HØGSKOLEN I ÅLESUND

Ingeniør

GENETIC ALGORITHMS: A REAL-WORLD APPLICATION

Date: Friday 20 February 2015

Course: Functional Programming and Intelligent Algorithms

Lecturer: Robin T. Bye

A receding horizon genetic algorithm (RHGA) for dynamic resource allocation: A case study on optimal positioning of tugs

Introduction

- Challenge: How to simultaneously
  i. coordinate control of resources;
  ii. assign tasks; and
  iii. track multiple targets

  in a dynamically changing environment?

Introduction

- Target assignment/resource allocation:
  - which agent (resource) shall track which target(s)?
- Collective tracking/positioning:
  - how should agents move to increase net tracking performance or minimise cost?
- Tracking performance:
  - how to define a cost measure?

Introduction

- Dynamic environment:
  - how can agents respond to
    - targets changing their trajectories?
    - new targets appearing and/or targets disappearing?
    - variable external conditions?

Case study: Positioning of tugs

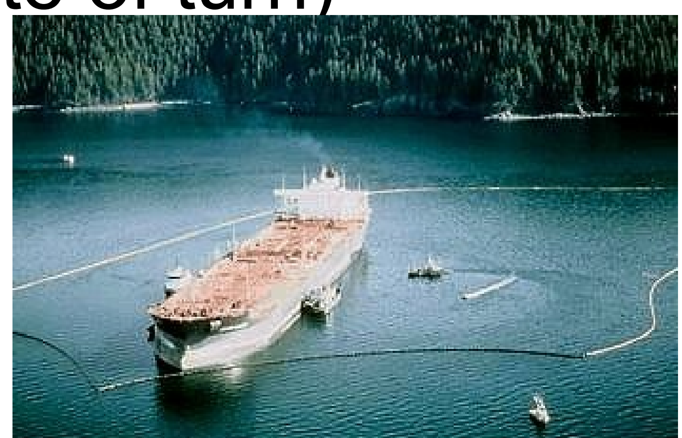
- Norwegian Coastal Administration (NCA)
  - runs a Vessel Traffic Services (VTS) centre in Vardø
  - monitors ship traffic off northern Norwegian coast with the automatic identification system (AIS)
  - commands a fleet of patrolling tug vessels
  - Mainly human control – a decision support system based on risk and statistics is implemented but with limited usability

Case study: Positioning of tugs

- Patrolling tug vessels (=“agents”)
  - must stop drifting oil tankers (=“targets”) or other ships and tow them to safety before grounding
  - are instructed by NCA to move to “good” positions that (hopefully) reduce the risk of drift grounding accidents

Automatic identification system (AIS)

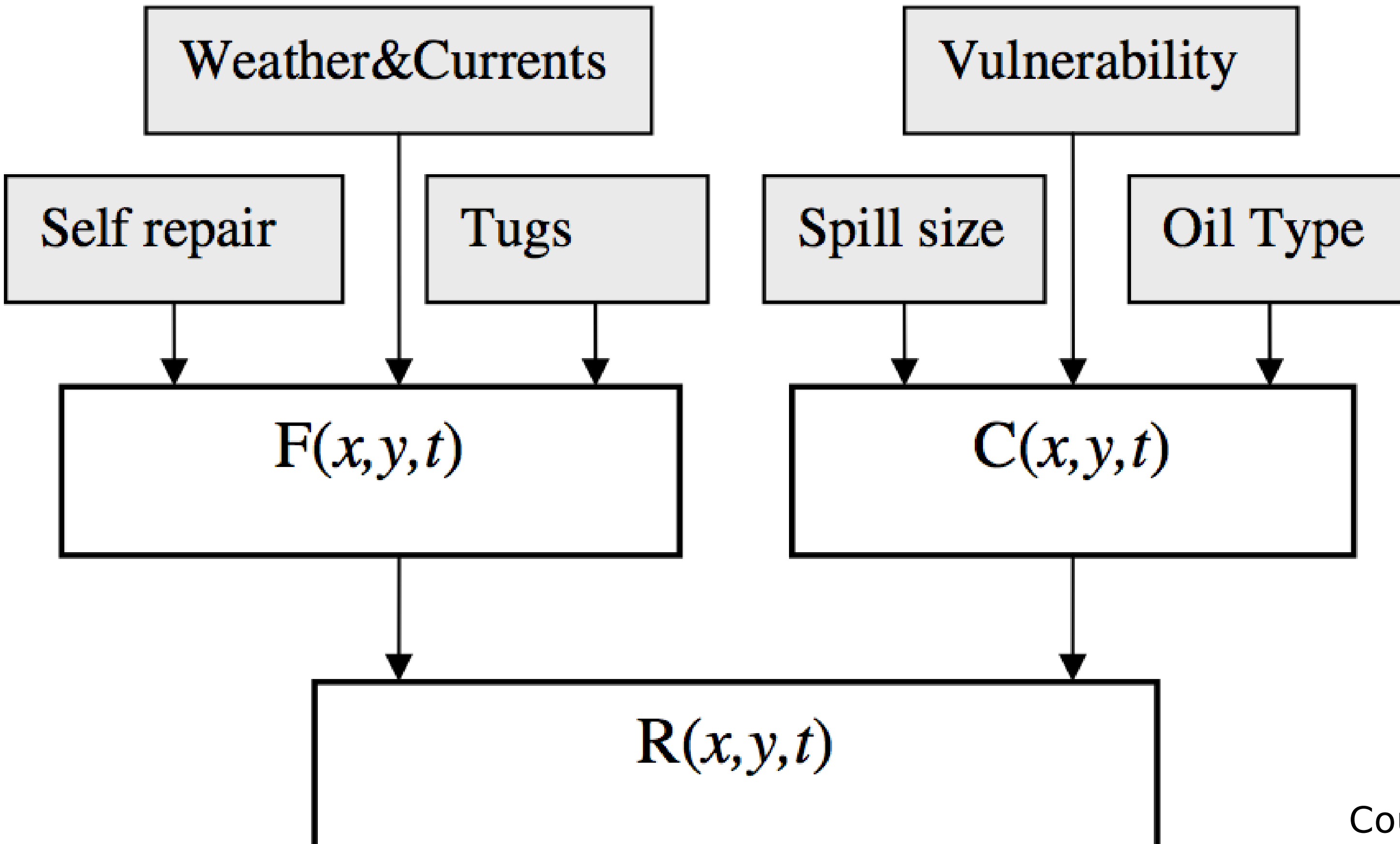
- Ships required to use AIS by law
- Real-time VHF radio transmission to VTS centres
- Static info: ID, destination, cargo, size, etc.
- Dynamic info: Speed, position, heading, etc.
- Enables prediction of future state of ships (e.g., position, speed, rate of turn)

