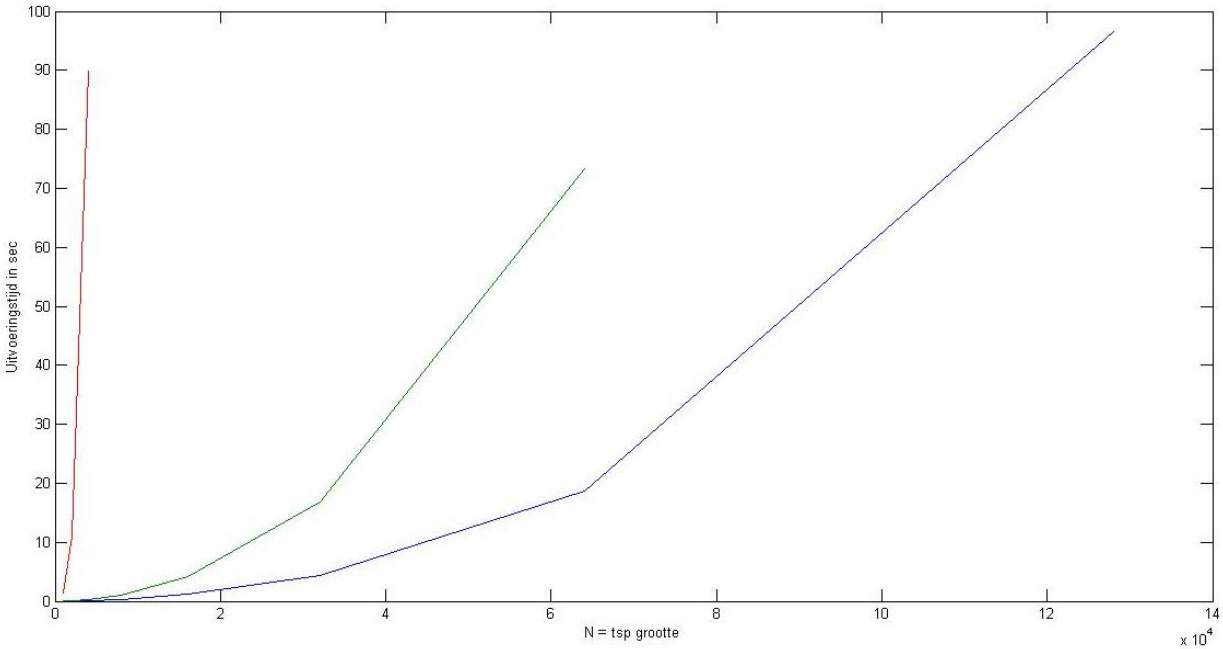
	1000	2000	4000
ni	0,021	0,034	0,087
mst	1	11	89,84

Grafiek 1



	1000	2000	4000	8000	16000	32000	64000	128000
ni	0,021	0,034	0,087	0,299	1,122	4,311	18,755	96,624
mst	1	11	89,84					

Grafiek 2



Grafiek 3

Bespreking:

Allereerst ga ik het even over de afstanden van de tours hebben voor de heuristieken. Wat meteen opvalt is dat smallest increase insertion het beste resultaat oplevert. Om dit resultaat te berekenen moet wel telkens de totale afstand van het pad berekend worden. Om dit alles efficiënt te laten verlopen maak ik steeds gebruik van de totale afstand van de vorige tour. Op deze manier kan met simpele basisoperatoren en minder rekenintensieve methodes deze nieuwe afstand efficiënt berekend worden. De MST heuristiek levert ook een goed resultaat maar boet dan weer in op vlak van efficiëntie. Voor de dichtstbijzijnde buur heuristiek en de smallest increase heuristiek is de asymptotische complexiteit $O(N^2)$. Dit valt duidelijk in de grafieken terug te zien. Het verloop van de functies is telkens kwadratisch en in een logaritmische plot zien we dat de twee heuristieken dicht bij elkaar aansluiten op vlak van uitvoeringstijd. Dit valt te verklaren door de manier waarop deze tours worden opgebouwd: er moeten telkens n punten worden toegevoegd en om de plaats te bepalen zijn er N vergelijkingen en berekeningen nodig. Het verschil is te wijten aan constanten die door extra berekeningen en vergelijkingen veroorzaakt worden. De uitvoeringstijd van de MST heuristiek vertoont echter hetzelfde gedrag. Hetgeen we niet zouden verwachten. Ik maak gebruik van een lazy prim algoritme en ga dus telkens op zoek naar de volgende kortste edge om aan mijn boom toe te voegen. Normaal gezien zou dit een complexiteit van O(E log E) opleveren om de MST te construeren. In ons geval is de complexiteit $O(N^2)$. Dit ligt aan de manier waarop de graaf wordt opgebouwd. Deze verwachte complexiteit is gevonden in graaf waarin alle edges al gekend zijn. Dit is bij ons niet het geval en deze moeten telkens berekend worden. Het aantal edges kan ook toenemen tot N^2 en dit gaat hier ook gebeuren, zei het niet exact. Dit drijft de complexiteit op tot $O(N^2)$.

Er kan dus besloten worden dat de SI heuristiek de beste oplossing is als benadering voor het TSP probleem. MST is hier te rekenintensief eenmaal de grootte van het probleem te hoog oploopt en de NN heuristiek is nogal inaccuraat. Moest echter snelle uitvoering aan de orde zijn zal de NN als heuristiek de aangeraden keuze zijn.