orbit

32

March • 2019

Planering med energi i fokus • Learning to Play Othello from Scratch • Writing Beautiful Optimization Code

ORbit

medlemsblad for
Dansk Selskab
for Operationsanalyse
og
Svenska Operationsanalys-
föreningen

Redaktion:
Ansv. Julia Pahl (jp)
Tor Fog Justesen (tfj)
Tomas Gustafsson, SOAF (tg)

DORS
DTU Management, bygn. 424
Danmarkt Tekniske Universitet
DK-2800 Kgs. Lyngby
Telefon: +45 4525 3385
Fax: +45 4588 2673
E-mail: orbit@dorsnet.dk

Næste deadline:
10. October 2019

Tryk:
Ecograf Gruppen

Oplag: 455
ISSN 1601-8893

Forside: Billede leveret af Lars Relund Nielsen og Erik Jørgensen (se side 25).

Leder

Kære læser

Det er første gang for mig som redaktør og at "gå i Sannes fodspor" - som vi siger i Tyskland - gjorde mig lidt nervøs. Sanne hjalp mig meget med at komme i gang som redaktør for ORbit og med de praktiske ting såsom at håndtere overgangsbrønden.
Mange tak for det! Det var helt fantastisk!

Jeg er også meget glad for, at så mange forfattere har bidraget til at vi kunne få gang i ORbit. Jeg var lidt bekymret for at få nok bidrag, men de sidste uger har vist, at vi har et meget aktivt og engageret samfund.
Vi har nu spændende artikler, der dækker et bredt udvalg af aktuelle emner lige fra grøn energiplanlægning, machine learning, writing optimization code, auction optimization for assigning TV broadcast spectra, udvikling af infrastruktur til arktiske havne til de mere humoristiske bidrag.

Jeg ser frem til samarbejdet og ønsker god læselyst.

Julia Pahl, ny redaktør

Aktuelt om DORS
juli 2010

Medlemsskab
Kontingentsatser for 2010

Personlige medlemmer	
(incl. ph.d.-studerende):	270 kr./år
Studerende:	60 kr./år
Firmamedlemmer:	3200 kr./år
Institutmedlemmer:	1800 kr./år

Indbetales på Giro 9123865 (reg.nr. 1199)

Sekretariat
DORS
DTU Management
Bygning 424
Danmarks Tekniske Universitet,
2800 Kgs. Lyngby
e-mail: dors@dorsnet.dk
Internet: www.dorsnet.dk

Svenska operationsanalysföreningen

Lagom till att ett nytt verksamhetsår tar sin början är det också dags att ge ut ett nytt nummer av ORbit. Årsmötet har denna gång hållits vid Uppsala universitet som bjöd på ett intressant upplägg för seminariet. Presentationerna hölls av tre kollegor vid lärosätet och de hade sitt fokus på olika typer av metoder för optimering:

Mathematical Optimization for 5G, Di Yuan

SAT and SMT for Program Verification, Philipp Rümmer

Combinatorial Optimisation with Constraint Programming (at Uppsala University), Pierre Flener

Årets vinnare av exjobbspriset, Oliver Sundell och Anders Nordmark från Chalmers, höll också en inspirerande presentation av sitt arbete Tactical Decision-Making for Highway Driving.

En fråga som aktualiserats under året och som vi höll en separat diskussion om under årsmötet är betydelsen av ORbit och vilket format den ska ha framöver. Slutsatsen var att för att kunna fatta ett bra beslut kring detta så behöver vi få en bättre bild av vad våra medlemmar tycker om ORbit och styrelsen tar tacksamt emot synpunkter kring detta. De i övrigt mest aktuella frågorna inom SOAF är årets konferens och att intressegruppen för forskarstuderanden håller ytterligare en aktivitet.

Hälsningar

Elina Rönnberg, ordförande SOAF

Indhold

Aktuellt om SOAF

juli 2010

Individuellt medlemskap: avgiften är 150 kr/år.

Företagsmedlemskap:
Avgiften för företaget/institutionen beror på antalet OR-intressenter enligt följande tariff:
Akademiska institutioner: 1500 kr/år
1-5 intressenter: 1500 kr/år
6-10 intressenter: 3000 kr/år
över 10 intressenter: 4500 kr/år

I företagsmedlemsskapet ingår 2-6 exemplar av ORbit, beroende på företagets storlek.

Betala in på postgiro: 19 94 48-2 (Svenska Operationsanalysföreningen)

e-mail: sekreterare@soaf.se

Internet: www.soaf.se

Företagsmedlemmar:

- FOI Försvarsanalys
- Swedish Institute of Computer Science (SICS)
- Optimization Partner
- Optimal Solutions
- Jeppesen
- Avdelningen för Kommunikations- och transportsystem vid Linköpings universitet
- Centrum för Intelligent Automation vid Högskolan i Skövde
- Preference
- Transrail
- Industrial Optimizers

Av Martin Waldemarsson

Planering med energi i fokus

Med anledning av den senaste tidens uppvaknande kring klimatfrågans akuta läge har jakten på förnybara bränslen intensifierats. Hur och när den fossilbaserade ekonomin kommer överges återstår att se, men mycket talar för att skogsindustrin med sina biobaserade energibärare kommer spela en fortsatt betydande roll, såväl i Sverige som i andra länder med mycket skog. Skogsråvarans betydelse i energiförsörjningen av svensk industri framgår i Figur 1.

Från Mill Mission till Energy Mission

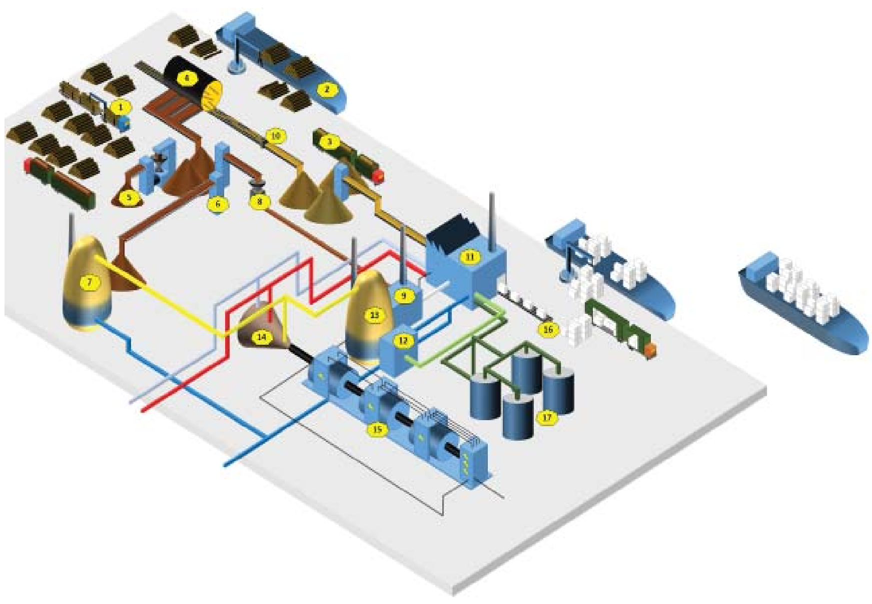
Södra Cell är en av de stora aktörerna i den svenska skogsindustrin och är, med sin årliga volym på c:a 1,7 miljoner ton massa (2017), en betydande leverantör av pappersmassa på den internationella marknaden. Planeringsmässigt utgör råvaruanskaffning, schemaläggning av recept/kampanj, samt distribution av massa till kund en stor del i den årliga budgetplaneringen. För detta används en optimeringsmodell (Mill Mission) som utvecklats av Helene Lidestam (f.d. Gunnarsson), Mikael Rönnqvist och Dick Carlsson och finns presenterad i Gunnarsson et al. (2007).

Huvudinkomsterna kommer från försäljning av pappersmassa, men betydande bidrag kan också uppstå från en rad olika biprodukter. För att analysera hur stor betydelse dessa biprodukter skulle kunna ha i den årliga budgetplaneringen har modellen Energy Mission skapats som finns presenterad i sin helhet i Waldemarsson et al. (2013) och Waldemarsson et al. (2017). Denna utgår från Mill Mission-modellen, men vissa avgränsningar och förenklingar i planeringen av råvaruanskaffning och distribution har gjorts och i modellen läggs istället fokus på att väva in biprodukter och energibärare i receptstrukturen.

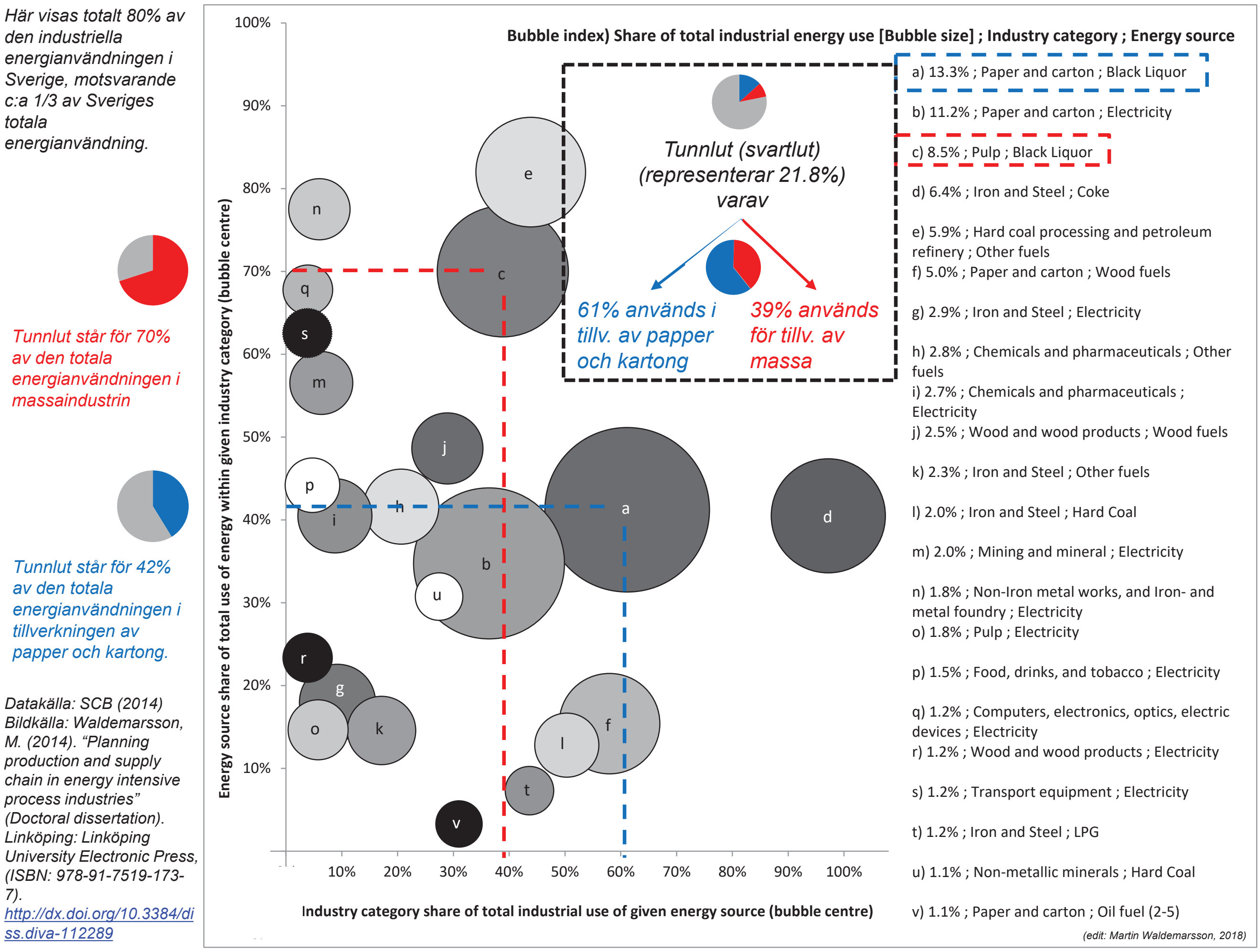
Systembeskrivning

En något förenklad systemöversikt över ett typiskt fabriksområde presenteras i Figur 2.

