Talk 28.08.2019

Siluette Coefficient:

Found it a little bit strange that it is always better to have a higher k with this method no matter how well clustered everything is it will always be better to remove one node from a cluster than to keep it in a cluster (i.e. the best way is to

Talk 21.08.2019

Clustering operator:

A bit unsure if my current method.

Clustering:

I am using a hierarchical clustering method to cluster the orders pickup/deliveries and stopping when I reach k equal to amount of vehicles. Its a bit of a heavy clustering method but it doesnt matter so much since I am only using it once per Instance.

Find a measure for how good a cluster is. Siluette measure. Then try different sizes of k and keep the best k.

Either use clusters to assign orders as an initial solution or as seperate operator.

Then for the operator. I have several options for removal/insertion. I am listed them up here:

removal: either use one of the removal heuristics I already have, so similar/expensive/random. Or I can remove a random order and then remove orders who are transported from/to the same cluster, until I reach some limit or until I have no more orders in the same cluster. (alternatively I can remove several clusters and save them together somehow)

check if there are some orders that are minorities.

If in one vehicle, we have one.

For one vehicle. Are they from to similar clusters. Make a similarity measurement. Try to insert where similarity is the highest with orders on vehicle.

Check also the instance no 35.

Make the excel table better based on the hemmati paper so that we have the information that we need. We need the initial value the average and best.

Check online for table latex converter.

Check the solution of the large instance and see if there are some strange pickups/deliveries and see if you can use this.

Insertion: find an empty vehicle and try to insert orders together into the vehicle in the best possible way.

Another alternative is to search for orders that are picked up/delivered in the same cluster and try to insert the order there.

A third alternative is ofcourse to combine the removal of similar cluster orders to just inserting with regret-k or

Uncertainty: Dont get a good feeling about any of the different concepts and I am unsure how much better it will be at finding solutions. Feel like it is also in a way quite similar to Shaw. Can maybe use it as a wild operator and then remove quite alot of orders and reinsert them again based on the clusters? A bit lost with the concept so maybe you have some good input.

Talk 15.08.2019

Remove similar:

1. Regarding the time window overlapping parameter: I ended up also subtracting the time when there is overlapping «no timewindow» from the dividend aswell. This makes it possible to reach a value of 1 for exact equal timewindows. Example:

two nearly identical time windows:

10 – 20 30 – 40 50 – 60

11 – 21 31 – 41 49 – 60

gave before:

(9 + 9 + 10) / (50) = approx 3/5

which is not very similar in regards to 1

however if you subtract the common «no timewindow» it gives:

(9+9+10) / 50 – 8 – 9 = approx 1

which is more accurate.

2. It does however produce a very ugly formula (see report) for the paper. Any tips on making this a little easier to read?

3. Also the Weight similarity I made now into |Q\_i – Q\_j| /max (Q). I thought about making the dividend max(Q\_i,Q\_j) since two orders with weight 10 and 20 is kindof 50% equal, however I found it made more sense that an order with 10 and 20 is very similar when the max weight possible is 12000. so i divide by the max Q overall. Make sense?

4. Discuss results after adjustements from last week.

Talk 05.08.2019

Greedy/regret-K operator Questions:

Removal of worst orders: As I now implemented the recalculation for each iteration for insert greedy insert, I here also recalculate now the worst cost for each iteration, I tried to just update the costs for the effected orders each iteration however I could not find a very effective way to do this. I currently calculate the new cost and replace it if it is better however I have to go through the list of orders each time. Seems unefficient since a HashMap for example can easily update the value based on the key. However a HashMap is not sorted. Do you have any suggestion?

Insertion greedy: I am here finding the min cost position to insert each order. Then I start with the cheapest one(or the one with the highest k) and insert this into its cheapest position. What do I do if I want to insert no. 2 in the same route? This obviously changes completely where I want to insert no.2 maybe a completely different route is now the cheapest position for no. 2 should I now check again for no. 2 which position is the cheapest and then insert it? That could then also change the order of things where no.2 might become no.4 in the original order. Should I then change the order and insert no.2 later? This was for me a bit unclear so I currently decided to if I have selected the same route I will check again which position is the best and insert here instead. That leads to a bit more runningtime but it should not happen too often, possibly only in the beginning when no routes have been inserted.

Change this operator to always calculate the cost in each iteration.

Removal of similar orders. Why do shaw say that two orders are similar when they have similar weight/quantity? ie. The difference between the weight of the orders multiplied by a constant? If two orders are large and only fit on a single vehicle i would think that we dont want to select them for removal at the same time. That they are of similar weight is rather negative in my opinion for high weights, however when their combined weight is small it would be positive ofcourse. Maybe I could have some parameter that gives a high similarity when both have small weight and very unscimilar when not. Have to consider this.

Aslo to shaw: what value exactly do I give each greek letter in the formula? How does the concept work exactly? Also It says in shaw that the values also range from 0-1 how does this work? Cleared..