Dynamical risk models of NCA

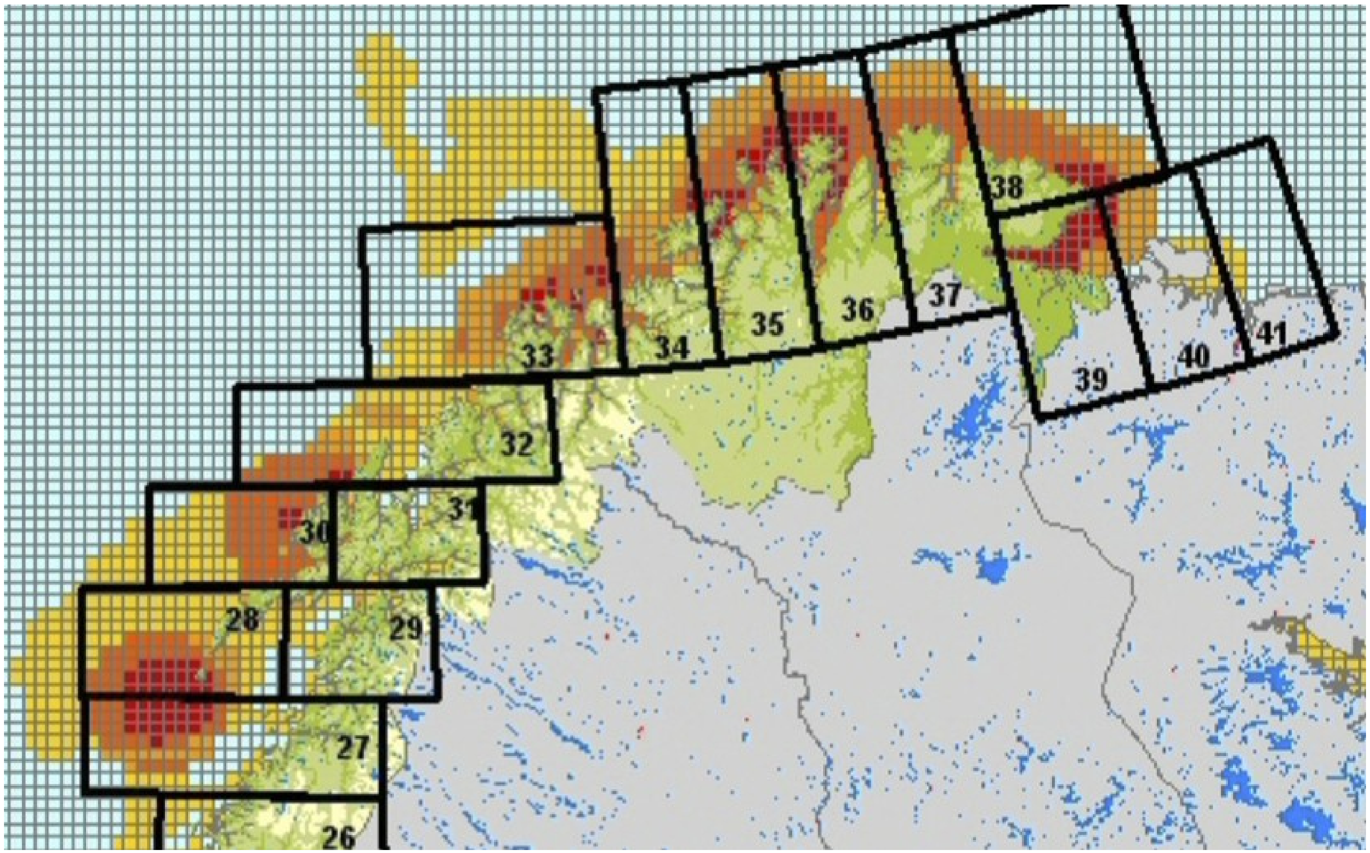
- Risk-based decision support tools
- Based on static information
  - type of ships, cargo, crew, nationality, etc.
  - geography, e.g., known dangerous waters
- … and dynamic information
  - Ships' position, direction, speed, etc.
  - weather conditions, e.g., wind, currents, waves, etc.
- Employs statistical models – focus on mean and variance from history → what about current and predicted dynamics?