Flödet börjar egentligen i något av de många råvarudistrikten runt om i landet där råvaran (massaveden) hämtas vid bilväg och körs med lastbil till fabriksområdet (se Figur 2 [nr. 1]), massaved transporteras även med båt eller järnväg [2]. Råvara i form av massaved och flis kan också anlända från sågverk [3] och i viss mån även från utlandet. Vid fabriken barkas massaveden [4] och den första biprodukten, bark, avskiljs från huvudlinjen för att sedan antingen bli pellets [5], beredas [6] för förbränning i ångpannan [7], eller pulveriseras [8] för användning i mesaugnen [9]. Den barkade massaveden flisas sedan [10]. In till själva massafabriken [11] trans-



Figur 2. En något förenklad systemöversikt över ett typiskt fabriksområde



Figur 1. Olika typer av energislag som används i respektive industrikategori (data hämtad från SCB, 2014)

porteras en viss typ av flis, eller en viss kombination av olika sorters flis (gran, tall, barr, löv), beroende på vilken slutprodukt som önskas. I massafabriken finns bland annat kokprocesser, blekningsprocesser, torkprocesser och förpackningsprocesser. Från kokeriet pumpas tunnlut (s.k. svartlut) till indunstning [12] för att sedan processas i sodapannan [13], som dels ger grönlut till mesaugnen [9] och dels genererar högtrycksånga till turbinen [14]. Turbinen, som även får högtrycksånga från ångpannan [7], driver i sin tur en (eller en serie av) generator(er) [15] som producerar elektricitet. Elen kan sedan användas i de egna produktionsprocesserna eller säljas på den externa elmarknaden. Turbinen ger även mellantrycksånga och/eller lågtrycksånga, beroende på hur mycket av högtrycksångans energi som systemet föredrar att omvandla till el. Både mellantrycksånga och lågtrycksånga behövs för diverse produktionsprocesser och det antas i modellen att överskottet av dessa kan säljas på en extern marknad. Totalt kan 15 olika slutprodukter från de fyra olika bruken produceras (när studien gjordes ingick även ett sulfatmassabruk i Norge i försörjningskedjan). Vilken produkt som produceras styrs genom val av recept. Det finns 20 olika recept som vart och ett är unikt för det enskilda bruket. Vilka recept som kan användas vid vilket bruk beror på produktionstekniska förutsättningar, men även på de inställningar som kan göras vid respektive bruk. Det finns således en begränsad mängd produktionsalternativ där en viss uppsättning recept kan användas vid ett visst bruk. Att byta inställningar (och ändra uppsättning av recept) är även förknippat med diverse kostnader, inte minst från utebliven produktion. Merparten av recepten resulterar i en enda massaprodukt, men några recept ger två olika sorters massaprodukter. Massaprodukterna fraktas sedan till kund via båt, tåg och/eller lastbil [16].

Vid sidan av huvudprodukten pappersmassa samt biprodukterna bark och tunnlut, faller det även ut en del andra biprodukter, såsom metanol, tallolja och terpentin, från produktionsprocessernas olika steg [17]. Även dessa antas kunna säljas på den externa marknaden eller användas internt för den egna energiförsörjningen (genom att styra dess utfall).

Modellen

Det unika med Energy Mission-modellen är att den kan ta hänsyn till de olika produkternas, energiintensitet vilket grundar sig i att de olika råvarorna som behövs har olika proportioner av energiprodukter inbäddade i sig och därmed sin egen energikarakteristik. Det innebär att varje recept blir unikt i hur mycket energi som kan tappas av i form av energiprodukter: bark, tunnlut, metanol, tallolja och terpentin. Låt oss nu fokusera på modellens energirelaterade delar. Hur mycket energiprodukter som ett recept kan generera formuleras i följande villkor:

$$u^N_{nilrt} \leq \left(\sum_{p \in P} u^P_{ilprt} \Big/ \sum_{p \in P} r^P_{pr} \right) r^N_{nr}$$
$$\forall n \in N, i \in I, l \in L, r \in R, t \in T$$

där $u^N{}_{nilrt}$ är variabeln för producerad mängd av energiprodukt *n* vid bruk *i* och lina *l* enligt recept *r* i tidsperiod *t* och där $u^P{}_{ilprt}$ är motsvarande variabel för massaprodukt *p*. Parametrarna $r^P{}_{pr}$ och $r^N{}_{nr}$ anger hur mycket av massaprodukt *p* respektive energiprodukt *n* som genereras under ett dygns produktion enligt recept *r*.

Massaproduktionen kräver också olika mängder av energibärarna; el, mellantrycksånga och lågtrycksånga. Därtill har de olika brukens pannor olika effektivitet och dess turbiner olika förmågor att producera el. Något förenklat kan man säga att det vid ett av bruken går att få ut elenergi på upp till 30 % av högtrycksångans energiinnehåll, medan motsvarande för de andra tre bruken ligger kring 18 %. Ju mer el som tas ut, desto mindre energi finns kvar att tappa ut i form av mellantrycksånga. Om bruket väljer att ta ut maximal mängd el blir det inget över till mellantrycksånga utan då fås bara lågtrycksånga. För att balansera åtgången av energiprodukter i omvandlingen till energibärare används följande villkor:

$$\sum_{n \in N} w^B_{nit} b_i = w^{MPS}_{it} + w^{LPS}_{it} + w^{El}_{it}, \quad \forall\, i \in I, t \in T$$

Där $w^B{}_{nit}$ är variabeln för hur mycket av energiprodukt *n* vid bruk *i* i tidsperiod *t* som omvandlas med verkningsgraden b_i vid bruk *i* till variablerna mellantrycksånga $w^{MPS}{}_{it}$, lågtrycksånga $w^{LPS}{}_{it}$ och el $w^{El}{}_{it}$. För att sen få rätt mix av energibärare i relation till gränserna för hur mycket el som måste ($h^{min}{}_i$), eller kan ($h^{max}{}_i$), produceras vid respektive bruk används följande villkor:

$$\sum_{n \in N} w^B_{nit} b_i h^{max}_i \geq w^{El}_{it}, \quad \forall\, i \in I, t \in T$$
$$\sum_{n \in N} w^B_{nit} b_i h^{min}_i \leq w^{El}_{it}, \quad \forall\, i \in I, t \in T$$
$$\sum_{n \in N} w^B_{nit} b_i \geq w^{MPS}_{it} + w^{El}_{it} / h^{max}_i, \quad \forall\, i \in I, t \in T$$

Därtill måste det råda energibalans mellan producerade energibärare och hur mycket av energibärare *e* som används i den egna produktionen $w^{ME}{}_{eilprt}$, som säljs $y^{SE}{}_{eit}$, eller som går till spillo $x^{WE}{}_{eit}$. För detta används följande villkor:

$$\sum_{l \in L} \sum_{p \in P} \sum_{r \in R} w^{ME}_{eilprt} + y^{SE}_{eit} + x^{WE}_{eit} = w^{MPS}_{it}$$
$$\forall\, e \in E_{MPS}, i \in I, t \in T$$
$$\sum_{l \in L} \sum_{p \in P} \sum_{r \in R} w^{ME}_{eilprt} + y^{SE}_{eit} + x^{WE}_{eit} = w^{LPS}_{it}$$
$$\forall\, e \in E_{LPS}, i \in I, t \in T$$
$$\sum_{l \in L} \sum_{p \in P} \sum_{r \in R} w^{ME}_{eilprt} + y^{SE}_{eit} + x^{WE}_{eit} = w^{El}_{it}$$
$$\forall\, e \in E_{El}, i \in I, t \in T$$

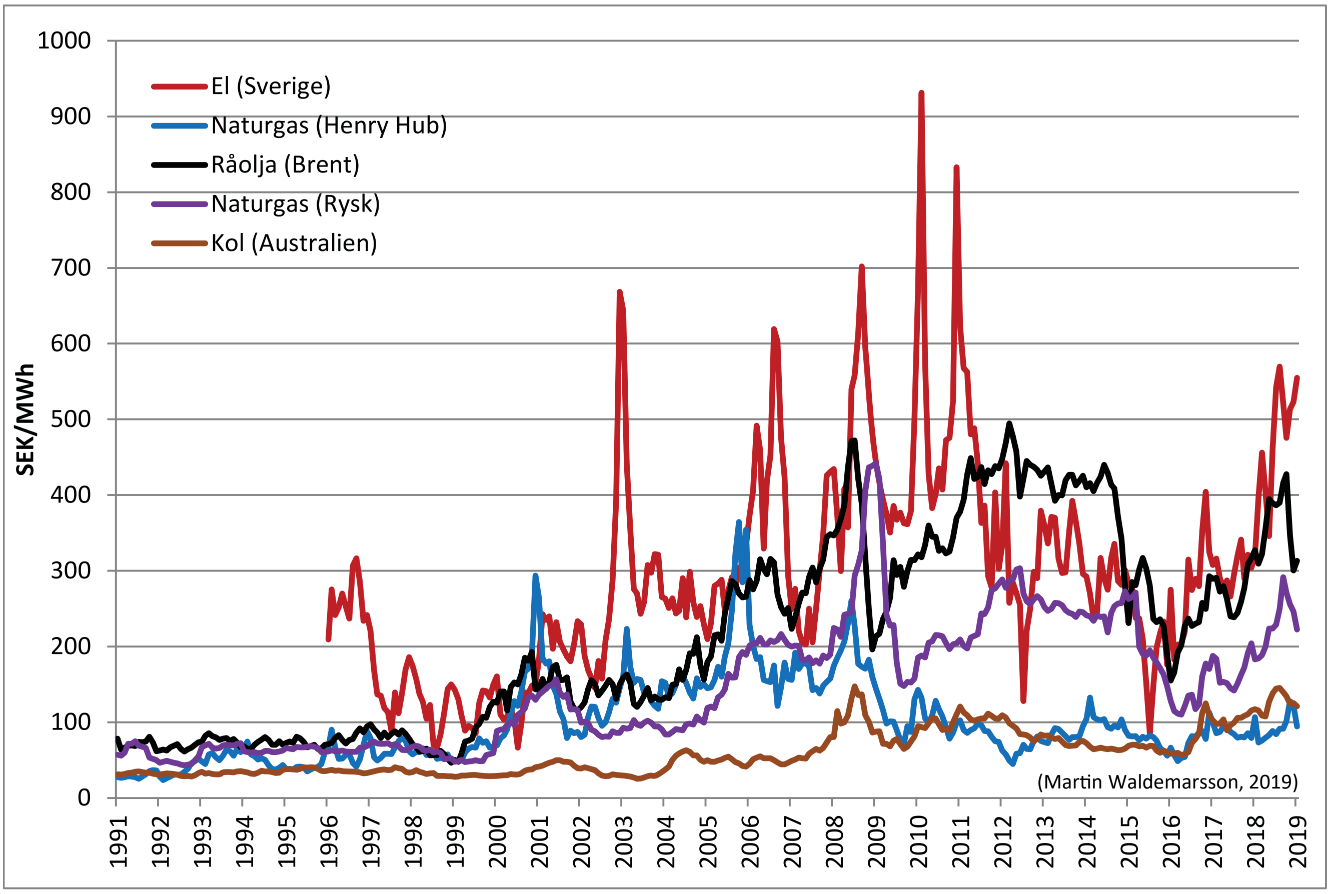
Slutligen krävs även ett par villkor för att styra hur mycket av energibärarna som åtgår ($e^{min}{}_{er}$), eller som tillåts ($e^{max}{}_{er}$), i produktionen av massa:

$$\sum_{p \in P} w^{ME}_{eilprt} / a^E_{er} \geq e^{min}_{er} \left(\sum_{p \in P} u^P_{ilprt} \Big/ \sum_{p \in P} r^P_{pr} \right)$$
$$\forall e \in E, i \in I, l \in L, r \in R, t \in T$$

$$\sum_{p \in P} w^{ME}_{eilprt} / a^E_{er} \leq e^{max}_{er} \left(\sum_{p \in P} u^P_{ilprt} \Big/ \sum_{p \in P} r^P_{pr} \right)$$
$$\forall e \in E, i \in I, l \in L, r \in R, t \in T$$

där $a^E{}_{er}$ motsvarar en nyckel för hur mycket av respektive energibärare e som åtgår när en enhet massa produceras enligt recept *r*.