Answer: Remove the U\_ij The amount that two are overlapping so the intersection of the time windows, divided by the total timewindow length (max upper – min lower). The point is to be able to replace the removed orders with eachother and find new solutions. Implement the weight aswell.

Insert Regret-k: Same question regarding the variety, should I sometimes not choose the highest regret value but rather implement some variety here?

Suggenstion: Add a factor (let say p) that I relate to the temperature so that it goes from being very flexible and not choosing orders so thuroughly in the beginning and that it gets more and more thurough through the algorithm and chooses more similar and more expensive orders in the end of the Heuristic. Definetly implement this. Should be fine.

Suggestion 2: Add a probability 5%’ish that a remove and reinsert operator removes aorund 30% of total orders, jumping from one neighbourhood to the next. Was thinking of random removal in perticular but possibly also remove similar and greedy if I introduce the p factor.

Remove 1- max of 5 and 10% of the orders.

For the 7 order case, find out why I am stuck at the solution. Can I reach the optimal from this in any way? Maybe it will give you an idea to how I can reach it and what I have to do.

If I after a certain amount of iterations do not find any other solution or better solution. Do something wild.. f.eks I accept every solution for 20 iterations. Or I can define an operator that is very diversifying, for eks.

Use Ahmads grubi options to solve also the largest inststances to show what is the upper bound. There is also an option to give information about the.

Results discussion:

Had a mistake in the AMPL model again, so I sent it to you again now as final model. This I am very sure is working as AMPL and Java always find the same soluiton now for smaller instances. AMPL uses about 3-5 hours to solve the instance with 10 orders and around 5 hours to find the solution to the instance with 12 orders.

Results seem very good at the moment. I find the optimal solution in at least a couple runs for each instance. And this is even without some of the unfinished operators. Very excited to see how it changes when I finish regret-k and my special operators.

Removal of orders comment: I had some trouble when I increased the instance size with my operators and I finally figured out that it was because I cant simply remove orders without checking for feasibility in my case. I have to check for feasibility before I remove it. This is due to the factories feasibility. If in solution (1,2,3,1,2,3) element 1 and 3 belong to the same factory and I decide to remove 2 from this vehicle the result will be unfeasible if factory visit limit is 1.

Plan ahead:

Finish operators,

Add finesse to operators, ie. Include chance of larger removal/insertion, include variety to operators etc.

Experiement with different operators.

finish section about the solution method, would give it to you for reviewal as soon as I feel its finished or when I have some questions.

Talk 29.07.2019

AMPL status: Finished, tried solving different sizes of the problem with ampl and around 9-10 Orders it becomes difficult to find a solution in a reasonable time (taking 5 min with 9 orders and about 20 min with 10).

Instances sizes:

Should I keep the instance sizes close to the real 4flow examples I got? Especially refering to the amount of vehicles here and orders/pickup/delivery ratio. Like we have talked about previously 4flow can have enternal amount of cars (so amount of vehicles compared to oders will be high), this will give a very low optimal solution (since each car is picking up very few orders and orders delivered and picked up from to the same location will be bundled together in the optimal solution)..

Make one of each instance with a smaller amount of vehicles and one bigger 5 instances of each. Reduce the amounts slightly below what is needed (start with 20 vehicles, if I only need 10 reduce it to 8). It would also give me alot more analytics if I added the start and finish position of the fleet to the problem with reduced vehicles to assume the position of the logisctics provider.

Make a good example of this since most pdp problems are logistics companies.

I adjusted the sizes of the final instances to the following:



Right now AMPL can solve the first and second problem, it does however use like an hour to solve the second one. I just wanted to see what the optimal solution was for a little bit bigger problem. My algorithm didnt find this solution after 10000 iterations but it got pretty close and I presume it will do so once I finish the new operators.

Check the solution from AMPL and try to put it into Java to see if I can find it.

Right now I am running the algorithm on 5 different lower sized problems while I test the different operators.

Operators:

2-opt: Created a 2opt swap operator that picks a random vehicle and performs a standard 2-opt operation on the route of the vehicle. ie. It performs reverses based on the cost (not distance like standard 2-opt) on the delivery schedule for a vehicle until there are no improvements anymore. For the smaller instances I am testing it is going very fast still but I presume this operator will take alot more time when I increase the size of the problem and also if I would reduce the amount of vehicles per problem (since each car will have to carry more then).

Right now it is only slightly slower than exchange 3 which only does one switch, but it will be much better at finding a local optimum.

Operators still WIP:

Reinsert-Greedy: Removes the most expensive order and reinserts (amount limits is based on the size of the problem but maksimum 5 orders are reinserted) them in their best possible place. Need to do more testing to be sure that this is working as it should. A bit conserned about the time since I have to check the cost of inserting the order in every part of the permutation (not just for a vehicle like the 2-opt). Do you think this will be a problem?

Be smart with how long you continue checking, if i breach f.eks. A timewindow, dont continue.

If its very expensive, try inserting the first better solution but try normal first.