Dynamical risk models of NCA



Courtesy
NCA

Dynamical risk models of NCA



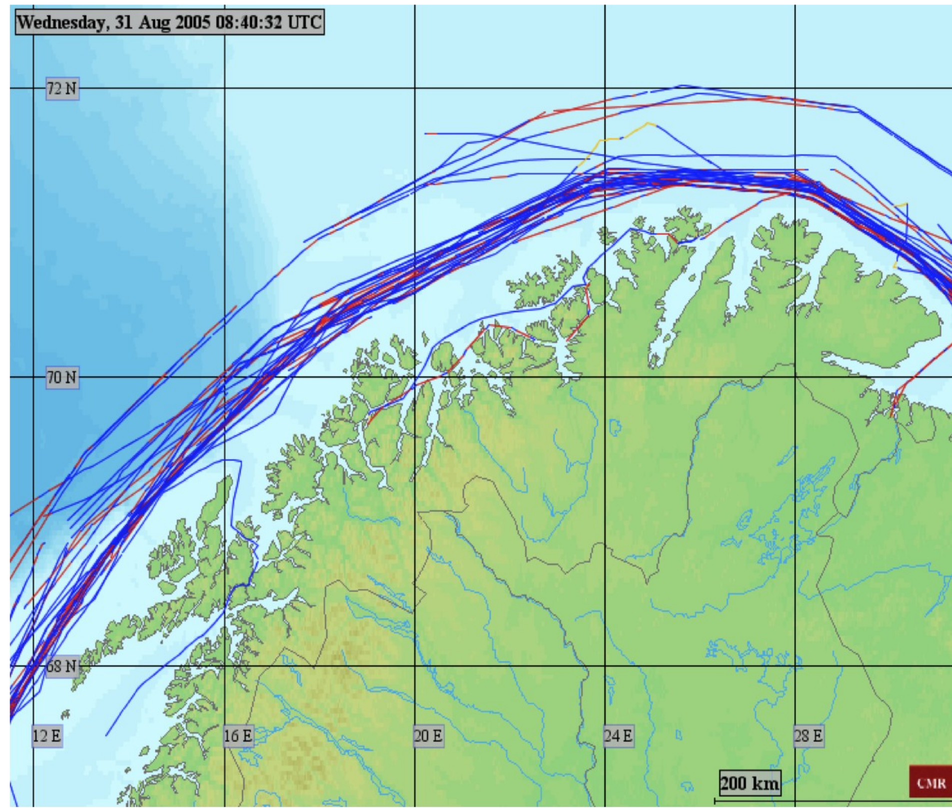
Courtesy NCA

Motivation

- Today: Human operator makes decisions based on dynamical risk models
- Limitation: Requires small number of tankers and tugs to be manageable by human operator
- Oil/gas development in northern waters will increase traffic in years to come →
  How should a fleet of tugs move to reduce risk of accidents?
- Real-time algorithm (decision support tool) needed for optimising tug positioning

