

# Discrete Optimization Methods in Maritime and Road-based Transportation

# DOMinant

Forskningsprosjekt med støtte  
fra Norges Forskningsråd

## Samarbeidspartnere



Norges Teknisk-Naturvitenskapelige Universitet  
Institutt for industriell økonomi og teknologiledelse  
Trondheim



Høgskolen i Molde  
Optimeringsgruppa  
Molde



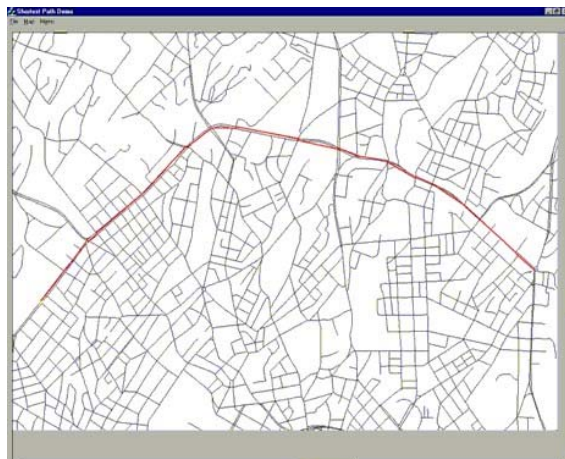
SINTEF  
Avdeling for anvendt matematikk  
Oslo

<http://www.iot.ntnu.no/forskning/forskerprosjekt/dominant/>

# TRANSPORTOPTIMERING

## Optimal transport

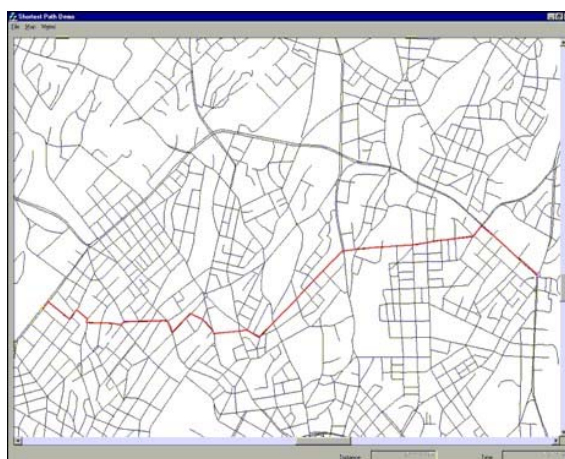
Det finnes mange vanskelige planleggingsoppgaver innen transport. Bedre koordinering av transportvirksomhet vil gi bedre kundeservice, bedre ressursutnyttelse, mindre unødig kjøring og derved store miljømessige og økonomiske besparelser.



## Transportplanlegging kan gjøres både på lang og kort sikt.

### Typiske eksempler er:

- Design av transportnettverk
- Dimensjonering av en flåte av kjøretøy
- Dynamisk flåtestyring
- Allokering av oppdrag til kjøretøy/turer
- Sekvensiering av besøk i en tur
- Korteste vei mellom punkter i et veinettverk
- Optimalisering av leveringstidspunkt
- Dynamisk vedlikehold av plan ved endringer



Figurene viser henholdsvis *raskeste* og *korteste* vei mellom Frognerparken og Carl Berners Plass i Oslo. Som forventet er raskeste vei å velge Ringvei 2. Korteste vei viser seg å gå gjennom Uranienborg, St.Hanshaugen og Dælenenga. De optimale rutene er beregnet av optimeringsprogrammet **SPIDER** som er utviklet av SINTEF.

# DOMinant

## Målsetning

*Utvikle og forbedre metoder for å løse tunge diskrete optimeringsproblemer i maritim og veibasert transport.*

**Spesielt fokus på to typer ruteplanleggingsproblemer:**

### **Fleet Size and Mix Vehicle Routing Problem (FSMVRP)**

*En type ruteplanleggingsproblem der man i tillegg til å finne den best mulige kjøreplanen også skal finne den best mulige sammensetningen av en flåte med ulike typer kjøretøy.*

### **Inventory Routing Problem (IRP)**

*En type ruteplanleggingsproblem der frakten ikke bestemmes av kundeordrer, men heller behovet for gjentatt å transportere varer til kundens lager. Kunden forbruker varene i et bestemt tempo, og rutene må lages på en slik måte at varelageret ikke blir tomt.*

# RUTEPLANLEGGING

## er det vanskelig?

*For mange ruteplanleggingsproblemer kan vi ikke garantere å finne en optimal løsning for problem av interessant størrelse innen rimelig tid, selv med en superrask datamaskin.*

### *Eksakte metoder:*

*Metoder der man sjekker alle mulige løsninger eller eliminerer bort deler av løsningsrommet som ikke kan inneholde den beste løsningen. På denne måten vil man finne den optimale løsningen, men dette kan kun benyttes på små problem.*

### *Øvre/nedre grense:*

*Vi kan løse et forenklet problem eksakt for å finne en grenseverdi. Det vil si, vi vet med sikkerhet at den optimale løsningen er ikke bedre enn grenseverdien.*

### *Hybride metoder:*

*Metoder som kombinerer to eller flere av metodene nevnt over.*

### *Tilnæringsmetoder/*

### *Heuristikker:*

*Noen smarte regler som normalt fører til en god løsning på kort tid, selv om man ikke kan garantere kvaliteten på løsningen.*