Try to use the cost calculation from previous iterations to save time on calculating the new.

Reinsert-k: Standard regret-k heuristic. Using currently the removal of the similar orders (based on Shaw but adapted a bit to my problem) and then regret-k to insert the orders again. Also working on testing here. Same concerns as above with checking every permutation of the solution for larger instances but I will see during the testing how it will go.

Special Operator 1: Working on a cluster operator. Since the orders from 4flow and in my instance generator are sometimes being transported from/to the same place or places close to eachother (see the tab with the clusters of Europe and Germany), making them very attractive to transport together, I want to let a cluster algorithm run in the beginning and then the algorithm tried to pickup and deliver orders based on the clusters, and always keep the best schedule. Still experimenting with this but I hope this operator will work well.

Special Operator 2: This operator goes through each factory and removes orders that should be delivered in the same factory and then compares delivering them together with the current cost. Very efficient operator That could be good but I am still working on the implementation here. Perhaps I will do it for all factories or something else but I thought this could be good to find some local optimums the other operators cant.

Speical Operator 3: Time windows operator. I thought of an operator that would try to make a delivery schedule of a vehicle more efficient based on the time windows so that it would be possible for the vehicle to deliver more orders on one trip. This does however not seem very good for how the 4flow instances and my instancegenerator work at the moment. This is because I have so many cars and very flexible timewindows so it is not actually beneficial to my current instances. But maybe you have some idea on how else I can use the multiple timewindows to make an operator? Experiment with this, good idea.

If you have any other input on operators and what area I should focus on to maybe get more Ideas please le me know. My plan is to finish as many operators as possible this week and I hope to be doing some experiementing with all of them the week after to see which is working the best and to finalize the instances and the sizes.

I also want to spend a bit time to write ahead. I thought I would start with writing the operator section since that is fresh on my mind and try to finish that part as soon as possible (after I finalize which operators I use ofcourse).

Lets discuss all of this tomorrow in the meeting.

Write about a meeting.

Looking forward to it.

See you soon.

Best Regards

Preben

Email 17.07.2019:

Hello Ahmad,

I found the mistake and after looking for the solution for a really long time in AMPL i finally figured it out. The problem was actually in Java and not in AMPL so very nice to have that mystery solved. Both models are now working for 5 different instances of the smallest size.

However when I moved up in size I encountered another problem that I was working on until my AMPL licence expired. I thought it would work to download the new version, but I think I dont have good enough internet here in the cabin to download it so, I will have to continue that next week when I am back in Berlin.

On the Java side of things I currently have the std output ready for the normal 10x runs etc.

I am also working on several operators that I think will have a great impact on the results. Hope to be testing them properly very soon. Lets discuss them in our next meeting.

I will be back in berlin on Monday and working full(overtime) from tuesday on. Should we have a meeting next week thursday? Would 11 be ok for you?

Hope you are doing well and enjoying the weather in Bergen? Another good summer so far here in Hardanger at least, so we are going back to berlin quite a bit tanner than we left it :D

PS: About the AMPL, just wondering how long the link you posted will be available? I will be in Oslo on the weekend and could download it earliest Saturday, let me know if that will be ok..

Talk to you soon.

Best Regards

Preben

Check tomorrow if I can use a formula contains uniques in AMPL to check if set already contains delivery node. Also find a way to go through each pickup and delivery node from locations and add only pickup/delivery nodes where I can visit both.

Talk ??

Hi Ahmad,

How is it going with you? How is the summer?

Status:

Currently working on the operators and the final output.

I tested the instance generator for 5 different instances of a small type (Ord\_4\_Veh\_3\_Loc\_7) to see if AMPL find the same solution as my Algorithm but I am stuck there on a problem:

My algortihm finds a better solution than the AMPL. The solution seems obvious and from what I can see very feasible but AMPL is not arriving at this solution for some reason. I have attached the AMPL files, the instance files and the results from running my algorithm to the email. Maybe you can see something obvious that I am missing or so? It could be something in the translation of the problem (from Java form to AMPL form) that gets lost but I cannot seem to find it. All information seems correctly translated. This was the case for 3/5 instances that I build. 2/5 AMPL and Java finds the same optimal solution.

Will continue looking for the problem and work on the operator in the next days so I will let you know if I have any more questions here. If you have any input please let me know.

Write you more soon when I have more updates.

Best Regards

Preben

The model seems to be working in both AMPL and Java. I Ran a test on 5 different instances

Talk ??

Status: Instance generator is finished. Started working on good/targeted operators.

Instance generator Questions/updates:

Currently I am generating the pickup/delivery locations for each order as follows:

assign random pickuplocation to order

until there are no more pickuplocations

then assign random pickuplocation

Result example:

if I have 4 pickup and 4 delivery locations the result of the first 4 orders will always be some combination of the 4 locations, for example:

order 1, pickup 2, delivery 4

order 2, pickup 1, delivery 2

order 3, pickup 3, delivery 3

order 4, pickup 4, delivery 1

....