Oil tanker traffic

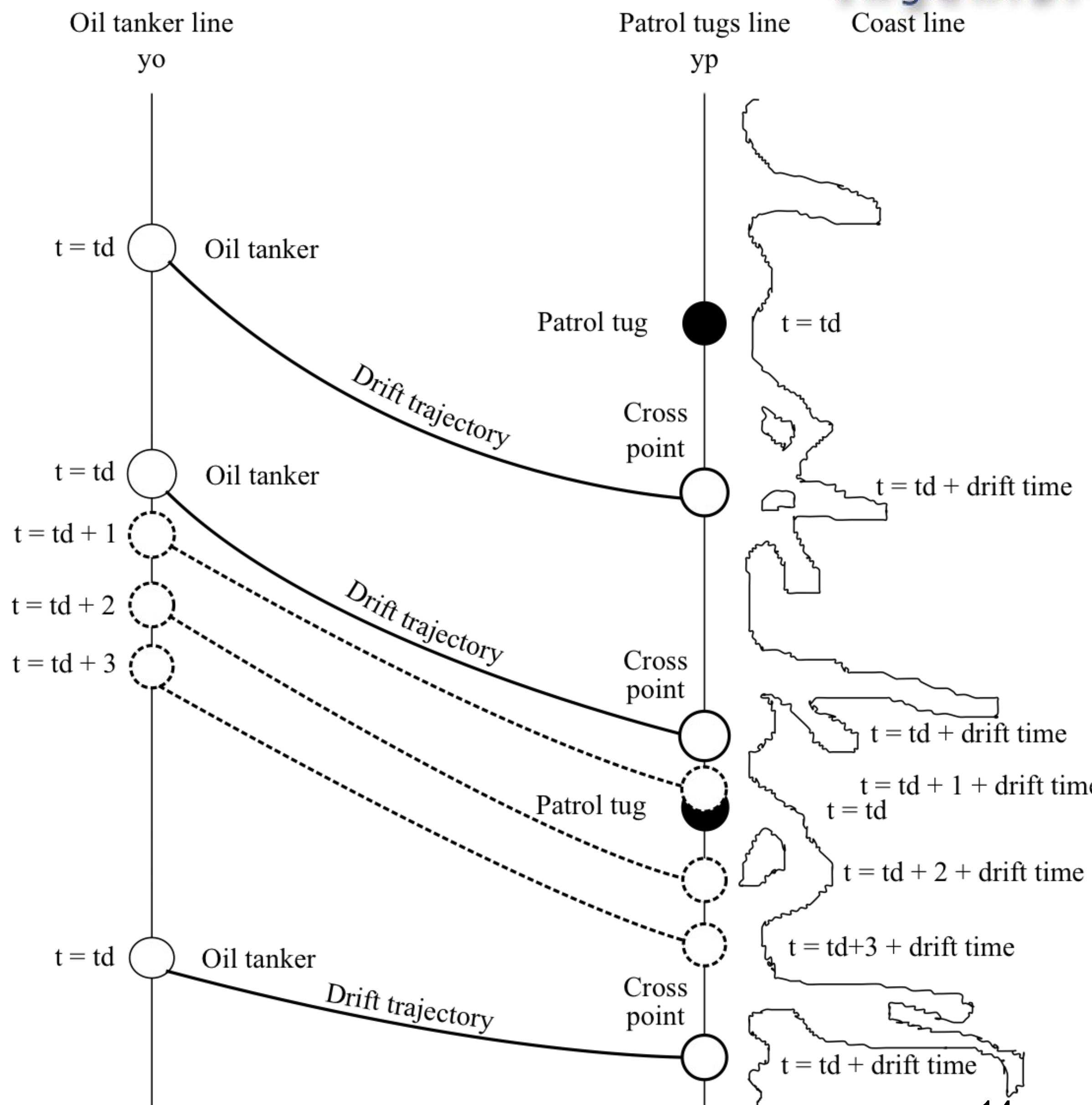
- Traffic: Along corridors
- Tugs: Near shore
- We can approximate corridors by parallel lines