Planeringen görs över ett år indelat i tolv tidsperioder. Efterfrågan på pappersmassa styrs av diverse kontrakt som består av en eller flera kundorder. Om ett kontrakt accepteras måste samtliga order i det kontraktet uppfyllas. Det finns även val-



Figur 3. Månatliga genomsnittspriser på naturgas, råolja, och kol (data från: Indexmundi, 2018) samt månadsgenomsnitt på elpriser i Sverige (NordPoolSpot, 2018) i SEK/MWh. Rådata är hämtad i USD/MMBTU (gas), USD/fat (olja), samt USD/ton (kol) och har valutajusterats med en per månad genomsnittlig växelkurs SEK/USD samt omvandlats med hänsyn till dess värmevärde. Således motsvarar en MMBTU 0,2931 MWh, ett fat olja 1,7 MWh och ett ton kol 7,32 MWh. Detta tar dock inte hänsyn till energibärarnas exerginivåer. Noteras bör även att priserna inte inkluderar skatter, avgifter och transmissions- eller transportkostnader

bara order samt fasta order som måste uppfyllas. Efterfrågan baseras på en årlig volym och därmed ges modellen valmöjligheten att förlägga produktionen av massa i valfri tidsperiod, givet att bland annat produktionskapaciteter och lagerkapaciteter inte överskrids.

Vikten av energi

På energimarknaden är det mer regel än undantag med turbulenta priser och kraftiga prissvängningar (Figur 3). Historiskt sett har elpriserna (i Sverige) generellt varit högre under vinterhalvåret än under sommarhalvåret. Vintertid finns ofta också större avsättningsmöjligheter för olika typer av överskottsenergi. Frågan som uppstår blir hur mycket som skulle kunna avsättas och hur mycket det skulle kunna vara värt.

Med denna prisvariation är det således värt att reda ut om det skulle löna sig att förlägga den mest energiintensiva produktionen till vinterhalvåret, då energipriserna förväntas vara högre och de mindre energiintensiva recepten till sommarhalvåret. Fördelen med modellen Energy Mission är att den gör det möjligt att undersöka om och i så fall, hur mycket en sådan omplanering skulle kunna vara värd.

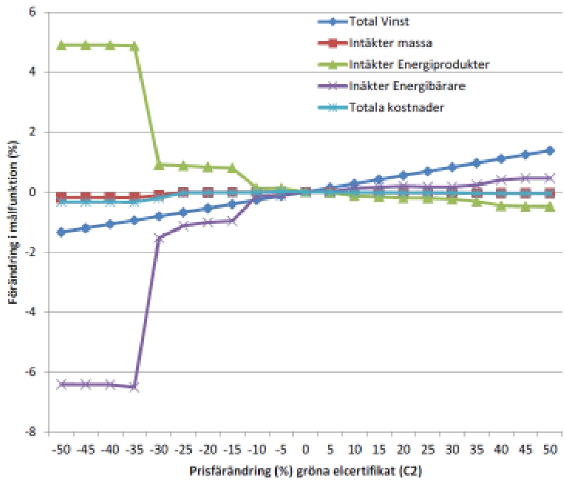
Vikten av vinst

Modellen är vinstmaximerande och har som mål att maximera intäkter från försäljning av massaprodukter, energiprodukter, energibärare och gröna elcertifikat och minimera kostnader för transport och inköp av råmaterial, produktion, byte av produktionsalternativ, massadistribution, lagerhållning vid råvarudistrikten, samt kostnader för lagerhållning av råvaror, massaprodukter och energiprodukter vid bruken.

Vid sidan av de villkor som redan beskrivits finns förstås även en uppsättning villkor som bland annat garanterar lagerbalanser, att inte råvarutillförseln överskrids och att valda order och kontrakt uppfylls.

Stresstest

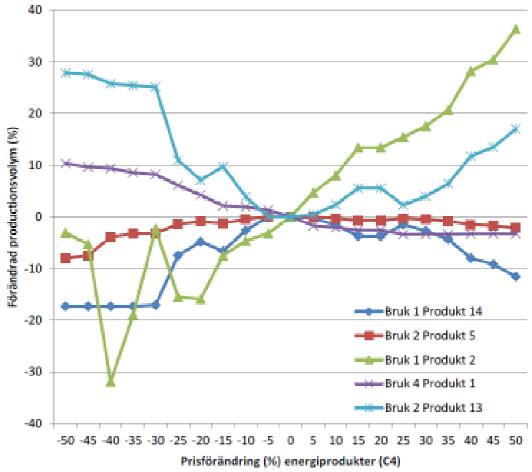
För att analysera hur systemet skulle uppföra sig i olika situationer med varierande energipriser har olika scenarier testas. Låt oss titta närmare på några av dessa. För att kunna jämföra med något initieras ett startscenario med standard-inställningar, som sedan ligger till grund för att mäta avvikelser gentemot andra scenarier. Fem scenarier där priset på olika energislag går (i steg om 5 procentenheter) från -50 % av default priset till +50 % testas. På detta sätt testas modellen genom att justera priset på el (C1), gröna elcertifikat (C2), både el och gröna elcertifikat (C3), energiprodukter (C4) och samtliga energirelaterade parametrar (C5). Resultatet från dessa scenarier skvallrar om att energiprisförändringar kan vara starka motiv för att prioritera vissa produkter framföra andra, att i mån av tillgänglig kapacitet omfördela produktionen från ett bruk till ett annat mot ökad distributionskostnad, samt att planeringsmässigt styra viss produktion till vinterhalvåret. Gemensamt är att den totala vinsten ökar till synes linjärt med ökat pris i samt-



Figur 4. Resultat från scenario C2 över hur några målfunktionsposter påverkas av en prisförändring av gröna elcertifikat

liga av dessa scenarier. Det går också att urskilja brytpunkter när modellen kastar om i planeringen och tydligt går från ett hörn i utfallsrummet till ett annat, vilket framgår i scenario C2 (Figur 4) där en prisförändring på gröna elcertifikat tydligt pekar på vikten av att behålla energiprodukter, för att i stället producera och sälja mera grön el. Konsekvensen blir att ett helt annat produktionsschema genereras, men också att andra kontrakt och order prioriteras på grund av den enskilda energiprisförändringen.

Figur 5 ges ett exempel på hur modellen omprioriterar några produkter i några av bruken vilket tyder på att modellen påverkas av de olika produkternas energikarakteristik och hur väl de platsar i de nya förutsättningarna.



Figur 5. Resultat från scenario C4 över hur några produkters produktionsvolymer påverkas av en prisförändring på energiprodukter. prisförändring av gröna elcertifikat

Från teori till praktik

Att Södra Cell helt skulle kasta om sin planering och vända upp och ner på massakontrakt enbart på grund av vissa rörelser på energimarknaden bör ses som något långsökt. Detta är heller inte tanken med Energy Mission. Snarare ger modellen företaget en möjlighet att sätta prioritet på vissa produkter, bruk, order, kontrakt osv. i några givna framtidsscenarier. En tanke och rimlig början skulle vara att använda sig av någon slags differentierad prissättning som har koppling till produktens energikarakteristik. De skulle exempelvis kunna ge rabatt på produkter som hjälper företaget att producera mer grön el när vintern är hård och elpriset högt.

Referenser

Gunnarsson, H, Rönnqvist, M., Carlsson, D. (2007) ”Integrated production and distribution planning for Södra Cell AB”, Journal of Mathematical Modeling and Algorithms, 6(1), 25-45. DOI 10.1007/s10852-006-9048-z

Waldemarsson, M., Lidestam, H. & Karlsson, M. (2017) “How energy price changes can affect production- and supply chain planning - A case study at a pulp company”, Applied Energy, 203, 333-347. DOI: 10.1016/j.apenergy.2017.05.146

Waldemarsson, M., Lidestam, H. & Rudberg, M. (2013) “Including energy in supply chain planning at a pulp company”, Applied Energy, 112, 1056-1065. DOI: 10.1016/j.apenergy.2012.12.032

Martin Waldemarsson är teknologie doktor i produktionsekonomi och universitetslektor i kvantitativ logistik på Institutionen för teknik och naturvetenskap vid Linköpings universitet. Studien på Södra Cell genomfördes i samarbete med Helene Lidestam, Martin Rudberg och Magnus Karlsson inom ramen för Processindustriellt Centrum Linköping (PIC-Li) med Stiftelsen för Strategisk Forskning (SSF) som huvudfinansiär.

DORS’ firma- og institutmedlemmer

Institutmedlemmer

- Datalogisk Institut, Københavns Universitet
- Institut for Virksomhedsledelse og Økonomi, Syddansk Universitet
- Institut for Matematiske Fag, Aarhus Universitet
- Afdeling for Anvendt Matematik og Statistik, IMF, Københavns Universitet
- CORAL, Aarhus University
- DTU Management, Danmarks Tekniske Universitet
- Department of Materials and Production, Aalborg University

Firmamedlemmer

- A.P. Møller – Mærsk
- Copenhagen Optimization
- DONG Naturgas A/S
- DSB S-tog
- Hermes Traffic Intelligence
- Københavns Lufthavne A/S
- MaCom A/S
- MOSEK
- Novo Nordisk (CMC Clinical Supplies)
- Optivation
- PDC A/S
- PostNord
- QAMPO
- Rapidis
- Transvision A/S
- Trapeze Group Europe A/S

artikel

By Jens Vinther Clausen

Learning to Play Othello from Scratch

Machine Learning (ML) has been a hot topic in recent years due to its many applications, e.g. image recognition, speech recognition, self-driving cars, etc. Its latest success can be partially attributed to hardware improvements, but also to algorithmic innovation. Due to their well-defined rules and often high complexity, games are frequently used when developing new algorithms. Recently, machine learning has been used with great success on a number of games: in 2016, Google DeepMind's[1] AlphaGo[2] beat the 18-time Go[3] world champion Lee Sedol 4 to 1[4] and more recently, in 2018, AlphaZero beat one of the strongest chess engines Stockfish 8 in a 1000 game match[5]. This text explains some of the algorithms and ideas in machine learning and presents an agent that has learnt to play the board game Othello[6] entirely from self-play.

Machine learning and deep neural networks

ML is a branch of Artificial Intelligence (AI) that relies on learning from large sets of data. In ML, a model is constructed based on a set of data, known as "training data"; the model is then used to make predictions or decisions. One specific type of model in machine learning is artificial neural networks (ANN). The structure of ANNs is inspired by that of biological neural networks, i.e. animal brains. An ANN consists of nodes (neurals) and connections (synapses) between the neurals. The neurals of an ANN are organized in layers, see figure 1. The input signals travel through the network and are subjected to various multipliers and functions in each of the connections and neurals until it is finally output. In a basic neural network, each neural is connected to all neurals in the next layer, called a fully connected layer. Some networks make exceptions to this by, e.g. having fewer connections, having connections bypassing one or more layers or having neurals with connections to earlier layers. If an ANN has more than one hidden layer it is a deep neural network.