1. After this the pickup and delivery are chosen at random. This was to insure that each location is used at least once but not always in the same order. Is this an ok way to do this?

2. I have randomized the cost of not transport from a lower to an upper bound based on the most expensive price possible.. is this (randomizing) necessary since we dont really have a cost of no transport or should I just set it to hiring the most expensive transport for all orders?

3. The price structure I have generated based on the instance from 4flow and made 3 variations of (3 different type of vehicles, large, medium, small). Then for the small instances I have a smaller version of the structure scaled so that the largest instance uses all intervals (10 distance dimensions and 28 weight dimensions). This way the upper interval is always the same (since the largest vehicle has to be able to pick up a large order if nessecary) but the intervals down to 0 are fewer for the smaller instances.

4. Like I mentioned I have 3 type of vehicles and in each instance I ensure that I have 1/3 of each type of vehicle. The vehicles are not too different, however the smallest one has 1/2 the capacity of the largest. Most orders generated will be smaller than that since they mostly are small at 4flow but maybe 1/10 might be too big for this vehicle. I will experiment on this and make sure the smallest size wont be a problem to solve the problem efficiently. The Largest vehicle is based on the vehicle from 4flow.

5. Locations, I generated locations based on 3 different scenarios. Europe, Germany and Uniform distribution. My Idea is to have 2 instances based on the europe version, two based on germay and one uniformly difstributed (see sheet «Map Coordinates» in the Results excel file). Thought this was a good way to test everything from large scale cluster, to small scale. Let me know what you think here.

6. Time, here i did like we discussed and scaled the travel time with 60% (checked a few travel times between different cities in europe on google and average was 60%).

7. Time windows: I generated timewindows at each location from 2-7 timewindows based on the instance size (The instance from 4flow was demand for a week). Sometimes I generated one to two timewindows per day meaning the maximum amount of days a location is open is 7. Meaning a truck can drive all night deliver one order, drive all night and deliver another, for up to 7 nights. This would entail a travel crossing europe back and forth at least 3 times so I think it should be solveable. Like we discussed the timewindows should be loose so I thought this would be the best. Let me know If you have any thoughts here.

Talk 18.06.2019

Generator:

Should I generate clustered instances? Or other types? Yes do clusters and possibly other instance types..

Up until now I have made no difference between the distances and the times. Should I generate data that have a difference here? Here I thought maybe I could presume that a truck drives 60km/h and therefore drives 1 km per min and then variate this with +-10-20%, or is this the same as for the distances, making random on random kindof..? alfa\*D+teta\*beta\*D -1<teta<1

A couple questions for the algorithm: Right now I have it like this:

for each run:

run1:

generate initial solution, save results before/after

start metaheuristic with initial solution:

find temperature

run alns with calc temperature

report results from alns/temperature run

run2:

generate initial solution....... (same as above)

....

....

....

run3:

....

....

...

and so on..

We agreed that I should choose the temperature before starting the Metaheuristic. Should I exclude the choice of temperature from the data reporting, just like I do while generating the initial solution? Should I choose a new temperature for every run or just once and use the same for every run?

I thought maybe its good to include the temperature in each run (like it is now) so that I also test if this works properly.

The same questions go for choosing initial solution.

If i keep it how I have it now I would report the data like the attached excel file..

Keep it and dont report before T0. If I do something not random consider putting initial outside.

Talk 14.06.2019

Calculating the initial Temp:

Am I only saving the difference in Objective func when I need to accept a worse solution or also for better solutions? It makes a big difference for the initial temperature however when I am testing on the current solution/operators I am not getting a very different result.

Also I am calculating the differences for 1 segment (ie. 100 iterations) enough?

Should I experiment alot with this?

Problem with few worse solution, add all worse solutions to the average, try also to calculate the temperature first then take an average of the temperatures.

If I am still finding very few worse solutions after generating the good instances, try to make a function that finds a better initial solution and then run the algorithm from there. If I find a good way to find an initial solution I can report the results here separately.

Generating data questions:

Any tips on generating good location data that insures feasible solutions and that makes sense? Generate locations as x/y coordinates based on the instances from lars.

Update status

Talk on 11.06.2019

Figured out why swap2 is so good. I very often find a solution that has the same objective value as before but different pickup/delivery or different car. This means that the function e^-|0|/temp = 1.0 will always accept the solution (since objective difference is 0).. is this how we want it? Or should I change something here when the two objectives are the same? It seems to always be like this when the objective function is close to eachother.. is this to support a «local» search especially in the beginning? Since my instances are sometimes consisting of up to 10 of the same type of order being transported from the same place to another this operator (swap) or similar types will be selected with very high probability even though it is unlikely to produce anything useful... let me know what you think about this.

Changed now to a minimum percentage of 5% (see results). This ensures all operators being selected (max for the best operator will be 80% and min for the worst ones 5%. Do you think I should make this «smarter»? Or is 5% an ok probability?