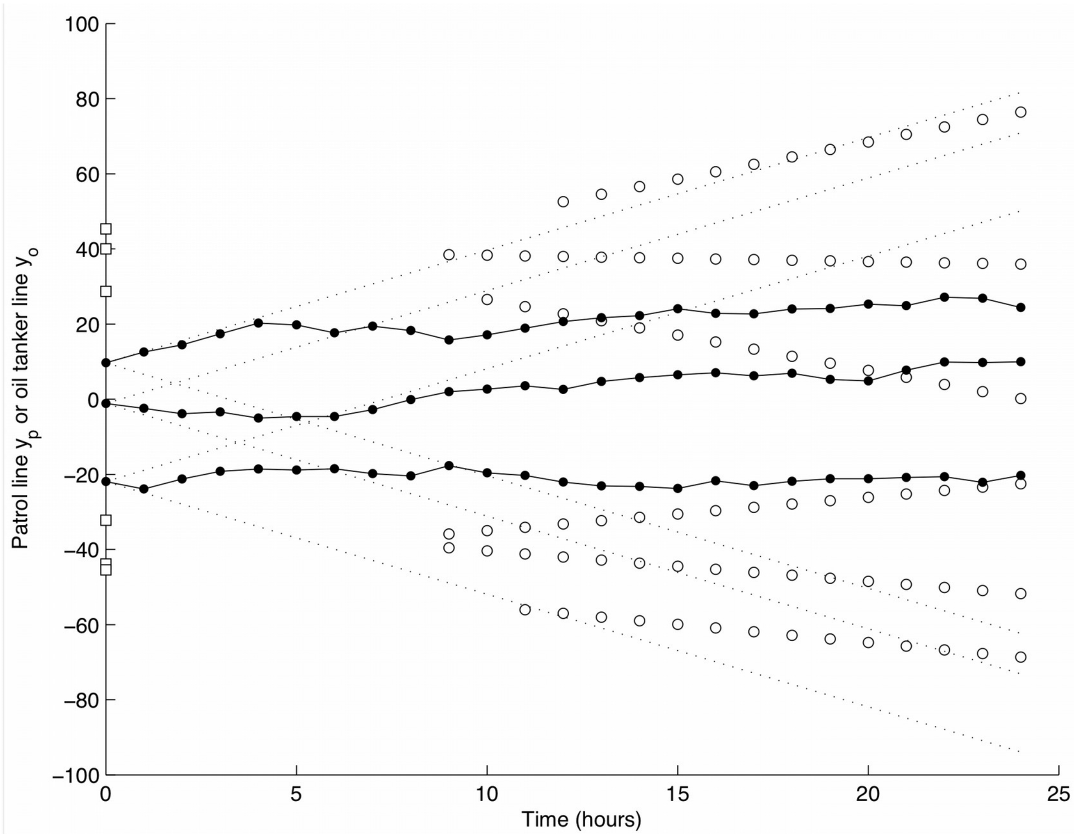
Courtesy NCA

Problem description

- Lines of motion for 3 oil tankers (white) and 2 patrol tugs (black)
- Predicted drift paths at future points in time
- How should tugs move?



Example scenario



Scenario explanation

- Crosspoint: Where drift trajectory of a tanker crosses patrol line of tugs
- Typical drift time: 8-12 hours before crossing of patrol line → entering high-risk zone
- White circles: Predicted crosspoints of drift trajectories of 6 oil tankers
- Prediction horizon T_h=24 hours ahead
- Black circles: Suboptimal trajectories of 3 tugs → How to optimise tug trajectories?

Method

- Examine a finite number of potential patrol trajectories and evaluate a cost function for each
- Use a genetic algorithm to find good solutions in reasonable time
- Use receding horizon control to incorporate a dynamic environment and update trajectories
- Plan trajectories 24 hours ahead but only execute first hour, then replan and repeat

Genetic algorithm (GA)

- Employs the usual GA scheme:
  1. Define cost function, chromosome encoding and set GA parameters, e.g., mutation, selection
  2. Generate an initial population of chromosomes
  3. Evaluate a cost for each chromosome
  4. Select mates based on a selection parameter
  5. Perform mating
  6. Perform mutation based on a mutation parameter
  7. Repeat from Step 3 until desired cost level reached

Some GA features

- Population size: Number of chromosomes
- Selection: Fraction of chromosomes to keep for survival and reproduction
- Mating: Combination of extrapolation and crossover, single crossover point
- Mutation rate: Fraction of genes mutated at every iteration

Cost function

- Sum of distances between all crosspoints and *nearest* patrol points (positions of tugs)
  - only care about nearest tug that can save tanker
- Define y_t^p as *p*th tug's patrol point at time *t*
- Define y_t^c as *c*th tanker's cross point at time *t*