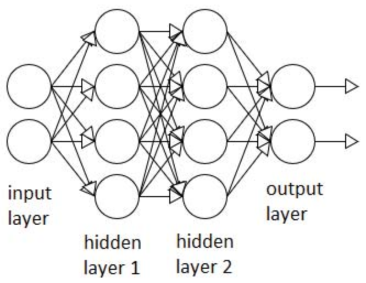


Figure 1. Artificial neural network with two hidden layers

Supervised learning

When machine learning is used to predict a known output from a given input, it is called supervised learning. An example of this is the version of AlphaGo that beat Lee Sedol, often referred to as AlphaGo Lee, which trained a network based on a large database of human moves. Each input was a game state and the corresponding output was the move chosen by the human player as well as whether he ended up winning the game or not.

Reinforcement Learning

Reinforcement learning deals with agents that have to take actions in some environment while maximizing performance. Unlike supervised learning, the training data does not need to consist of correct input-output pairs, instead the goal is to learn how to maximize performance. AlphaGo Lee also used reinforcement learning; after having learned from human moves, it continued learning from self-play. It played a large number of games against itself, learned from those, then played again, etc. gradually improving its performance. A more recent version of AlphaGo, called AlphaGo Zero[7], differs from the earlier on multiple points; most notably it skips the supervised learning portion and learns completely from self-play, the only input are the rules of the game. After just three days of training, AlphaGo Zero beat AlphaGo Lee 100 games to 0.

An Othello playing agent

Othello is a board game that shares some resemblances to Go but is much less complex. In both games, two players take turns placing a stone of their color in a cell on the board. However, in Go the board is 19 by 19 cells, it has

artikel

an average game length of 150 moves and a branching factor of 250. In comparison, Othello is played on an 8 by 8 board with an average game length of 58 moves and average branching factor of 10. The agent presented here is inspired by AlphaGo Zero and uses many of the same ideas to learn Othello.

The Othello agent is essentially an algorithm that given the current game state can output the move it finds the most promising. It uses two main components, Monte Carlo tree search (MCTS) and a deep neural network.

Network

The overall structure of the neural network is illustrated in figure 2. Given a game state, the network does two things, both from the perspective of the player who's turn it is. The policy head assigns a probability to each of the possible next moves, a high probability means the move is considered good. The value head evaluates the state, 1 means the player is winning and -1 that he is losing.

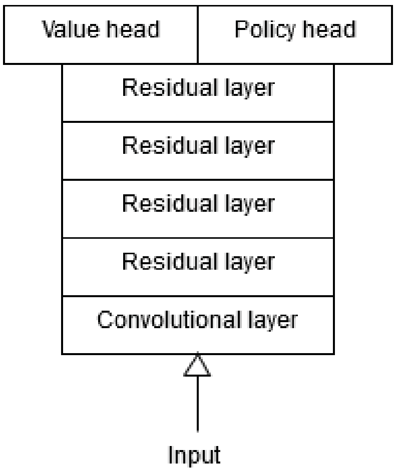


Figure 2. Structure of the network of the Othello agent

The network uses convolutional layers, which have fewer connections between the neurons, making them faster. The main benefit, however, is that they allow the network to learn structure in the position of the inputs relative to each other. This technology is known to work well for image recognition where the position of pixels relative to each other is highly important; analyzing the cells of a game state is essentially much like analyzing the pixels of an image. Each of the residual layers can be bypassed by connections directly to the next layer, this allows the network collapse into fewer layers while training, which has been known to work well for image recognition as well. Each of the residual layers actually consist of two convolutional layers, likewise, the two heads both have convolutional and fully connected layers.

Monto Carlo tree search

MCTS searches a tree of game states that has the current game state as its root node, the tree is gradually expanded throughout the search by simulating games. Each simulation choses its moves from information about performance from earlier simulations (exploitation), while also steering towards new paths in the tree (exploration). The balance of exploitation vs exploration is achieved using a so called upper confidence bound (UCB) function.

In the Othello agent, the neural network is used to guide the search. This is done by using the probabilities from the policy head in the UCB, hence favoring moves that the network finds promising. The evaluation from the value head is used whenever the search reaches a game state it has not seen before, ideally giving a good estimate on which player would eventually win the game if it continued.

Training the agent

The agent was trained for a week albeit on a single core CPU that is probably not the best on the market. Recently GPUs have been used to drastically reduce training time for deep neural networks; this could probably be applied here as well. The training consists of three steps, self-play, update, and evaluation. In self-play, the agent plays a number of games against itself to generate data. The update step is where the network is actually trained using the data from self-play; the network is essentially learning from its mistakes and successes. Finally, in the evaluation step the agent plays against the best version of the agent so far, if it wins a significant margin of the games it will henceforth be considered the best version. These three steps are repeated iteratively, gradually improving the strength of the agent. Across the entire 7 days of training, the agent played approximately 10,000 games against itself.

Results

In order to show the learning process of the agent, every 50th iteration/generation played 10 games against each other, each going first in half of the games. The best version at the end of training also played. The results are shown in figure 3. Each cell shows the number of points (a win gives 1 point, a loss 0 and a draw 0.5) from the perspective of the row player, e.g. the 350th generation scored 3 points in its 10 games against the 300th generation. The agent is clearly improving throughout the first 200-300 iterations but seems to stagnate towards the final iterations.

The best version of the agent was also matched against another agent that uses a different tree search, minimax with alpha-beta pruning[8]. More importantly, rather than using a neural network to evaluate game states, the minimax agent weighs a number of properties of the game state that are considered important by human players, e.g. it is good to control the corner cells of the

artikel

board. The minimax agent was implemented by Hjalti Dam Joensen as part of his bachelor's project. Figure 4 shows the results of playing 10 games between the two agents with varying thinking time per move. The scores are from the perspective of the MCTS agent, e.g. when both agents were given 0.1 seconds per move, MCTS scored 7 of 10 possible points. The MCTS agent won most of the games, even when given much less time than the minimax agent, scoring a total of 449.5 points out of 490 possible. In a second test, the minimax agent beat the MCTS agent 10 to 0, but only because it was given a time limit of 50 seconds compared to 1 second for the MCTS.

In conclusion, the MCTS agent managed to teach itself Othello without learning from humans and even ended up becoming a better player than the minimax agent, which relies heavily on human knowledge in its evaluation.

References

[1] Deepmind. URL: https://en.wikipedia.org/wiki/DeepMind.

[2] Alphago deepmind. URL: https://deepmind.com/research/alphago/.

[3] Go (game). URL: https://en.wikipedia.org/wiki/Go_(game).

[4] Alphago versus lee sedol. URL: https://en.wikipedia.org/wiki/AlphaGo_versus_Lee_Sedol.

[5] Alphago versus stockfish. URL: https://www.chess.com/news/view/updated-alphazero-crushes-stockfish-in-new-1-000-game-match.

[6] Othello. https://en.wikipedia.org/wiki/Reversi.

[7] David Silver, Julian Schrittwieser, Karen Simonyan, Ioannis Antonoglou, Aja Huang, Arthur Guez, Thomas Hubert, Lucas Baker, Matthew Lai, Adrian Bolton, et al. Mastering the game of go without human knowledge. Nature, 550(7676):354, 2017.

[8] Minimax alpha-beta pruning. URL: https://en.wikipedia.org/wiki/Alpha%E2%80%93beta_pruning.

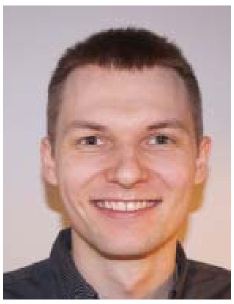
	0	50	100	150	200	250	300	350	best	sum
0	-	0	0	0	0	0	0	0	0	0
50	10	-	0	2	0	7.5	0	0	0	9.5
100	10	10	-	5	0	5	0	0	0	25
150	10	8	5	-	0	5	0	0	0	18
200	10	10	10	10	-	0	5	5	5	45
250	10	2.5	0	5	10	-	5	7.5	10	40
300	10	10	10	10	5	5	-	7	5	52
350	10	10	10	10	5	2.5	3	-	5	45.5
best	10	10	10	10	5	0	5	5	-	45

Figure 3. Results from games between generations

		Minimax						
		0.1	0.2	0.5	1	2	5	10
MCTS	0.1	7	9	10	9	5	8.5	9
	0.2	10	5	6	7	7	8	10
	0.5	10	9	10	10	10	8	9
	1	10	10	10	10	10	10	10
	2	10	10	10	10	10	10	10
	5	10	10	10	10	5	10	8
	10	10	10	10	10	10	10	10

Figure 4. Results against minimax

Jens Vinther Clausen is a Ph.D. student at the Technical University of Denmark, Department of Management Engineering. His main research interest is exact integer linear programming methods, specifically when coupled with decomposition techniques. Other interest are machine learning and artificial intelligence.



By Richard Oberdiek

Writing Beautiful Optimization Code

When we are taught optimization, we learn about duality, column generation and metaheuristics. And sure, we often also solve a lot of example problems along the way. However, most often, at least in my experience, you either will be given an OPL file[1], or you get away with scripting away in your favorite modelling environment like GAMS or Xpress Workbench.

However, if you ever try to do something more involved, or send your code to somebody else to work jointly on it, then it suddenly hits you that just happily scripting away is not the most sustainable path forward when it comes to code development. So why don't we write beautiful, easy-to-maintain code from the beginning? Because we don't know how to. After all, we are optimization experts/operations researchers/data scientists, not software developers.

Well, it's not as hard as it looks, and in this article I'm going to present you with my take on some basic principles on how to write beautiful optimization code. In short, these principles are:

1. Use a general, object oriented programming language
2. Use classes for your model and indices
3. Every variable is a property in your model class
4. Every constraint, objective function and callback is a method in your model class
5. There are no other methods except these in your model class. If there are, have a good reason for them.

Use a General Object-Oriented Programming Language

I love GAMS as much as anybody. However, if you every want to do anything else except solving an optimization problem, you need another software. Want to plot something? Export and import in Python and go from there. Data prep? External as well. So if you model is in an algebraic modelling environment, it gets ripped away from the rest of the code that exists around it. While it might seem easier at first to get started with e.g. GAMS than with Python, the interfaces (or APIs[2]) for these languages have become so good that this extra piece of software is redundant.

Let me give you an example. Below, I will write some Python code without any explanation. See whether you can understand what's going on:

```
model = xp.problem("A test problem")
x = {k : xp.var(vartype = xp.binary, name = f'x_{k}') for k in range(10)}
model.addVariable(x)
model.addConstraint(xp.Sum(x[k] for k in range(10)) <= 5)
model.solve()
```

The only line that is probably a bit weird is the second line, because it involves some syntax. But after looking at it for 2 minutes, you will probably realize that this does nothing else than generate a list of 10 binary variables called x_0 to x_9, introduce the constraint $\sum_k x_k \leq 5$ and solve the problem.

This was a valid Python example using Xpress and Python (but it would be very similar in all the other major languages). So why am I showing you this? Because it illustrates that the power of GAMS of being human-readable is not a real power anymore, as more and more languages are starting to have this.

There are though two reasons where GAMS (and similar tools) still have the edge: first and most importantly this is model creation speed. Especially for large model instances, this can be a problem in interpreted languages such as Python (and a reason why compiled languages such as Julia are faster in that regard). There are often[3] ways around this, however it can be cumbersome and require more time fiddling around in the code. The second thing is abstract nature of a model: in general programming languages, we always work with specific model instances when we instatiate the model, while in GAMS we can

1 This is a IBM CPLEX specific file format.

2 API stands for Application Programming Interface

3 How to do this is very solver and language-dependent, so I won't dwell on it here.

write the actual mathematical equations. In my opinion this is not a huge trade-off (if at all), but for some people it may be.

So what language should you use? In my opinion, the number one example is Python due its extremely large and active user base, followed by Julia and C#. If you don't know any of these languages, start with Python[4], since it is extremely intuitive and you can find tons of tutorials out on YouTube.

Why Object Orientation?