Running times are now also being printed.

Still working on the output data format and the new operators, hoping to finish format soon and operators by the end of this/next week.

Fix an instance generator that includes all aspects of the model. Its important to have for further testing of operators.

Talk on 03.06.2019

Scores on the y axsis and iterations on x axis.

The graphs will be in the paper so keep a fixed format.

Check performance of each operator, by ignoring swap for instance. How does the data change..

set up my program to run by turning on or off the different operators. So that I run the program for each

Talk with Ahmad:

Data results:

- First data results, WIP but a good start

- want to add several runs/datasets to compare

- presenting weights, rather relative representation, than accumulated right?

- Any data you are missing from the results?

ALNS:

- What to do when an operator is not picked regarding scores/weights? Rather leave them the same? They are now slowly deminishing because of the 80% history weight.

Operators, Ideas

- I want to use the fact that it does not make sense to pickup and deliver right after eachother, ie. Permutations like 112233. Sending 3 cars instead will always be better. Possible tweak for all operators, however I am not sure if this is perhaps better to not do as with other operators (exchage etc) 112233 can easily turn into 123321, so an operator ruling out 112233 might work against me.. Need to analyse this further..

- Bundling orders together that have the same/similar starting point. It seems that the data from 4flow are mostly based on many orders transported to and from a few locations. (working on an illustration of the data here). I should be able to build something similar to the relatedness measure from shaw, but I want to adapt this a bit to 4flows specific characteristics. Make sure that the algorithm understands something from the input file and then does something. Make it as little manual as possible. Depend the weight/similarity operator to depend on the type of problem. If I

- k-means clustering is also an option for the last point however I need to see if this will not just take too much time.. can be used in preprocessing to give you insight on the input file based on the locations. We dont want pickup and delivery in two different clusters with the same cars, flow from one cluster to another should be covered with the same vehicles, use to assign orders to the same vehicles.

Print also the size of the instance. done

Then print amount of segments and segment length done

also print time of operators

change the representation of the weights to percentage summing up to 100. done

Change the order of the operators, try the other graph, make also the old graph with absolete done

numbers instead of. done

Make a report so that its easy to see the instance size etc, in general informative

show running time of operators compared to eachother.

Show where I got the best result in the graph.

At the end of each segment, maybe use the following operator:

2 opt looks at all possible combinations of two orders and keeps the best.

Add a preprocessing that changes the/which operators based on the instance, also preprocessing on initial solution generation.

Talk with Ahmad 20.05.2019

Finished the import of 4flow’s data. Its a bit useless as many of the aspects Lars talked about are not included in the model. Trying to make a random generator that generates data based on the data from 4flow.

Generated orderpenalties based on 2 times the worst tarif from the input data. Sound good to you to base it on 2x the «worst case»?

Adapt the penalties during the search to give the model the possibility to not include orders in the beginning and then increase towards end.

Amount of vehicles are assumed given, solve the problem many times and see what I end up with. Look at the original paper and check how they did it. But I should definetly have enough. Find the right and make it depend on size.

Talk with Lars 14.05.2019:

Are stop costs calculated for each supplier/factory or also for each dock?

The tarifs are still a bit of a mystery to me. In the excel sheets there is a different tarif from each origin-destination pair. How are they calculated? I tried dividing on amount of km but this didnt work. I need the basic tarifs to be able to read it in my model. (how much is the tarif per km for the distance 100-200 KM etc) The pair costs from supplier 1 to factory 3 are not relevant to me. I can make an average myself to be able to work with it, but there always seem to be some fixed cost or something in each calculation so it would be very useful to have the raw data.

Talk Ahmad 13.05.2019:

When I am now changing to that orders can have the same origin and destination nodes, what if a solution is to visit the same node twice? For example first order is to pickup from node 1 and deliver to node 3 and the second is from node 2 also to node 3, and solution is to pick up the first order, then deliver it, then pickup second order and deliver it (visiting the same node again). Feel like there is alot of the math that wont be working anymore if I make this change, maybe its not even possible. And if I cant really save any space regarding the nodes its not really any sense in changing it.

I need to make a converter between the input file and the problem.

Would like to describe and discuss the docking constraints in a factory. Lars told me that if I should be able to pick up several orders that belong to the same dock and then deliver all of them to the same dock in one delivery kindof.

Need to add one function for lars to be able to handle the dock visits

Are there any big implications on the mathematical formulation when I change from order nodes to location nodes that you can see?

Talk with Lars 07.05.2019:

Wegen Tarife:

Wir haben besprochen das es für eine intervalle (km und kg) gibt es ein tarif. In die daten scheint es aber mehrere zu sein. Meine modelle rechnen schon mit eine gewissen tarif für eine intervalle und kann nicht den gunstigste aussuchen. Ist das ein problem? Was ist dann der unterschied zwischen beispielsweise LTL||Current\_LTL||LTL\_DE\_W-DE\_W\_100 und LTL||Current\_LTL||LTL\_DE\_W-DE\_W\_1500. Ist die zahl am ende der kilometer distance?