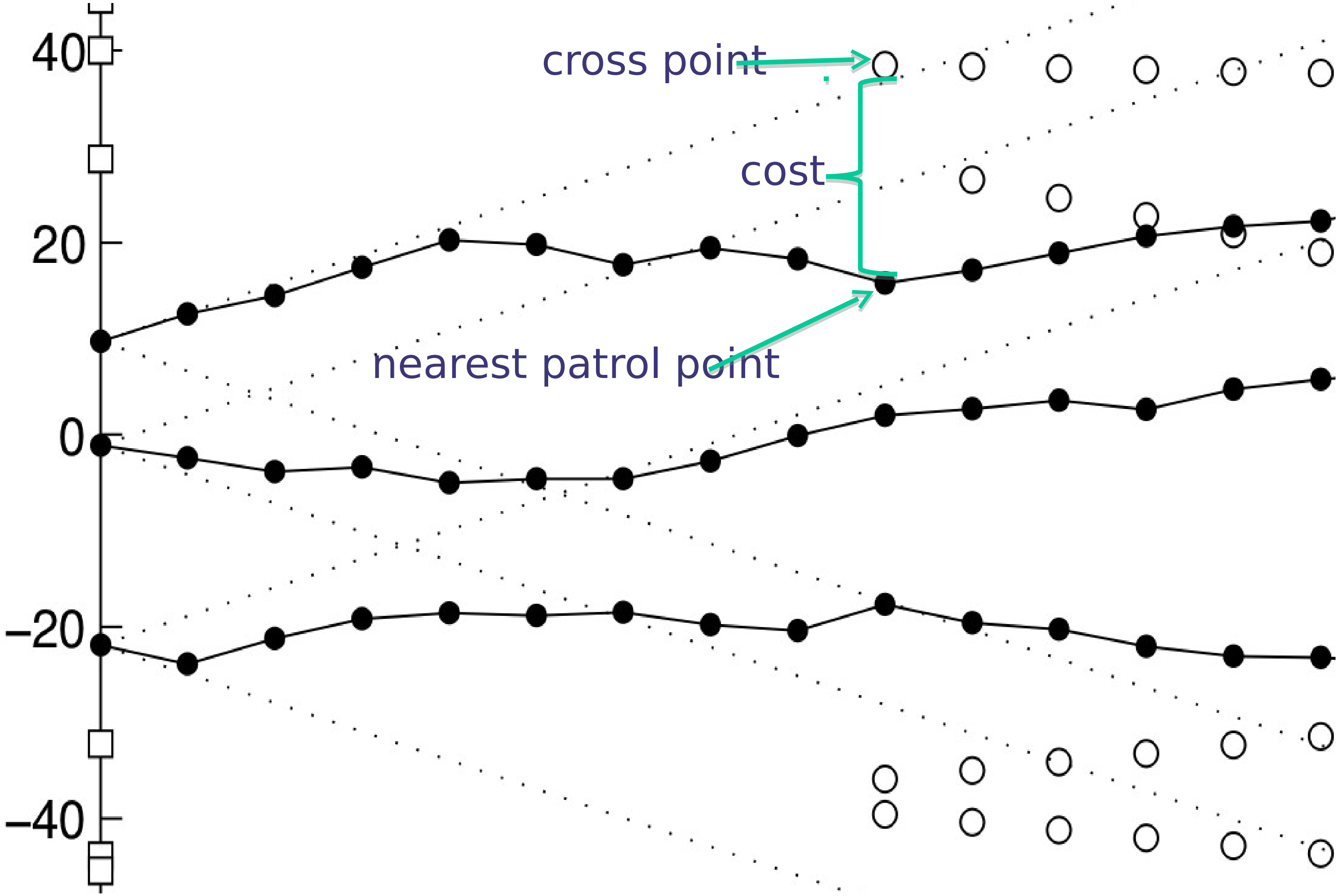
Cost function

- Consider N_o oil tankers and N_p patrol tugs
- Define y_t^p as *p*th tug's patrol point at time *t*
- Define y_t^c as *c*th tanker's cross point at time *t*
- Consider N_o oil tankers and N_p patrol tugs

Function of time *t* and chromosome C_i:

$$f(t, \mathbf{C}_i) = \sum_{t=t_d}^{t_d+T_h} \sum_{c=1}^{N_o} \min_{p \in P} |y_t^c - y_t^p|$$

Cost function cont'd



Chromosome encoding

- Contains possible set of N_p control trajectories:

$$\mathbf{C}_i = \left[u_1^1,\ldots,u_{T_h}^1,u_1^2,\ldots,u_{T_h}^2,\ldots,u_1^{N_p},\ldots,u_{T_h}^{N_p}\right]$$

- Each control trajectory $u_1^p,\ldots,u_{T_h}^p$ is a sequence of normalised control inputs with values between -1 (max speed south) and +1 (max speed north)
- Sequence of patrol points for tug p at time t from differen … sample time):

$$y_t^p = y_{t-1}^p + u_t^p v_m^p t_s$$

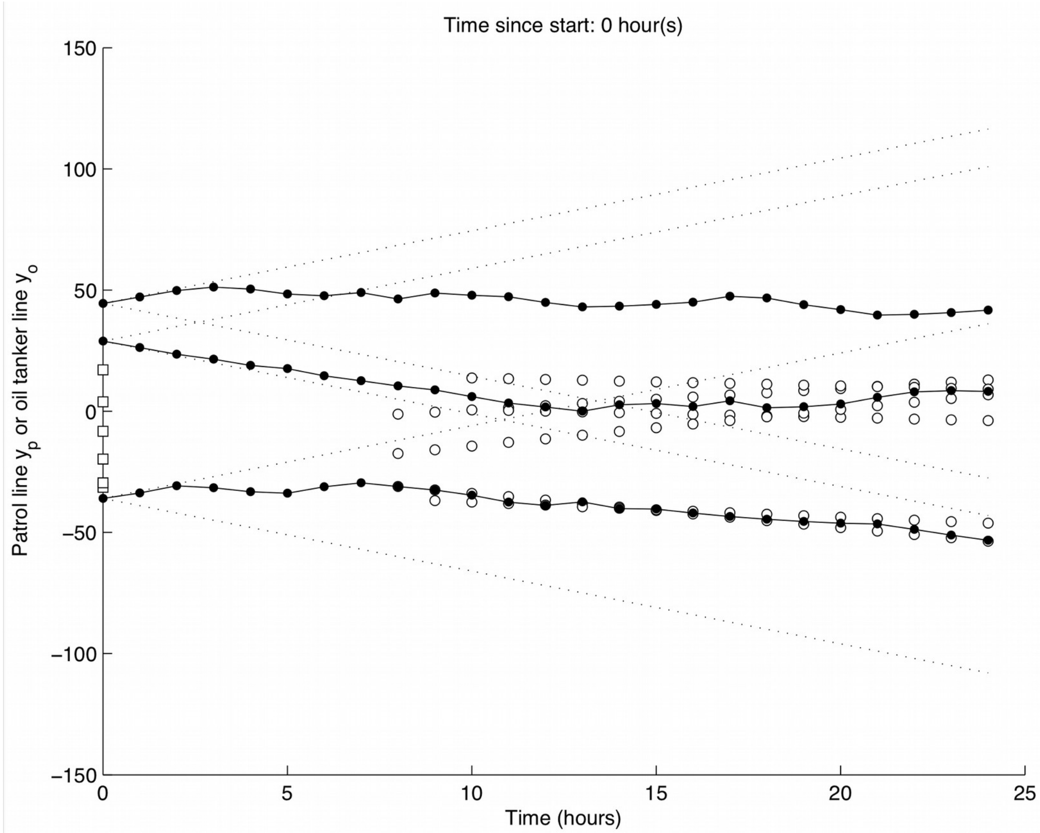
Receding horizon genetic algorithm (RGHA)

