

Verbetering:

Tijdens de nabespreking is duidelijk geworden dat de twee-fasen aanpak zoals in het originele programma beschreven, niet optimaal is. Beter is om het algoritme sneller kleinere verbeteringen per iteratie te laten maken. Daartoe hebben we in deze verbetering een volledige tabu-search toegepast. Een iteratie zal nu alle adressen in een oplossing proberen in te voegen op elke andere mogelijke positie. De verplaatsing van een adres naar een locatie wat de laagste kosten teweeg brengt, zal de nieuwe oplossing worden. De tabu-lijsten worden nog steeds bijgehouden zoals in het originele programma.

Nog een kleine opmerking; omdat het plaatsen van adressen in een route onder een agent anders dan zijn default agent niet mogelijk is, hebben we weer ervoor gekozen parallel te zoeken bij alle agenten. Één iteratie voert dus eigenlijk één iteratie uit voor elke agent die bestaat. Dus als er zeven agents in het probleem worden gebruikt, komt één parallelle iteratie overeen met zeven normale iteraties (uitgevoerd op elk agent-deelprobleem).

Een andere handige wijziging is een zgn. tracker-functie welk een gegeven aantal keren een parallelle iteratie zal aanroepen. De oplossing met de tot dan toe laagste kosten zal worden onthouden, evenals de geschiedenis van tegengekomen kostenwaarden.

Ook bleek tijdens de nabespreking dat onze kostenfunctie wat te pessimistisch was. We hebben inderdaad een paar fouten ontdekt en verbeterd, wat resulteerde in lagere kosten. Na veel gezwoeg in een achteraf toch onhandig Haskell (erg lastig debuggen) hebben we een werkend experiment kunnen uitvoeren:

54 min, 26 iteraties, History:

[474837,427387,372696,336421,304661,286842,267973,256232,242722,232093,224109,218693,210538,203961,198411,197546,196547,194805,175748,174406,174058,173472,172421,172391,172391,172391]

oplossing in: output.xml

We zien dat hij blijft steken op kosten 172391. We kwamen er op het laatste moment achter dat routes standaard op de starttijd van de agents beginnen, ook als de eerste winkel in de route dan nog niet open is. Het zou kunnen dat dat onnodige kosten veroorzaakt. Helaas hadden we niet genoeg tijd dit nog uit te proberen.