### *Metaheuristikker:*

*Heuristikker som benytter en form for læring. Dette gjør at de kan utforske større deler av søkerommet uten at de vender tilbake til løsninger som tidligere er sjekket. Eksempel på metaheuristikker er Tabu Søk, Genetiske Algoritmer, Simulert Herding og Maur Algoritmer.*

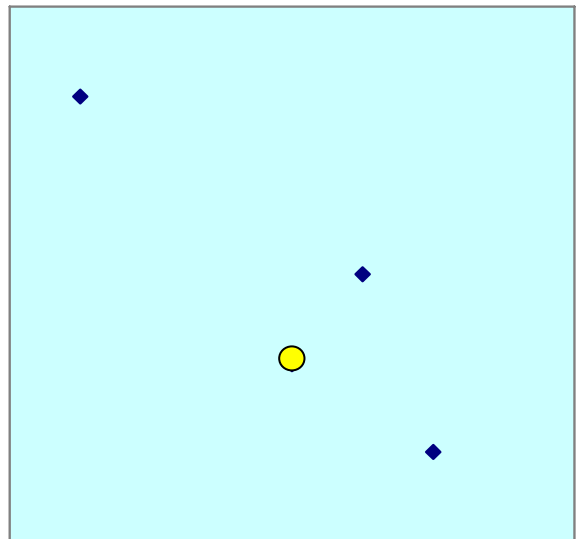
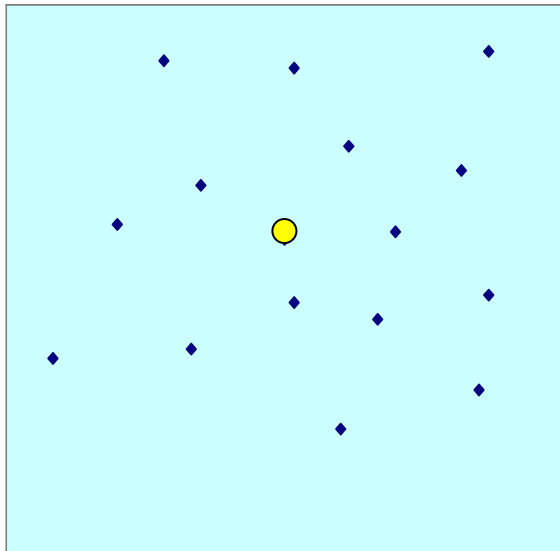
*Optimeringsprogrammet SPIDER benytter slike metaheuristikker.*

*... det er vanskelig, men vi får det nok til!*

# PRØV SELV ...

Problemet:

Finn den korteste ruten som starter i depotet, besøker alle kundene og returnerer til depotet.



# KONKURRANSE

## Problemet:

Finn den korteste ruten som starter i depotet, besøker alle kundene og returnerer til depotet.

Avstanden mellom hvert punkt er gitt ved den Euklidske distansen, dvs. den direkte linjen mellom punktene.

Premier til den/de som finner den korteste ruten.


Et slikt problem med  $n$  kunder har  $n!$  ( $n$  fakultet) mulige løsninger ( $n \cdot n-1 \cdot n-2 \cdot \dots \cdot 3 \cdot 2 \cdot 1$ ).

## Eksempelvis

100 kunder:  $9 \cdot 10^{157}$   
mulige løsninger

50 kunder:  $3 \cdot 10^{64}$  mulige løsninger

10 kunder: 3 628 800 mulige løsninger

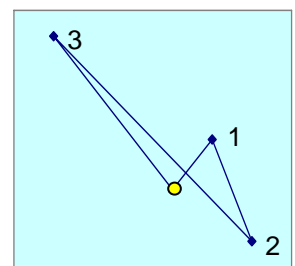
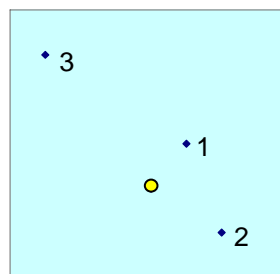

$$3 \cdot 10^{64} = \overbrace{30.000.000.000.000.000.000.000.000}^{64 \text{ nuller}} \\ .000.000.000.000.000.000.000.000 \\ .000.000.000.000.000.000.000.000$$

# HEURISTIKKER

## hvordan virker de?

En "grådig" heuristikk går ut på å konstruere en rute ved alltid å velge det nærmeste punktet i grafen. Merk at denne strategien ikke finner den optimale løsningen for denne instansen.

Eksempel med tre kunder.  
Løsning funnet med grådig heuristikk.

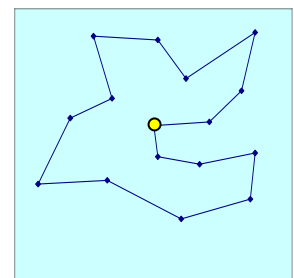
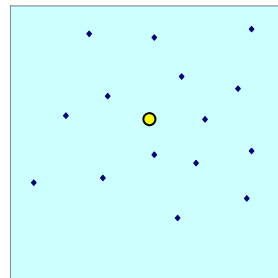


Tabu Søk er en metaheuristikk som ved hjelp av små endringer på en løsning gradvis prøver å forbedre denne.

Dette kan for eksempel være å bytte rekkefølge på to kunder i turen.

For å unngå at vi vender tilbake til en løsning som tidligere er prøvd, vil de endringene vi utfører ikke kunne endres tilbake igjen før det har gått en viss tid.

Eksempel med 15 kunder.  
Løsning funnet med Tabu Søk



Eksempel med 261 kunder.  
Løsning funnet med Tabu Søk