- Scenario changes over time:
  - Winds, ocean currents, wave heights, etc.
  - Tanker positions, speeds, directions, etc.
- Must reevaluate solution found by GA regularly → receding horizon control:
  1. Calculate (sub)optimal set of trajectories with duration Th (24 hours, say) into the future
  2. Execute only first part (1 hour, say) of trajectories
  3. Repeat from Step 1 given new and predicted information

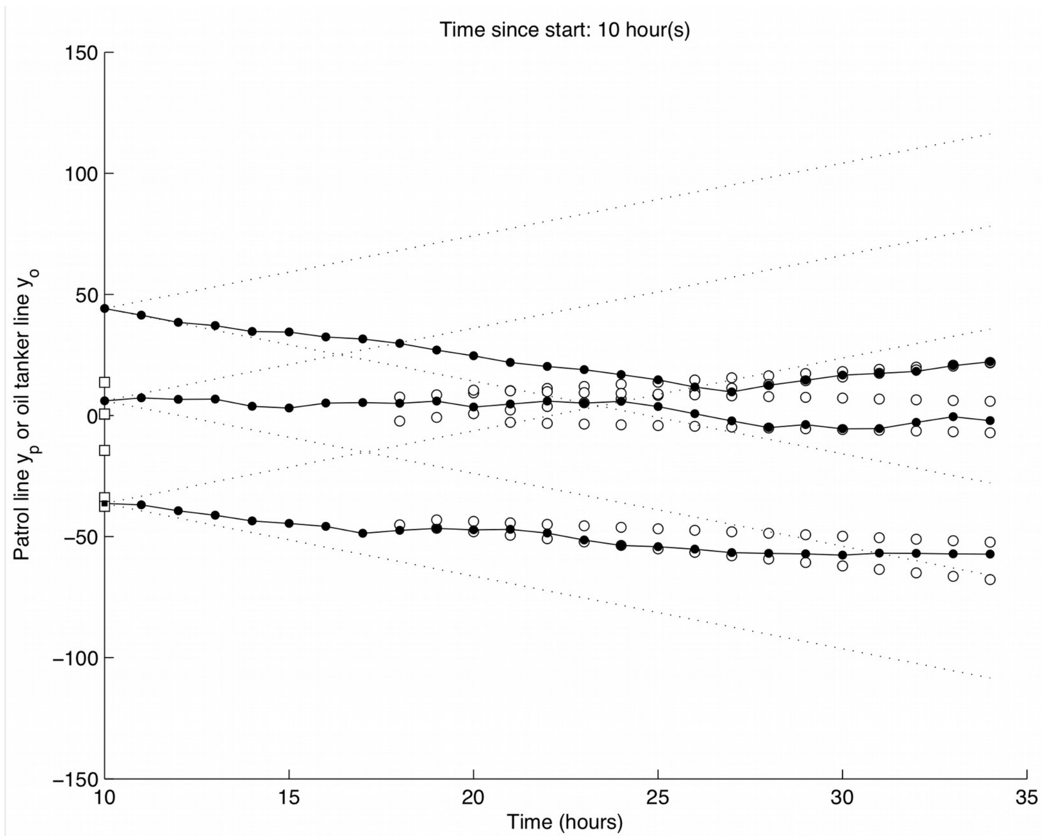
Simulation study

Oil tankers	
Number of tankers N_o	6
Random initial position	$[-50, 50]$
Random velocity	$[-1, 1]$
Drift direction	East
Random drift time Δt (hours)	$[8, 9, \ldots, 12]$
Patrol tugs	
Number of tugs N_p	3
Random initial position	$[-50, 50]$
Max velocity	± 3
GA settings	
Iterations N_{iter}	100
Population size	10
Mutation rate	0.1
Selection	0.5
RHC settings	
Prediction horizon T_h (hours)	24
Simulation step (hours)	1
Number of steps N_{RHC}	26
General settings	
Number of scenarios N_{sim}	20
Cost comparison	$\mathbf{f}_{RHGA}$, $\mathbf{f}_{static}$

