

# Automatisk planlegging i oljeindustrien

av

Teis Lindemark

**AVHANDLING**

*for en grad av*

**MASTER OF SCIENCE**

*Master's Avhandling, Institutt for informatikk*



*Fakultetet for teknologi og realfag  
Universitetet i Bergen*

*Juni 2012*

*Fakultetet for teknologi og realfag  
Universitetet i Bergen*

## **Sammendrag**

TEST

# Innhold

<b>1</b>	<b>Forord</b>	<b>4</b>
<b>2</b>	<b>Introduksjon</b>	<b>6</b>
2.1	Bakgrunn . . . . .	6
2.2	Målet med prosjektet . . . . .	6
2.3	Motivasjon . . . . .	7
2.3.1	Kort om begrensingsprogrammering . . . . .	7
2.3.2	Kort om ILOG . . . . .	7
2.3.3	Utfordringer med begrensingsprogrammering . . . . .	7
2.3.4	Begrensingsprogrammeringsverktøy idag . . . . .	7
2.4	Problembeskrivelse . . . . .	8
<b>3</b>	<b>NEW METHODS</b>	<b>9</b>
<b>4</b>	<b>Eksperimenter</b>	<b>10</b>
<b>5</b>	<b>Fremtidig arbeid</b>	<b>11</b>
<b>6</b>	<b>Konklusjon</b>	<b>12</b>

# 1 Forord

# Nomenclature

CP      Constraint programming

## 2 Introduksjon

I dette prosjektet vil utvikleren bli gitt et eksisterende ILOG Scheduler program og benchmarksett. Det eksisterende ILOG Scheduler programmet skal utvides med flere ressurser og benchmarksettene skal kjøres for å kunne sammenligne løsningene med de nye resurssene mot de løsningene uten de nye resurssene.

### 2.1 Bakgrunn

Begrensningsprogrammering er en programmeringsparadigme hvor relasjoner mellom variable blir satt i form av begrensninger. Begrensninger er en form for deklarativ programmering, som skiller seg fra den mer vanlige imperativ programmeringsspråk<sup>1</sup> ved at løsningen blir til ved å tilfredstille begrensningene. Det er forskjellige områder i begrensningsprogrammering som "Constraint Satisfaction problems" og planleggingsproblemer. Det mest kjente planleggingsproblemet er "Job Shop Scheduling".[1]

I dette prosjektet, verktøyene som er brukt for å utføre eksperimenter på emnet er ILOG Scheduler som er endel av IBM sitt ILOG CP. ILOG Scheduler er et C++ bibliotek som gjør det mulig å definere planleggingsbegrensninger i form av ressurser og aktiviteter. Planlegging er en prosess ved å tildele ressurser til aktiviteter og tildele en tid til aktivitetene så det ikke er noen konflikt med begrensningene.[3]

Det overordnede målet med denne avhandlingen er å utvide ILOG Scheduler løsningen til Bård Henning Tvedt med flere ressurser for å sjekke om det å legge til flere ressurser vil gi bedre og flere løsninger enn de opprinnelige løsningene til Bård Henning Tvedt .

### 2.2 Målet med prosjektet

Målet med prosjektet er todelt, og består i å vurdere den modifiserte problemstillingen mot den opprinnelige i forhold til:

- antall begrensninger
- implementasjon i ILOG Scheduler

I den opprinnelige problemstillingen vil noen aktiviteter være relativt lite begrenset. Dette gjør at løsningsrommet er stort, og traverseringen opp og ned i søketreet tar lang tid.

---

<sup>1</sup>Imperativ programmeringsspråk har sekvenser med som blir utført.

På tross av et antatt stort løsningsrom så sliter den ILOG Scheduler implementerte løsningsstrategien med å finne løsninger i mange av probleminstansene.

- Vil flere begrensninger gjøre det lettere å finne en løsning?
- Er det noe spesielt med akkurat disse instansene eller er det implementasjonen i ILOG Scheduler som er årsaken?