Great, now that we agree that we should use a general-purpose language, but why do object-oriented programming? What is even object orientation? Let me explain this with an example: Say you want to calculate the area of a rectangle (A=wl, where w is the width and l is the length). Intuitively, you may do something like this:

```
def rectangle_area(width, length):
return width * length
```

This means, you would simply write a function that takes two numbers and returns the result. While this is a great way to do things for some tasks, more complex programming often becomes cumbersome this way. But it does not have to be this way. Some very smart people (see e.g. in [2]) came up with the idea of classes and objects. Let's see what that means:

```
class Rectangle:
def __init__(self, width, length):
        self.width = width
        self.length = length
        def area(self):
return self.width * self.length
```

So what are we doing here? Instead of simply writing down an equation, like in rectangle_area, we create an instance or a specific rectangle, which has a given width and length. And then we use these properties to evaluate the area:

- This means if you want to "do" a lot of things to this rectangle, you can have all of this information contained in one class. This makes it safe and easily readable.

- You can use classes in other classes. So for example you could make a Point class, which is then used in a Line class, which is then used in a Triangle class etc. This makes for intuitively readable code.

In terms of mathematical programming, it represents more precisely how we think about our optimization problem, and the more our code follows our intuitive thought processes, the easier it is to understand. Let's see in the next sections what I mean by that.

Use Classes for Your Model and Indices

Whenever you start coding up a new optimization problem, you should always start with a Modeller class. This class contains everything you need for your optimization problem: your objective function(s), your variables, your constraints and your callbacks. But nothing else. If you were to draw up a diagram of the flow of your program, this would be were you write "Optimization".

Why? Because this way you have everything in one place. So if something doesn't work, you can check block by block why it doesn't. Equally, if you are sure that it is correct, you can easily port it and just use it without thinking about it. Therefore it represents the logical entity that you anyways think about when you write your problem.

But what about the indices? Well, optimization, and mixed-integer programming in particular, needs a lot of indices: visited cities, products, time indices, you name it. So when you write down your model on paper, something like this

$$x^u(t,c,a)$$

may not be uncommon. But you don't really want to continue writing `x[t,c,a,u]` in your code for every instance of x, because it becomes cumbersome. The solution? Classes[5]! Every index tuple you use for a variable should be its own class. So in this case:

```
class event_one:
def __init__(self, t, c, a, u):
        self.t = t
        self.c = c
        self.a = a
        self.u = u
```

This way, it becomes `x[event]`, and you can easily filter this when formulating constraints. So e.g. say you want to write down the following constraint:

$$\sum_{c,a,u} x^u_{t,c,a} \leq 5, \quad \forall t$$

4 I'd recommend Pyomo as it interfaces with almost every solver out there.

5 In some languages, e.g. Python, it may in fact make sense to use a `namedtuple` instead of a class. But that is diving too much into the specifics.

You can simply write[6]:

```
for t in range(n_t):
        model.addConstraint(xp.sum(x[event]
        for event in events if
        event.t == t) <= 5)
```

where n_t is the number of *t* variables. While this does look a bit intimidating at first, it becomes second nature as soon as the basic principle is understood. The advantage? Slicing and dicing becomes super easy and can be done on the fly, even for very complicated cases. Another advantage is the following: let's say you wrote your model, however then you suddenly realize you want to add the index p to your variable. Or anything with relation to the indices needs to change. If you hard-coded `x[t,c,a,u]`, this is going to be really (!!) annoying. However, if you have used classes, then you simply add p and `self.p` in the `__init__` method of the class, and you are good to go.

In short, it gives your code extreme amounts of flexibility, while again representing something logical. For example, if you have a routing problem, you may have $x^o_{(d,v)}$ to represent that a vehicle *v* goes from origin *o* to destination *d.* This is a logical brick in your model, so just call it that: `x[tour]`.

Every Variable is a Property in Your Model Class

When you build up your `Modeller` class, then each variable used in the model should be a property. Why? Because that makes them available to the entire model (i.e. to all the callbacks, constraints and objective functions), and that is precisely what we want to do.

There are certainly cases where you don't need to do this, however I follow this rule almost always, since I find it so much nicer to work with, and it also makes intuitive sense to me: your variables are properties of your model, that you use in your constraints, callbacks and objective function.

Every Constraint, Objective Function and Callback is a Method in Your Model Class

Often when we formulate our model, we have constraints like "Demand satisfaction", "Flow balance" etc., i.e. one or multiple lines in our model belong to a logical entity that expresses a limitation on the current system. Such a logical entity should be packaged away in a method/function. So you should write something along the lines of:

```
class Modeller:
def __init__(self):
        self.model = xp.problem()
        self.x = None # We leave this out for
        clarity
def connection_limit(self):
self.model.addConstraint([x[connection] <=
connection.limit
for connection in list_of_connections])
```

Why do this? It feels cumbersome, doesn't it? No, not really. Because now, when you build your model, you can simply call `connection_limit`, like you would intuitively want to do in your optimization model. As you can see, object orientation is the very natural vessel for optimization models to live in. The same holds for callbacks and objective function. They just become a method the Modeller class, and you are done. Awesome!

There are No Other Methods Except These in Your Model Class. If There Are, Have a Good Reason for Them

This basically means that you should keep your `Modeller` class clean. Don't throw your data preparation or post-processing into this, build your model, solve it and get out. That is all the class does. This modular thinking is key for object-oriented programming, because it enables you to test and review code much easier. If you send me a code of 400 lines with a mathematical model next to it with what it is supposed to do, I can probably understand it within an hour. If you give me a 2000 line beast with some data prep and post-processing and plotting in there, it'll probably take me a day.

Other Development Practices and Conclusions

While rules are boring, coding principles really help you to write beautiful code. Trust me, your colleagues and your future self will appreciate. But these are only guidelines. Don't take this as the gold standard that you should never deviate from. Try to understand what the idea behind these principles is, get an intuition and then use that.

Once you have done that, you can venture into practices on what you should do when you code, i.e. about the process. The two items most important items are:

1. Code review: grab a co-worker/fellow PhD student etc., and just ask him/her to look over the code for an hour, and tell you everything that does not make sense. You'll be surprised as to how many things there are that make sense in your head but don't make sense to others. There

6 In reality I would write this using list comprehension, but to make it easily accessible, I kept it like this.

are a ton of tools out there to help you do this, and if you are worried that it takes too long: trust me, in the long run it will save you so much more time because your code quality will constantly increase.

2. Unit testing: a unit test is a small script that you execute, and where you expect a result. Let's take our area from the Rectangle class from before. A unit test would give in the values 5 and 3 and see whether 15 comes out. While this might seem like a lot of work, it is extremely effective at catching errors, because you sometimes just lose sight and put a minus where a plus should be, or a > instead of a <. At least I do. Unit tests prevent these things from becoming problems. This is also another reason you should use genera programing languages, because to the best of my knowledge, GAMS and all the other algebraic modelling languages do not offer unit testing possibilities.

If you have any ideas of extra principles, or think these ones need revising, or have general comments, hit me up at r.oberdieck@gmail.com or on Twitter under @ROberdieck.

Acknowledgements
I would like to thank Ruth Misener, Michael Bussieck and Miles Lubin for their comments on draft versions of this article.

References:

[1] https://github.com/amontalenti/elements-of-python-style/blob/master/README.md#the-elements-of-python-style

[2] http://web.eecs.utk.edu/~huangj/CS302S04/notes/oo-intro.html

Richard Oberdieck has a B.Sc. and M.Sc. from ETH Zurich and a PhD from Imperial College on advanced process optimization, where he focused on the development of algorithms for a specific class of optimization problems. After his graduation in 2017, he joined Ørsted as a Numerical Specialist, where he develops optimization tools deployed across the entire company to create a world that runs entirely on green energy. Privately, he is married and has one son, and lives in Hvidovre.

Her kunne din artikel have været!

Send dit bidrag til orbit@dorsnet.dk

By Jakob Blaavand

Repurposing the Radio Spectrum – delivering on the promise of next-generation mobile services

More and more devices in our daily lives require radio waves to work. These includes smartphones, Wi-Fi networks, over-the-air TV and radio, remote controls and baby monitors. We consume more and more data on our smartphones[1] and with 5G telecommunication on our doorstep, there is no end in sight for demand for high quality spectrum. The part of the electromagnetic spectrum usable for communication is called the radio frequency spectrum[2].

National telecommunications regulators manage the radio frequency spectrum in each country: in Denmark it is Energistyrelsen, and in the US it is the Federal Communications Commission (FCC). The regulator sets the telecommunications policies and objectives and in particular determines in which geographical regions the spectrum can be used and for what purpose.

Worldwide there is little unassigned spectrum left to support the rapid growth in demand for mobile data. One way to increase the supply is to repurpose spectrum from its existing use to a higher value purpose. The frequencies in the band 500MHz – 1GHz are the prime spectrum for communication, because they have both very good coverage properties and high capacity, two properties that usually do not coincide in other bands. For these reasons, this part of the radio spectrum was assigned to TV broadcasting in the 1950s and

1 Current growth rates are about 50% annually, see the 2018 Ericsson Mobility Report (https://www.ericsson.com/assets/local/mobility-report/documents/2018/emr-q2-update-2018.pdf)

2 Specifically 20 kHz to 300 GHz

1960s. In 2010 through scenario analysis and the use of optimisation methods, the FCC discovered that assigning channels to over-the-air TV stations in a more efficient way could free up parts of the valuable 600MHz spectrum in the US. The released spectrum could then be made available to mobile operators for their future deployment of 5G mobile networks[3]. The FCC also found that over-the-air TV broadcasters could be packed more efficiently if also Canadian TV broadcasters were given new channels, and the FCC estimated how much spectrum could be released if some TV broadcasters would be willing to relinquish their broadcast licenses (for a monetary compensation). In the past, the TV channel assignment has primarily been done manually. However, with 3,000 TV broadcasters and their requirements for interference-protection, finding a channel assignment, let alone an optimal one with respect to an objective function such as minimizing the use of a particular set of channels, is a near impossible task to do by hand. It is only through the use of bespoke optimisation techniques and state of the art solvers that the FCC was able to conduct such analyses and implement a visionary market-driven repurposing of spectrum.

In 2012, based on the FCC's studies, the US Congress passed a law paving the way for the world's first two-sided spectrum auction, which would repurpose spectrum from TV broadcasters to mobile operators. The two sides of the auction are 1) buying spectrum from TV broadcasters, and 2) selling it to mobile operators. The auction would not close until the revenue raised by selling spectrum exceeded the costs of purchasing licenses from TV broadcasters. This auction, which is known as the Broadcast Incentive Auction (BIA), started in March 2016 and ended a year later in March 2017. At the end of the auction, 84 MHz of spectrum had been repurposed from TV to mobile use. In the process, mobile operators paid $19.8bn for the spectrum of which $10bn went to mobile operators for compensation for either transferring to a new channel (in a lower-quality TV band) or relinquishing their broadcasting license.

A full description of the BIA is beyond the scope of this article. Thus, here we will focus on how OR and optimisation in particular played a key role, not only in the preparation for the BIA but also as a key component within the auction mechanism itself. Specifically, we will consider the optimisations that converted the relinquished TV broadcast licenses to new mobile licenses of the highest quality. In doing so, we will only be scratching the surface. A fuller account of the BIA and the mathematical details of the optimisations are given in [1].

Interference Constraints

An essential component of the BIA was the participation of the TV broadcasters. The more broadcasters that were willing to participate the more spectrum could potentially be released to the mobile operators. FCC's mandate from Congress included TV broadcasters' right to refuse to participate or to drop out of the auction at any time, but they could potentially be required to move to a different channel of the same broadcast quality. In other words, all broadcasters that remained on air after the auction, were guaranteed to reach the same number of people after the auction as they could before the auction (the viewers might just have to retune their TVs).