Sonstige fragen:

Ich habe es jetzt so gemacht das ich die daten die ich brauche ausgeschnitten haben (zb. Bei tarife habe ich DE\_W-DE\_W ausgeschnitten als ich die startlocation nicht berücksichtige). Ist das in ordnung? Die tarife sind die pro km oder wie funktioniert das?

Wann berechnet man ein preis von LTL und wann von FTL?

Was is quasi die kriterium von eine FTL Truck?

Was ist die Hubs für? (mein modell hat kein möglichkeit zu unterscheiden zwischen hub oder direkttransport, es wird nur demand von A nach B geliefert)

In die Commodities Tabelle, wie berechnet man die kg? Spalte B hat kg label aber manchmal wert 0?

Checklist Data:

- Prices for different intervals regarding weight – done

- Prices per km?

- Prices per stop?

- Fixed costs?

- Distances between locations – lars delivering, could use dist\_class from hubs\_dist\_tarifs\_assignment..

- Time between locations – gleich distance..

- Demands for each order – done

- pickup and delivery of each order - done

- Weight of each order - ??

- Size of each order - done

- Factory docking capacity?

- Time windows?

- Any Vehicle limitations?

- Weight capacity of each vehicle?

- Different type of vehicles? (size etc.?)

Talk 25.04.2019

Questions:

Regarding solution representation and the representation of Orders, I currently have one pickup node per order and one delivery node per order. Wondering if it could be more efficient for the data from 4-flow to have nodes per location instead. (would mean I have to change a bit of the code/solution representation, but could possibly save us some runningtime/space while reading the data. Do you think I should make this change? Can doif I want to.

Ahmad is not available next thursday..

Talk 12.04.2019

Status:

ALNS – Some open questions, see below

Operators – Made a few basic (quick) operators, remove and reinsert (basic), working on a more complex one

Still open: Random instance gereator, regret-k operator, greedy operator (similar to what I am working on, will be good to compare), targeted removals, similar removals.

Questions to ALNS:

- Should I keep track of all solutions? How do you determine if a solution has already been found, or do you not care aslong as it is different and with a better current objective? Check that its a new solution is not found before, save possibly a string in a hashset to check.

- We give a score to solutions only because they are accepted? So the amount of points delegated wiill depend on the cooling schedule of the simmulated annealing? Also need to be a new solution

- I currently decrease Temperature after each segment (so every 100 iterations?)? Change temperature after every iteration.

- T0 is 1000 and is decreased by 1% each time. Or should I use Boltzman function from Crama and Schyns like in Benchmark? Choose an initial Temperature, If its better, always keep, In the first x iterations I fix the probability to 80%, x should be low. Then use the deltas etc to choose a T0, you gather all the deltas and take the average, do the inverse to find the initial Temperature, After that we decrease the T0 by 0,5% or maybe less, fine tune this parameter. 20% r approx to get a smoothe change.

- With the weights I am currently just giving points as above ( 4 pt, 2pt, 1pt ) and then I have a function that determines the probabilities of picking each based on the points + a part of the previous value (like benchmark paper). Think about using a minimum percent so its never going below perhaps at 1% or so.. Keep track of the weights over each iteration to make a graph.

- I have made a clock to take the time of different parts of the algorithm. What kind of data should I keep for us to compare? Should I time all Operators separately and take an average over the total? Or are you only interested in the total runningtime of the ALNS? Let me know what we should keep and I will summarize some sort of table for you to show you how it is going and how the different heuristics/Operators are working. Better to have more sensors for now to measure, each operators average times usually and make it up to me for now, rather too much

- In the End I am a bit confused. Which part of the algorithm is still Large Neighbourhood? Seems alot like what I build so far is very similar to simmulated annealing somehow.. The focus is on the A-Adaptive. The 2-5 is is the 5 remove and reinsert. When I describe the adaptive part I will refer to a paper. We start with 25% 2-5, and then we change the distribution,

Next week: Holiday, but I will work, any paper recommendations for me to read on the plane? Inspiration for operators? Clustering papers, k-mean clustering, wikipedia article, maybe for initialising the problem or something like an operator.

Talk 04.04.19

- Efficient complete Feasibility check ready. Checks all in time solution.length\*timewindow.amount

- Random heuristics finished, finds optimal solution for my small example.

- ALNS – in progress

- Removal Insertion heuristics – in progress

- Question to the Benchmark paper...

Whenever you develope a new operator. Whenever quality analysis. Check the time compared to amount of iterations to be able to

keep in mind the time

One of the main contributions should be to design an operator that is unique for this problem, Use the adaptive model and keep trak of the scores of the operators So that I can analyse and find which operators that I design myself work well for this type of problem.

2 switch is quick, but you have others that takes one call and tries to take one and find the best placement. Dont let the long runningtime scare me from using the operators.