## 2.3 Motivasjon

Forskningen er motivert av praktisk erfaring at planleggingsproblemer er veldig aktuelt i bedrifter og i samfunnet idag. Det er ikke bare i oljeindustrien som jeg fokuserer oppgaven på hvor planleggingsløsninger er aktuelt, men generelt bemanningsproblematikken som alle personalavdelinger sitter med i det daglige. I det hverdagen er det også mange planleggingsproblemer fra buss- og togtabeller til personlige gjøremål med å få tid til alt man skal ha tid til.

Planlegging i oljeindustrien er viktig for å på en mest mulig effektiv måte benytte seg av de ressursene som er tilgjengelig til enhver tid, samtidig som visse begrensninger blir fastsatt med tanke på sikkerheten. Det er mye penger involvert i olje- og gassindustrien og det å utføre aktiviteter på en ineffektiv måte kan koste selskapene veldig mye penger. Det er derfor viktig å ha gode løsninger for å ta seg av planleggingen av aktivitetene og ressursene.

### 2.3.1 Kort om begrensningsprogrammering

nn

### 2.3.2 Kort om ILOG

ILOG[2]

### 2.3.3 Utfordringer med begrensingsprogrammering

Challenges

### 2.3.4 Begrensingsprogrammeringsverktøy idag

Det finnes idag flere forskjellige verktøy for begrensingsprogrammering, både i form av egne programmeringsspråk som er skreddersydd for begrensingspro-

grammering og biblioteker til godt kjente programmeringsspråk som Java<sup>2</sup> og C++<sup>3</sup>. I begge disse kategoriene så finnes det løsninger som er kommersielle og med åpen kildekode. Noen eksempler på egne programmeringsspråk for begrensingsprogrammering er Prolog og Comet<sup>4</sup>. Sistnevnte er et programmeringsspråk for begrensingsprogrammering med lokalt søk og er en kommersiell løsning. Eksempler på begrensingsprogrammeringsbibliotek så er det IBM ILOG CP.

Dette er ikke løsninger som passer uansett hva slags del innenfor begrensingsprogrammering du skal gjøre. Logisk programløsning eller planleggingsløsning må tas til med når det skal bestemmes hvilke verktøy som brukes.

## 2.4 Problembeskrivelse

Problemstillingen tar utgangspunkt i den opprinnelige problemstillingen til Bård Henning Tvedt, men er utvidet med ekstra ressurser på beligenhet. Hvert mannskap avgir en varme og hver lokasjon har en gitt varmekapasitet. Summen av varme kan ikke overstige varmekapasiteten.

$$\forall t, l : \sum \{c_{heat}(Crew_j) \mid t \in [v_{sta}(Act_i), w_{end}(Act_i)) \wedge c_{crew}(Act_i) = Crew_j \wedge c_{loc}(Act_i) = Loc_l]\} \leq c_{heatcap}(Loc_l)$$

I tillegg kan en beligenhet ha begrensninger på hvor mange av et gitt mannskap, som kan arbeide på en lokasjon samtidig.

$$\forall t, l : \#\{Crew_j \mid t \in [v_{sta}(Act_i), w_{end}(Act_i)) \wedge c_{crew}(Act_i) = Crew_j]\} \wedge c_{crewlimit}(Loc_l) = Crew_j \leq c_{crewcapacity}(Loc_l)$$

Antallet beligheter er redusert fra 25 i de opprinnelige problemene til 10 i de modifiserte. Med aktiviteter spredt utover 25 beligheter, så ville det vært så få aktiviteter på hver beligenhet at varmekapasiteten til en lokasjon aldri ville blitt oversteget. Det er ikke sjekket om en reduksjon til 10 beligheter er tilstrekkelig.

---

<sup>2</sup>Java

<sup>3</sup>C++

<sup>4</sup>Comet



### 3 NEW METHODS

## 4 Eksperimenter

## 5 Fremtidig arbeid

## 6 Konklusjon

## Referanser

- [1] Wikipedia. [http://en.wikipedia.org/wiki/Constraint\\_programming](http://en.wikipedia.org/wiki/Constraint_programming).
- [2] IBM. *IBM ILOG Solver V6.8*. IBM.
- [3] Claude Le Pape. Implementation of resource constraints in ilog schedule: A library for the development of constraint-based scheduling systems. *Intelligent Systems Engineering*, 3:55–66, 1994.