To honour this guarantee to the TV broadcasters, the FCC had to protect the covered population for each TV station in all optimisations. To do this, the FCC determined for every pair of TV station and broadcast channel which of each cell in a 2x2 km grid covering the entire North America could receive their TV signal. Combining this data with census data about the population in each grid cell, the FCC determined the size of the covered population for every TV station. In any reassignment of TV channels, the size of the covered population should be approximately the same. For example, if two neighbouring TV broadcasters were assigned the same channels, then their coverage areas might overlap. In the overlap, the signal from neither of the two broadcasters will be viewable because the signals will interfere with each other. If the overlap is more than 0.5% of one

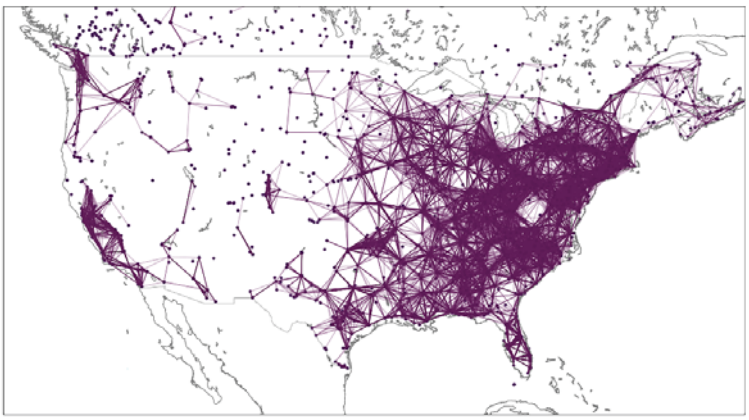


Figure 1: The TV broadcast interference graph. Nodes are TV broadcasters and an edge between two nodes indicates that TV broadcasters' coverage areas overlap by more than 0.5%

3 Spectrum Analysis: Options for Broadcast Spectrum, OBI Technical Report No. 3, FCC, June 2010. (https://transition.fcc.gov/national-broadband-plan/spectrum-analysis-paper.pdf)

of the broadcaster's covered population, then they could not be assigned the same channel to avoid interference.

Finding a channel assignment satisfying all the interference constraints can be thought of as a graph colouring problem. The graph has TV broadcasters as nodes and edges between them if there is potential interference when broadcasting on the same channel (See Figure 1). Colouring nodes on a graph so that no adjacent nodes have the same colours is a well-studied problem. One formulation of the graph colouring problem is as an integer program with a binary variable for each pair of node and colour (determining whether the colour is used at the node or not) and pairwise constraints corresponding to the edges of the graph preventing the nodes at the end of each edge to have the same colour. There is an additional constraint that each node must be given a colour. This formulation is particularly suitable for the FCC's requirement, as it is easy to extend the formulation to include more constraints in the integer program and add a meaningful objective function.

The determination of TV-TV constraints was not only restricted to broadcasters in the US. Radio waves do not care about political borders, so an important part of a spectrum regulator's job is to ensure cross-border harmonisation of the spectrum bands and prevent cross-border interference. Therefore, it was crucial to establish the TV-TV constraints with Canadian and Mexican TV broadcasters too. Canada decided to go even further and to have their TV broadcasters be repacked as part of the repacking in the US, as this would make their 600MHz band available for mobile operators. This was very important to the FCC because otherwise the north of the US would be severely limited in the repacking possibilities. Mexico, however, did not participate in the repacking, and it created strong constraints on which colourings were possible along the US-Mexico border.

Optimisations

The mobile licenses for sale in the BIA provide mobile operators with usage rights of a specific set of frequencies in a restricted geographic area. The US is divided into 416 such geographic areas. A TV license does not conform to these areas. It is therefore not possible to do a one-to-one trade between TV broadcasters and mobile operators. Furthermore, the spectrum is most useful to mobile operators if it is contiguous across its service areas. Therefore, optimisation is needed to maximise the quality of the new mobile licenses by reassigning channels to TV broadcasters. This is partly done by keeping TV broadcasters below 600MHz and mobile operators in 600MHz – 700MHz, to the greatest extent possible. However, it may not be possible to keep all TV broadcasters below 600MHz, in which case some TV broadcasters will use the same spectrum as the mobile operators. This can cause TV- Mobile interference. To maximise the value of mobile licence, the FCC had to ensure their quality and therefore the FCC wanted to find a channel assignment that minimised the potential for TV-Mobile interference.

The TV-Mobile interference is modelled through continuous variables measuring the quality of a mobile license by the amount of interference from TV broadcasters using the same frequencies in the same geographic area as specified in the mobile license. Adding these variables (and the constraints determined by the TV broadcasting channel variables) turns the integer programming formulation of the graph colouring problem into a mixed integer program.

In total, the FCC solved 18 different optimisation problems in sequence, each one preserving the results of the previous optimisations. Each optimisation implemented a different spectrum management policy. For example, to clear an extra channel in Canada, to maximise the quality of the new mobile licenses, or to gather interfering TV broadcasters on specific channels. Finally, changing the channel on a broadcaster's equipment is not easy. TV broadcasting towers are usually very high and are often placed in inaccessible locations where crew and equipment is flown in by helicopter. It requires specialist engineers to climb the towers and change the equipment. There is a limited supply of tower crews who can do this. To speed up the process of making the spectrum available to the mobile operators (as the economic value only can be fully unlocked when they get access to the spectrum and start using it!), the FCC wanted to keep as many stations on their current channels. Another consideration was that up to 0.5% of a TV station's covered population could be lost due to interference in the new channel assignments, so another optimisation was to minimise this loss. Furthermore, each station was assigned an indicative cost of transferring to a new channel (to replace old transmitters, build a new tower, etc.) and a further optimisation was to minimise these transition costs.

About 3,000 TV broadcasters were repacked in the FCC optimisations and along with the mobile licenses, this led to an optimisation problem formulated as a mixed integer program with 100,000 binary variables, 30,000 continuous variables and 10 million constraints. The full sequence of optimisation problems took several weeks to solve, using a bespoke distributed solver running on a large cluster of servers, developed specifically for this purpose. The distributed solver was built around Gurobi 6.5, the latest version of the Gurobi solver at the time.

Benefits

The benefits of the BIA should not only be measured as the size of the auction revenue. The main benefits are to the US consumers because the new spectrum can be used to address the increasing demand for mobile data. For the mobile industry, the BIA boosts the roll-out of 5G networks enabling them to facilitate economic growth and thereby to unlock more economic value than the initial $20bn in upfront spectrum payments. The economic growth will come from the new products and services that are enabled through consumer access to fast mobile broadband. The total economic value created through the BIA is therefore a significant multiple of $20bn.

The BIA furthermore establishes a reference for further spectrum repurposing initiatives (next incentive auction is already being planned in the 39GHz band) .

In recognition of these benefits and the excellence of application of OR, FCC's BIA Team won INFORMS' prestigious 2018 Franz Edelman award.

Spectrum Management and OR

Spectrum management is a new application area for OR. The preparation of the BIA used modelling and optimisation to do extensive "what-if"-scenario analyses to convince stakeholders (US Congress, in particular) of the potential benefits of the BIA mechanism.

Spectrum managers in other countries are taking up the idea of using optimisation to organise their bands. In the UK, government agencies pay an annual fee to incentivise less use of spectrum and release it to other users. Building on the work in the BIA, we, at the Smith Institute, have worked with one agency to explore how much of their current spectrum in a specific band could be released while still supporting their current systems, and consider different scenarios where some systems are retired or moved to different bands altogether.

Through examples such as the BIA and efficient spectrum allocation in the UK, spectrum regulators are seeing how optimisation can be used as an efficient decision-making tool and are starting to appreciate the economic value it can unlock.

References

[1] Tarnutzer B, Centazzo C, Leese R, Blaavand J. Unlocking the Value of Spectrum Using Optimization tools. Colorado Technology Law Journal. 2017; 15.2: 393 - 434

Smith Institute:The Smith Institute (www.smithinst.co.uk) is a UK-based consultancy company that specialises in the application of mathematics, statistics and advanced analytics to solve business problems for businesses and governments across a variety of sectors such as telecommunications, energy, rail, fast-moving consumer goods and security.

The Smith Institute was one of a small group of external consultants working for the FCC on the Broadcast Incentive Auction. Our role was to validate that all algorithms and optimisation models were fit for purpose, to audit the algorithms and the models, to provide advice on accuracy and speed improvements, and to do system integration testing.

Jakob Blaavand is the Director of Mathematics at the Smith Institute. He has a DPhil in mathematics from the University of Oxford, and a M.Sc. and B.Sc. in mathematics from Aarhus University. He works on optimisation and forecasting problems across multiple sectors, specifically telecommunications, energy and rail. He is a leading member of the Smith Institute's spectrum team, who received the UK OR Society's 2018 President's Medal for work on the Broadcast Incentive Auction.

By Marcel Turkensteen and Wilco van den Heuvel

An attempt at green lot sizing

Introduction: Sustainable OR

There has been a large deal of attention in Operations Research (OR) to sustainability issues, in particular the negative impact of carbon emissions from transportation. However, it appears that `doing something about carbon emissions' may be something that is theoretically interesting, but has little practical relevance. One of the potential reasons is that reduction of carbon emissions is often presented as an expensive option: A company can throw money at green initiatives and hope that customers are willing to buy their products at a premium price as a consequence. In reality, a company would invest in carbon emission savings if they make sense, also from a long-term financial perspective.

In our study, we evaluate how much it costs to save carbon emissions in an Economic Lot Sizing setting (introduced in the next section). We do so by constructing many instances with realistic parameter values, in particular for carbon emissions. Then, we use a bi-objective optimization approach to construct the trade-off between emissions and costs.

Economic Lot Sizing

An important sustainability issue addressed in many OR studies is the emission of carbon dioxide resulting from road freight transportation (for an overview, we refer to Dekker et al. 2012). One possibly attractive way to do so is by allowing for larger shipment sizes, which in turn makes that a vehicle can be utilized better. However, the consequence of the larger shipment size would be that more costly inventory should be held.

In our study, we assess this trade-off in a situation where in a given number of future periods demand is known and deterministic. In order to satisfy demand, we have to ship items from a supplier at another location. We currently make a plan to determine how much we should transport at the beginning of each period, i.e., we determine whether and how much to ship. Typically, there are fixed costs associated with each shipment, and there is the cost of holding an item on inventory at the end of each period. This set-up is an example of an Economic Lot Sizing (ELS) problem, a problem that is common in, e.g., process industries (Jans and DeGraeve, 2008).

Some studies in the field of ELS add a second objective of minimizing carbon emissions to the standard objective of minimizing costs. Emissions are related to the transportation and to the number of items on inventory. However, these studies tend to focus on mathematical properties of the problem, such as the conditions under which the problem is efficiently solvable, and do not attempt to estimate realistic carbon emission parameter values. Thus, the results of these studies cannot be used to determine how sensible it is to reduce carbon emissions through shipping larger quantities.

In our study, we consider the trade-off: What does it cost to reduce carbon emissions within a lot sizing setting? It is not a ‘usual’ OR study in the sense that we do not have a well-specified solvable problem from a given company, but try to determine how the trade-off between carbon emissions and costs depend on key factors (and determine what these key factors are). That obliges us to find realistic ranges of values on the emissions related to lot sizing decisions.

Emissions from Lot Sizing Decisions

In order to make a realistic trade-off, we need to know how emissions depend on the transportation and inventory decisions in ELS. This is not a straightforward counting exercise, but there are difficulties in the collection of carbon emissions. Many emission estimates, e.g., from company reports or environmental agencies, are averages. The problem is that, if a company finds that storing 5,000 units for a year gives carbon emissions of 200 kg, it is by no means so that storing 10,000 units lead to emissions of 400 kg: if these additional 5,000 units can be stored within the same space, no emission increases may occur. This becomes even more challenging when resources are shared for multiple products.