Talk 28.03.2019

- Random Solution Generator:

- Probably doesnt matter so much since I wont use the random generator in the heuristic but is the following procedure ok for generating random solutions:

1. For each order, I assign the order to a random vehicle

2. For each vehicle I then either pickup new order or deliver already picked up orders

until all assigned orders are done (this part particularly turns into a pretty ugly code)

- Random heuristic

1. Should I report the same as we did in the Assignment in 379?

2. What information do I need if not and what do we use it for? Just testing or benchmarking?

We need a type «Random» operator later to drastically change the solution and get unstuck.

For the operators we need to keep them feasible. Input is a feasible solution and output the same.

One normal operator is to take one order and reinsert it in a good place. Another version of this is to take x amount of orders and remove them and reinsert them. Then you can end up in a situation where you cant put one order somewhere. Regret k-function it is called. Look it up in

aks about the feasibility checks.

- Status update

- Objective function is done,

- Feasibility checks.

- Keep the fool proof feasibility check for the final solution.

- Finish simple feasibility check

- Heuristics

- Basic solution , dummy

Next step: Framework of our algorithm. ALNS, Follow the framework not the operators, We will have 20% probability to pick an operator, then you get a solution, and keep track of the performance (this makes it adaptive). Give points to operators during 100 iterations and then update probability. 1 point for new solution, 2 points for new better solution 4 points for best so far solution. Basic paper has description.

Boltzman function, acceptance like simmulated annealing, Temperature start with 1000 and decrease to 99%. Will be logarithmic.

Talk 18.03.2019

Discovered with the volume that if a load is heavier/bigger than the capacity of a car the Big M in constr. 7 and 10 does not work (value has to be less than 0 when vehicle not visiting. Should we change this or leave it? Leave for now. If it is the case at 4flow that this happends the program will anyways not mind.

Volume constraints seems other than that to work fine in AMPL. The toughest constraints win

Define all indices, also introduce alfa and beta. Done

Try to formulate amount of more specific as numbers of or other.

Amount of weight after visiting node i\* check this for liv and others. - done

When I define B and Z in table try to find a better wording more precise. done

Ci has a problem in same table. Done

Add Ndf instead of just Nf since Nf is a subset of Nd. done

Change A\_v after instroducing the indices before the sets. done

Amount is written wrong in first Set. - done

P\_i is a set of time window indices. And T\_i is timewindows. done

Maximum sounds like we optimized, probably better to use the biggest load on the vehicle. done

Talk 13.03.2019

For tomorrow I have the following:

- Update on programming status

- Solution representation example: [1 2 9 8 0 3 10 0 4 11 0]

Make sure I dont loose any runningtime, possibly do a preprocessing (only one time) to pair nodes if that can solve potential problems.

- Keep A\_v and for each vehicle. And define alfa and beta for each vehicle.

- Attachments input?

- Still missing data from Lars...

- Some private news.

TimePlan: finish by easter and fine-tuning after

20.-25. april ahmad is gone

23.-26. june a conference

29. juli until 02.august holiday

Talk 07.03.2019

Try to keep the objective function quick.. Rather see if I can change the solution representation if neccesary...

Talk 03.03.2019

Q1: Instance generator creates data readable for AMPL right? So far I made the java program able to read AMPL data so that instances can easily be generated and sovled by both models.

Talk 28.02.2019

Changed the constraint we talked about to exclude fixed costs and added the constraint regarding pickup and delivery time. This worked and the model should now be complete.

Had a talk with Lars and we looked at some of the data together. It could be that I need to change the model a bit due to the following points:

- Not just weight but also volume is relevant for loading. Might need another constraint to make the model take the volume into account.

- I will get anonymous data to use for quite a large instance. I can play with this data however I want.

- Also the origin and destination might also be a problem regarding the data that we do not take into account.

- Some data are incomplete. Time windows and factory stop limits are definetly missing and we will have to generate this our selves.

- Might be other problems coming up when I have studied the data further.

Q1: Thought that I would keep the model as it is now and continue writing the paper as it is. Might be several problems regarding the data, and I find it a bit unprofessional that Lars seem to have a new constraint every time I talk with him. So I will rather deal with whatever adaptations he wants later and continue working for now. Do you find this reasonable? Only add this if I can do it without too much work and nothing more.

Q2: How should I generate the missing data? Should I just make up some data that is reasonable and where a good solution should be possible? Try to vary it but not too constrained.

Q3: Data that I generate might then be a bit similar (lots of equal time-windows etc.) is this a problem? Dont waste too much time on the data. Just check how the real data varies and generate data based on the real instance. So dont solve his now just use it to generate my instances and to make an illustration of the problem and use this to describe the problem and solution in the paper.

Take a sample of my problem and illustrate it on a map of germany and use this in the solve and present a solution of the problem.

Make a program to generate instances and use ampl to solve it. Gradually increase the instances until AMPL doesnt work anymore. Make vehicle types and make the program able to select different properties to the data.