Regarding transportation by truck, the relevant factors are the vehicle type (including load capacity), the load on the vehicle, the driving conditions, and obviously, the distance driven. We have taken the results from a study that takes these factors into account, namely the Finnish organization LIPASTO. For a large 40 ton vehicle, for example, carbon emissions in an urban area are 1034.8 g/km for the empty and 1518.4 g/km for the full vehicle, and under highway conditions, the corresponding numbers are 668.2 and 907.4 g/km, respectively. Given that emissions increase linearly with the load on the vehicle, these emission figures enable us to determine total emissions for this vehicle, driving any distance with a given load on any mix of urban and highway driving conditions. We found that the results were similar to those of other studies that included these factors.

Regarding the inventory levels, we could not find any studies that specifically relate carbon emissions to the amount of inventory in a given period. This makes sense as it may not be possible to operate a warehouse specifically for a given amount that is left at the warehouse at the end of a given period. However, existing studies did relate carbon emissions to the size of the warehouse, of around 33 kg of CO_2 per m^2 (but with large variations).

The question is whether larger inventories for a given product requires us to have more warehouse space and thereby larger emissions. It could be that the warehouse space rented is fixed for a longer period; it could be that the warehouse size is given and independent of the amount stored in the ELS problem; or it could be that the product, say product A, shares a warehouse with other products. If we then store more A, a larger share of the total emissions are allocated to A, but if the warehouse size remains the same, the inventory increase has no impact on overall carbon emissions.

To summarize, the costs and carbon emissions are related to the shipments during the horizon and the inventory levels at the end of each period. Regarding emissions, some good measurements relate emissions to the vehicle type and the load on the vehicle. Measurements on emissions from inventory are scarce, and mainly relate them to the number of square or cubic meters of storage space.

The Trade-Off in a Model

We determine the trade-off between carbon emissions in numerical experiments. In these experiments, we create instances based on the data and the considerations described above. Each instance is solved as a bi-objective Mixed Integer Linear Programming (MILP) model. The decisions are the amount shipped, the number and types of vehicle used, and the amount stored in inventory. A feasible ELS solution ensures that the quantity shipped in a period + inventory from the previous period suffices to cover demand. We do not present the model, but illustrate it in Figure 1 for a three period problem.

single 100%*1900/2250=84% loaded truck. We use that there are 8.7 km of urban roads and 238 km of highways, so the reader can check that emissions are 194.05 kg (an extension in our experiments is to include the empty haul back). There should at least be storage space for 900 items; given 100 items per m2, this means that at least 9 m2 should be operated.

Our bi-objective model contains the objective Z1 representing the cost of schedule and the objective Z2 representing the carbon emissions. It can provide a set of efficient solutions, i.e., we cannot improve one objective without deteriorating the other. The set of efficient solutions (the efficient frontier) describes the trade-off between costs and carbon emissions.

There are two types of efficient solutions in a bi-objective MILP; see Figure 2. The black points are strongly supported solutions, meaning that there exists a value of α such that S is the optimal solution to the problem αZ1 + (1- α) Z2. The yellow point is an example of a weakly supported solution, where no such value of α exists.

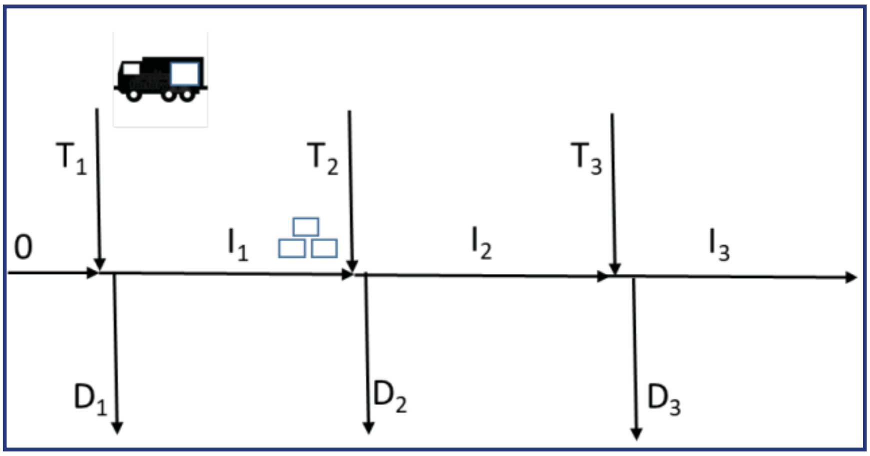


Figure 1: Economic Lot Sizing graphically

In our cost-optimizing solution of an example instance presented in Table 1 below, we serve demand in period 1 (1000 units) and 2 (900 units) with a single shipment in period 1 (T1=1900) and with 900 items in inventory at the end of period 1 (I1=900). We incur costs of one shipment and 900 items on inventory. The emissions of the shipment in period 1 are caused by a

When we place a cap on emissions, this corresponds to an upper limit that can gradually be decreased (this corresponds to the so-called Ɛ–constraint method). For both the weakly and strongly supported solutions, there exists a cap for which the solution is cost optimal. However, a cap on carbon emissions would be relatively arbitrary for management purposes. For example, in Figure 2, the slope between S2 and the yellow point is quite shallow. So if the cap lies just above the yellow point, the reduction of the last set of emissions is expensive.

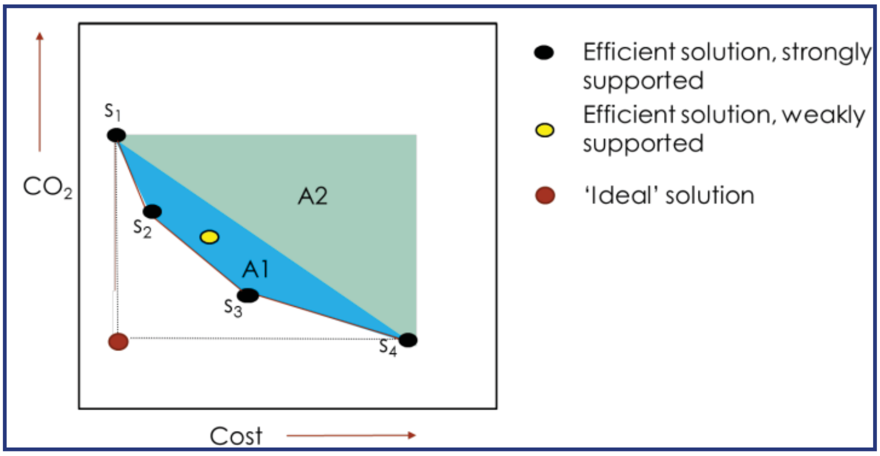


Figure 2: Representation of an efficient frontier, with 4 strongly supported solutions and a weakly supported one conomic Lot Sizing graphically

When a given price or tariff is charged on each ton of carbon emitted or saved, only the strongly supported solutions apply. These can be found using the weighted method (we do not go into details here). On the other hand, in the weighted method, assuming that Z2 measures the emissions in tons of CO2 and Z1 the costs in €, it holds that the cost of reducing emissions by a ton would be represented by (1-α)/α. We call this the shadow price of carbon emissions. In fact, this also applies to a cap-and-trade scheme, where one can buy additional trading permits or sell them if they are not needed.

A Numerical Example

We explain our experiments, main results, and insights through our base case. In our base case, there is a shipment between the Dutch cities of Rotterdam and Groningen. Demand and shipments take place on a monthly basis over a period of a year so there are 12 periods' (note that the notation T is already used for transportation) and one large vehicle with gross weight of 40 ton and the capacity is C1=2250. Only one product type is considered. Demand quantities and further model parameters are given in Block 1.

There are three supported strongly supported solutions with, in increasing order of costs, (Z1; Z2) = (5541.67; 2055.16), (5666.67; 1920.16), (6645.83; 1851.16), respectively. In these solutions, we use 8, 7, and 6 shipments, respectively. The shipment schedules are given in Table 1.

Period	Demand	Solution 1 T	I	Solution 2 T	I	Solution 3 T	I
1	1000	1900	900	1900	900	2050	1050
2	900	0	0	0	0	0	0
3	1000	2100	1100	2150	1150	2250	1250
4	1100	0	0	0	0	0	0
5	1200	1200	0	2250	1050	2250	1050
6	1100	2100	1000	0	0	0	0
7	1000	0	0	1800	800	2250	1250
8	800	1500	700	0	0	0	0
9	700	0	0	1900	1200	0	0
10	1200	1200	0	0	0	1750	550
11	1300	1300	0	1300	0	2250	1500
12	1500	1500	0	1500	0	0	0

Table 1: Properties of the three efficient solutions in the numerical example

It costs €5666.67-€5441.67 = €125 to reduce 2055.16 – 1920.16 kg = 135 kg carbon emissions, giving a first shadow price of €925.95 per ton. The shadow price of carbon increases strongly, to €14,191 per ton, when going from solution 2 to 3. Compared to the EU Emissions Trading Scheme price of around €10 or the carbon emissions damage estimate of between $40 and $80, this shadow price is very high.

In this base case, we observe that: 1) There are only few solutions in the efficient frontier; 2) The carbon emission reduction potential is only 9.93%; and 3) Total emissions in kg are only a third as large as costs in €, which means that a very strong curve is needed to achieve a low shadow price.

In our base case, the period length of a month is long, so it is polluting and relatively costly to store items. Transport costs are set to the standard freight tariff of €500, which makes it expensive to reduce the number of shipments. More shipments options are available when a second, smaller vehicle type is available. This could increase the number of strongly supported solutions. Finally, demand is about half the vehicle capacity; Lower demand quantities could also result in a larger emission reduction potential. We have tested these in further set of experiments with more than 800 instances. We do not go into detail but report the main insights.

Can emission reduction costs be reasonable?

In a large set of experiments, we have varied parameter values in our ELS model: emissions, costs, the number of periods, product characteristics, and so on. The purpose is to determine in which cases carbon emissions reductions can be achieved at reasonable costs. We find that there are instances for which this is so if:

Demand in 12 periods:

1000; 900; 1000; 1100; 1200; 1100; 1000; 800; 700; 1200; 1300; 1500.

Transport:

Cost: €500 per vehicle making a shipment.

Emissions: 168 kg (empty truck), 229.2 kg (full truck); emissions linear in the load on the vehicle.

Warehouse:

Cost: inventory holding cost of €0.42 per item per period.

Emissions: $0.33s$ per year, where $s \geq I_t$ (inventory in each period t) for each $t=1,\dots,12$.

Block 1: The data in the numerical example

1) We increase the number of periods and reduce the period length, so concretely, set the decision period to two weeks and there are 26 weeks in total. (We tried the 52 week case as well, but it turns out that CPlex has large difficulties with such instances. One of our future research directions is to improve the LP formulation or derive new exact methods.)

2) The transport costs decrease, in our case to €150. That would mean that vehicle and driver related costs are mainly sunk, and only the marginal fuel related costs of driving are considered.

3) The emissions of the (empty) return haul are included. It makes that more emissions are saved when the number of shipments are reduced.

Some other factors tend to decrease the price of emission reductions as well, such as the presence of a second type vehicle and no warehouse emissions.

There is one big \`but'. In reducing the shadow price, we align the objective functions Z1 and Z2. If we go too far, it means that the cost-optimal gets closer to the carbon-optimal one. In Figure 2, for example, solution S1 may no longer be cost-optimal, but may lose its status to S2. The cost of any further reduction increases, paradoxically, to one that roughly corresponds to the slope of the line segment from S2 to S3.

Conclusions

We started by arguing that carbon emission reductions are not a luxury that should be pursued at all costs, but it is a financial decision that many companies take. We consider emission reductions from transportation, through larger shipment quantities, in a specific situation. Demand quantities at the destination are known and the costs related to shipments and inventory are fixed. In that case, we find that it is often quite costly to reduce carbon emission, even a small amount.