Talk 21.02.2019

Q: Time windows are getting bound upwards (always choosing last time window if possible. Dont understand why (isnt AMPL trying to lower the variables?). Not really a problem for solution but could be a problem later when I have bigger routes and want to present solutions. It would look better if they were bound downwards so that I would end up with the fastest solution instead of the slowest possible. Also even though travel from last node til destination node is free it sets the time at final node to upper timewindow at final node. Solve this by running another model with only arrival time as variable, if it is important

Fixed costs are always included since my model always choose an interval for weight/distance inspite it being 0. Should probably do something about that, especially if I want to have too many cars in my instances, which should lead to some cars being idle and my model will then still include the costs for these cars. Solve this by changing constraint no 21 equal to left had side of constraint no. 4. Check constraint 18-21 that they are still valid.

Fix Time constraint, no. 11 from Hemmati et al. Needs to be there to ensure subsequent delivery

Talk 14.02.2019

- Have to make sure that the cost in the maximum weight dimention is increasing if not the model will just ajust l\_max upwards as long as everything else stays the same. Should however not be cheaper to send something heavier (other costs will be more expensive).

- Add high costs/distance/time on going to starting point and from finish. done

- remove all constraints and solve problem with limits on variables

- add one by one and finally remove limits

- model: start with calculating the objective function, and solution representation

- then move on to feasibility and check against AMPL results

- Then build model and see if I get the same solutions.

Talk Ahmad 31.01.2019

* Change the fix parameter - done
* make o(v) and d(v) parameters - done
* waiting time at node i and maximum weight in v - done
* What is n in 2n - done
* what is the alpha and beta in the set A - done
* make it clear what is the difference between Pi and Ti - done
* change the variables b and d sets descriptions - done
* Work on defining the nodes as they sometimes share locations and use that to explain L and S etc. Define S before Ls – done, though need to describe better
* Add the remaining variables at the end of the constraints and try to minimize them - done
* If I can add a similar constraint as 21 and use the same b
* Next steps: improve writing and start programming.
* If it is super difficult to use the solution representation from the course, change it but try to leave it minimize it. Write one sample and try to find the objective value and see if its enough.
* Give one solution to AMPL with a small sample and check that AMPL gets it right. Make a small solution by hand and use this for everything, also programming later.

February 11 Monday is cancelled.

Q&A 17.01.2019

Question for Lars: Do we have different costs per leg? Or is it purely per km/kg? My presumtion only per km.

Changes to make:

Cost structure and paper: Make it clear that we have the responsibility to plan the travels with which vehicles etc.

Q&A Lars 22.01.2019

For Ahmad: How do we want to use/publicise the instances?

There is not really different costs per leg but there is a possibility to expand the model to take into account the start and goal location. The cost structure can be different starting from one region to another. Lars says its also ok if we rather adapt the heuristics later to this part.

It also might be that sometimes you have costs per stop (which means that since we are travelling from i to j and not from supplier to factory) that we are not able to count it with xij (unless we somehow say that costs from i to j within a supplier/factory is 0). Lets discuss how to handle this.

These are both things that we dont have to handle but are more nice to haves that I could adapt the heuristic model for 4flow rather.

24.01.2019

We need to say we have 4 vehicles, 10 stops, bla bla but nothing more.

Need one instance, realistic,

we will not make the instance public, we are not going to write about the customers. We write general about the instance, number of request, number of vehicles, etc. To show how small or big is the problem. To illustrate the problem its nice to have a map with locations, but it doesnt have to be a real example, its completely fine if its fiction but ofcourse the closer to a «real» problem the better for the model.

Many interesting problems, would you be willing to define a good comprehensive model for a phd who will start in 3 months for 4 years.

Preben: Think about the problem if we open a facility we have a fixed cost, if not we dont have it. 5 binary variables, sum equal to one, determines

sum of (j \* xo(v)j) will give me a j which I can multiply with only parameters to determine which cost structure I should use i.e which location I start from.

If our visit in a factory is visited our variable

Focus on point 1 & 2 question, on the weekend and see if we need to adapt alot..

28.01.2019

T1: Adding weight and fixed costs to the model. Should work. Change to greater than instead of max..

T2: Taking starting location into account.

What about all the extreme cases here? If you go from the middle, first north then south. If it is simple and you can only start from cologne or berlin og munich and its always simple then it doesnt make sense for the mathematical model. Then we will work on this later rather as it requires some special knowledge of the geography of the company. Need to ask Lars what all the edge cases are and how it should work for a general case.

T3: Adding stop costs to the model.

Should work.. need to change to Ci and Cj to make the model more general. Write it in the pdf and send to ahmad.

Questions for Ahmad:

* Why do we have to add half the costs? Cant we say that we add all costs after a stop? So an example o(v) ->C=0 i -> C=10 j -> C=10 k -> C=10 d() C(total) = 30
* Very interested in working together about a project. He will bring the topic around the house and make a list of problems in Optimization direction and then let you know about a meeting. I will send you both an email to start your thread and either he or someone else will be the contact person.