One objection is that we, as many other studies in green OR, solely focus on carbon emissions. It is well-known that transport has other negative externalities, such as local pollution and congestion. Reducing the number of shipments would lead to a reduction in these externalities as well. A direction for future research is to take this into account.

Our analysis can be extended to more general shipment decisions, both with stochastic demand or with delivery routes instead of direct shipments.

Reference:

Jans, R., Degraeve, Z., 2008. Modeling industrial lot sizing problems: a review. International Journal of Production Research 46 (6), 1619-1643.

Marcel Turkensteen works as associate professor at the Cluster for Operations Research, Analytics, And Logistics (CORAL) at the Faculty of Business and Social Sciences, Aarhus University. His research interests include the combination of environmental impact measurement and Operations research and logistics, decision modeling on routing and location.

Wilco van den Heuvel is an Associate Professor of Operations Research at the Econometric Institute, Erasmus University Rotterdam. His research interests include developments of operations research techniques with applications in production planning, supply chain coordination and cooperative game theory.

By Michael Patriksson

M'n'M: Morons and Mendeley – a citation analysis gone slightly too far

Abstract: Is there a gain in adding cuss words in order to get cited more frequently, and, if so, which cuss words work the best? That is the research question in focus in this paper. We define a measure, or score, called *impact*, which is the quotient between the number of citations to a document, and the frequency of the use of a cuss word in it; the assumption is that the right cuss word in a paper enhances it, while others don't. The outcome of this analysis is a short-list of cuss-words that – from the analysis – aid the most in the striving for many citations.

The motive

A main goal for an author is to be cited. In order to maximize citations, authors may use a variety of tools. One tool is based on citing oneself; while it is effective, it is also frowned upon, if readers find that the self-citations are rather too frequent, or even irrelevant. Another tool is to make sure that the article causes a stir, provoking its reading through curiosity. The "click-bait" that is discussed in this article is the utilization of cuss words in the text, or in the article title. We analyze the strength of cuss words, through a citation analysis; the simple tool of counting citations is the basis of our analysis. We are well aware that other factors lie behind the citations, but we blatantly ignore them, as we are sure that all our colleagues also do. Here we go.

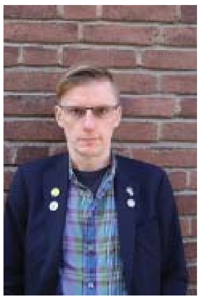
The analysis

We utilize the article database at Mendeley, in order to collect, for each cuss-word chosen, the impact of a cuss-word. Below we list – in increasing order of impact – the cuss-words studied:

In Mendeley the word "weirdo" is mentioned in 15 documents, having 5 citations; hence an impact of 5/15=1/3. The search term "idiot" is mentioned in 8573 documents, having 11351 citations; hence an impact of 11351/8573=1.32. The search term "fuck" is mentioned in 2116 documents, having 457 citations; hence, we conclude that the impact of "fuck" is 2116/457=4.63.[1] The search term "bastard" is mentioned in 1635 documents, having 14,005 citations; hence an impact of 14,005/1635=8.57. The search term "moron" was unfortunately scrapped, as there are several authors who bear that surname. Finally, Mendeley has 50 articles in which the word "asshole" is mentioned at least once, in the title or in the text. The total number of citations for these articles is 470. So, by definition, the impact of the word "asshole" is 470/50=9.4. The asshole is hence the winner; surprised, anyone?

Conclusion and future research

The purpose of this article is to gain citations – isn't it obvious?? In order to achieve this goal, we have tried the complete opposite to what "normal" scientists would: rather than citing the obvious old and boring sources, and develop some new theory or practice that no-one will understand, or bother to read five years later, we strive to offend, cause a stir, even get sued – anything to get immediate attention. This is publishing, Donald Trump style: this is click-bait, the instant research fix, the new new deal – haven't you paid attention during the last ten years? We guess that you, who read this, is a scholar. Well, if you're offended – get another job, moron! [2]



Michael Patriksson is a professor of applied mathematics at the Department of Mathematical Sciences, Chalmers University of Technology and the University of Gothenburg, SE-412 96 Göteborg, email: mipat@chalmers.se

1 In the database there is also mentioned a journal by the name of Journal of I Don't Give A Fuck. That journal does, however, not appear to exist in the real world.

2 We have been alerted that the editor of this esteemed journal has been deeply offended. We beg forgiveness! In future research we shall strive to be much more politically correct. We acknowledge the great help from the team at Orbit – they are simply wonderful people – some of them are even my personal friends!

By Julia Pahl and Brooks Kaiser

Port and Service Development in the Arctic

This article is an extraction of the book chapter of J. Pahl and B. Kaiser "Arctic Port Development" in: Arctic Resource Governance and Development, N. Vestergaard, B. Kaiser, L. Fernandez and J. Nymand Larsen (editors), Springer Polar Sciences, Pages 139-184, 2018

Introduction

Melting sea and shore ice open up new options in maritime shipping. Two routes are particularly interesting to stakeholders. One is the *Northern Sea Route* (NSR), which is particularly viable as an option for containerized commodity flows between Asia and Europe, permitting a shortening of sailing distances up to 50% from, e.g., Yokohama to Rotterdam. This route is also very attractive in light of more and more congested routes such as the Suez Channel. The other route is the *Northwest Passage* (NWP), which seems less attractive for containerized shipping due to more challenging navigation requirements and less infrastructure supplies, although it is increasingly of interest to the tourism and fishing industries which are extending themselves into new "exciting" waters (see Pahl et al. (2017) and the references therein).

The more attractive these routes are in the future, the more requirements and thus challenges rise with respect to port infrastructure. This includes services from providing supplies to search and rescue (SAR) for ships traversing Arctic waters. Questions on how stakeholders in the Arctic can prepare themselves for increasing maritime activities are sparking increasingly exciting and stimulating research on: "Will Arctic ice disappear to such an extent that the Arctic becomes an option for liner shipping services?" to "How can the sensible ecological and sociological environment of the Arctic be protected from future rising maritime activities?"

Arctic Main Drivers

The authors investigated, among other questions, how Arctic ports can serve to mitigate concerns regarding the passage of the NWP or NSR with respect to safe and secure traversing as well as what will drive their future development in terms of economic viability and environmental requirements for the stakeholders (see Pahl et al. (2017)). The 2009 Arctic Marine shipping assessment report (see Arctic Council (2009)) emphasizes three fundamental areas that need to be accounted for when discussing and evaluating Arctic maritime activities: 1) Arctic maritime safety, 2) Arctic people and their environment, and 3) Arctic marine infrastructure. The latter includes improved navigational aids in order to render navigation in Arctic waters more safe, e.g., vessel tracking, traffic separation, identification of areas with special concern etc. When it comes to predictions regarding the future usage of Arctic waters, "voices" recorded in the media and academic literature could not be more opposite: some say that the likelihood is high that the ice will disappear, so that year-long maritime transit will be possible and others that shore-based routes are unlikely to serve as consistent high-volume cargo shipping pathways due to economically and technical reasons – not to mention relatively inhabited and geographically challenging territory (see Hansen et al. (2016)).

Arctic Infrastructure

When analyzing the available operating and planned Arctic maritime infrastructure, we find numerous very small ports with entry ice concerns, some small ports, but a very low number of medium sized and just one large sea port in Russia (Murmansk); see Table 1. The demand of different types of services requiring infrastructure comes along with several maritime activities, such as oil and gas extraction, importing and exporting commodities, and traversing Arctic waters with containers or tourists. Projects regarding Arctic port infrastructure exist, e.g., extending the port of Iqaluit in Canada to a deep-water port with berthing and docking facilities for cargo handling. Norway is currently the best prepared player to accommodate maritime activities. It has a comparably great number of large and medium sized ports in the Arctic including one with an airport. Iceland has extensive extension plans for its port infrastructures, including a new deep-water port in the north east of the island which should serve as a transshipment hub for containers coming from the Arctic to eastern North America and Europe, a role currently performed by the Russian Port of Murmansk. Iceland is also counting on tourism attracting the cruise business. Together with Norway, Russia is the greatest stakeholder of the NSR in terms of vicinity. In fact, the Russian development plan has the objective to provide safe and reliable navigation for ships on the NSR until 2030.

When analyzing existing and planned port infrastructure, it might be misleading to draw attention especially to ports with depths that can accommodate heavy loaded ships, e.g., Panamax ships, requiring at least a 30 feet water depth (ca. 9.1 meters) that can be interesting for bulk and container shipping operations. Yet, this may become a self-fulfilling prophecy: high-volume transit will not come to the Arctic without appropriate depth clearances of ports. The same hen-and-egg problem is valid for enhancing infrastructure especially with respect to SAR service by coast guards: private investment lowers if safety and security for maritime activities cannot be assured. As a result, there is a need for a detailed, structured analysis of the transport and logistic system of the NSR (and the NWP) including SAR as well as navigation and communication systems, which are also very important for insurance companies to perform risk assessments and allow for insurance coverage.

Figure 1: Operating and planned deep, large, and medium sized ports in the Arctic also including airports on the NWP and NSR (the latter are inserted without navigational accurarcy)

Country	Large	Medium	Small	Very Small	Total Ports	Percent of Ports in the Arctic
Canada	0 (4)	1 (14)	3 (72)	14 (196)	**18 (286)**	6,3%
Greenland	0 (0)	0 (0)	8 (8)	13 (16)	**21 (24)**	90,0%
Iceland	0 (0)	0 (2)	0 (2)	13 (24)	**13 (28)**	46,4%
Norway	0 (1)	3 (10)	9 (34)	35 (90)	**47 (135)**	34,8%
Russian Federation	1 (4)	2 (5)	9 (21)	12 (44)	**24 (74)**	32,4%
United States	0 (21)	0 (38)	0 (132)	9 (475)	**9 (666)**	1,4%
International (Svalbard)	0	0	0	3	**3**	
Total Arctic Ports	**1**	**6**	**29**	**96**	**132**	

Table 1: Number of Arctic national ports in size

Conclusion

The analysis shows that there is a lack of maritime infrastructure to provide the needed service in terms of suppliers and SAR for Arctic maritime shipping on the NWP and NSR. But desipte the contractictory statements regarding Arctic maritime activities, companies express a great interest in these new opportunities that open up with the deicing of Arctic Waters.

References

[1] J. Pahl and B. Kaiser, Arctic Port Development, Book Chapter in: Arctic Resource Governance and Development, N. Vestergaard, B. Kaiser, L. Fernandez and J. Nymand Larsen, Pages 139-184, 2018

[2] C.Ø. Hansen, P. Grønsedt, C.L. Graversen, C. Henriksen, Arctic Shipping, CBS Maritime, ISBN 978-87-93262-03-4, 2016

[3] Arctic Council Assessment Report 2009, available at: http://www.pmel.noaa.gov/arctic-zone/detect/documents/AMSA_2009_Report_2nd_print.pdf, 2009, last access: 02.04.2019

Julia Pahl is an associate professor at the University of Southern Denmark. Her research is in operations research for (maritime) logistics.

Brooks Kaiser is a professor mso at the University of Southern Denmark at the management and economics of resources and envorinment (MERE).

HICSS

SHIDLER COLLEGE OF BUSINESS

January 7-10, 2020 | Grand Wailea, Maui

Participate at the the *Decision Analytics, Mobile Services, and Service Science Track!*
Visit us at: http://hicss.hawaii.edu/tracks-53/

TRACK CHAIRS

Christer Carlsson

Institute for Advanced Management Systems Research
Abo Akademi University
Auriga Business Center
20100 Turku, Finland
Tel. +358 400 520 346
christer.carlsson@abo.fi

Haluk Demirkan

Milgard School of Business
Center for Business Analytics
University of Washington – Tacoma
1900 Commerce Street, Box 358420
Tacoma, WA 98402-3100
Tel: +1-253-692-5751
haluk@uw.edu