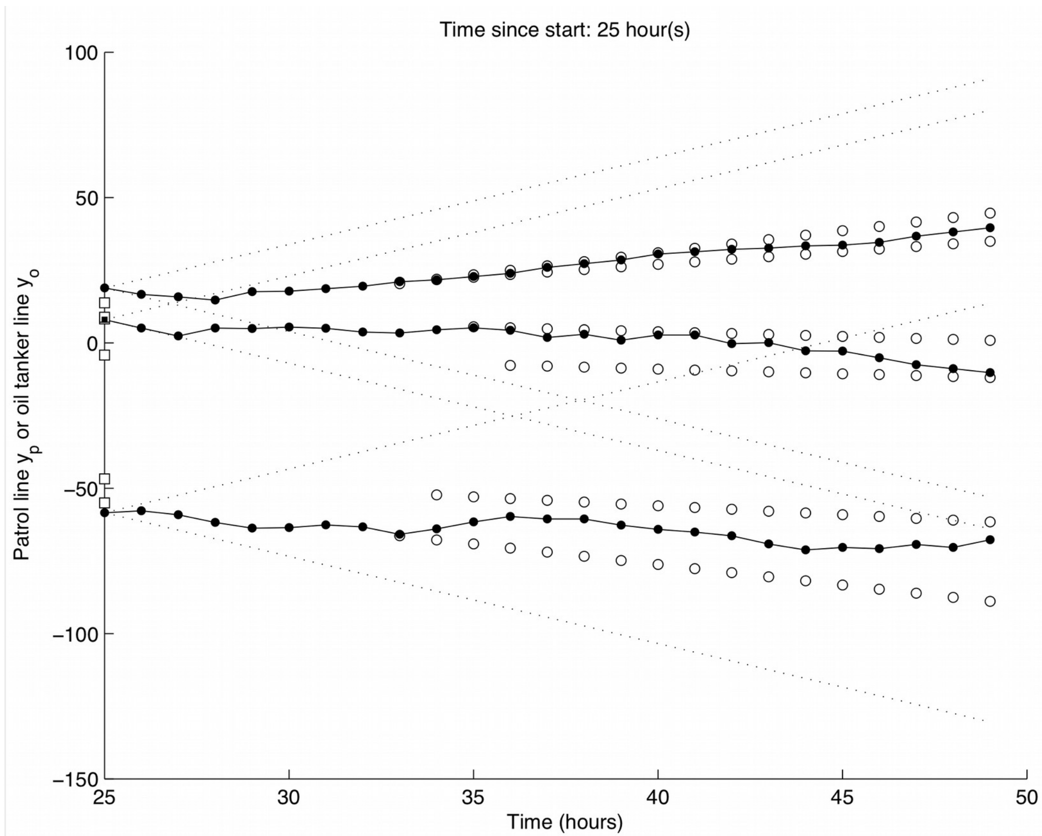
Simulation example, t_d=0 hours



Simulation example, t_d=10 hours



Simulation example, t_d=25 hours



Results

- Mean cost
  - Static strategy: 2361
  - RHGA: 808
  - Performance improvement: 65.8%
- Standard deviation
  - Static strategy: 985
  - RHGA: 292
  - Improvement: 70.4%

Conclusions

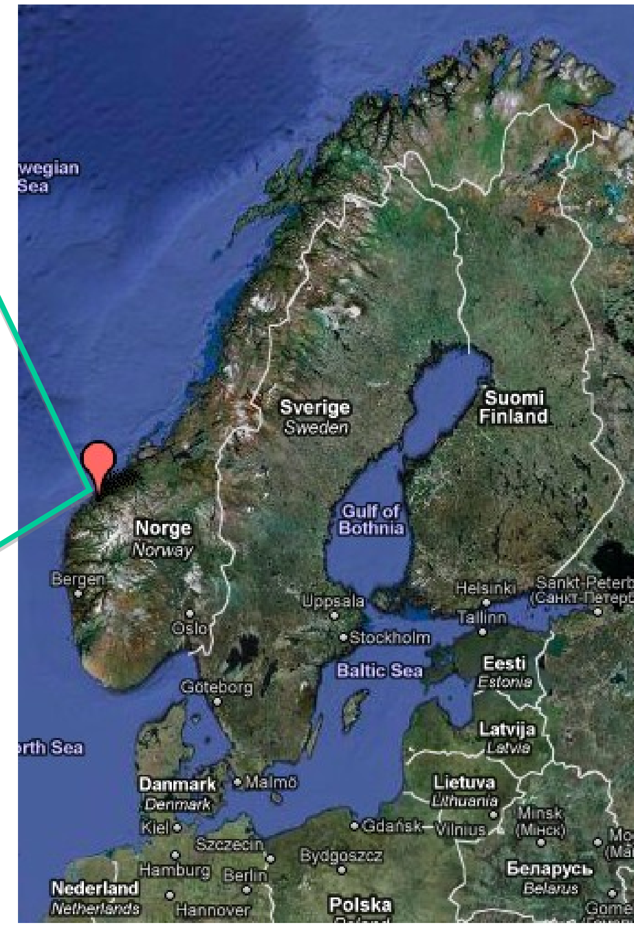
- The RHGA is able to simultaneously perform multi-target allocation and tracking in a dynamic environment
- The choice of cost function gives good tracking with target allocation “for free” (need no logic)
- The RHGA provides good prevention against possible drift accidents by accounting for the predicted future environment

Future directions

- Comparison with other algorithms
- Extend/change cost function
  - punish movement/velocity changes (save fuel)
  - vary risk factor (weight) of tankers
  - use a set of various max speeds for tankers/tugs
- Incorporate boundary conditions
- Add noise and nonlinearities
- Extend to 2D and 3D
- Test with other/faster systems

Questions?

AAUC campus



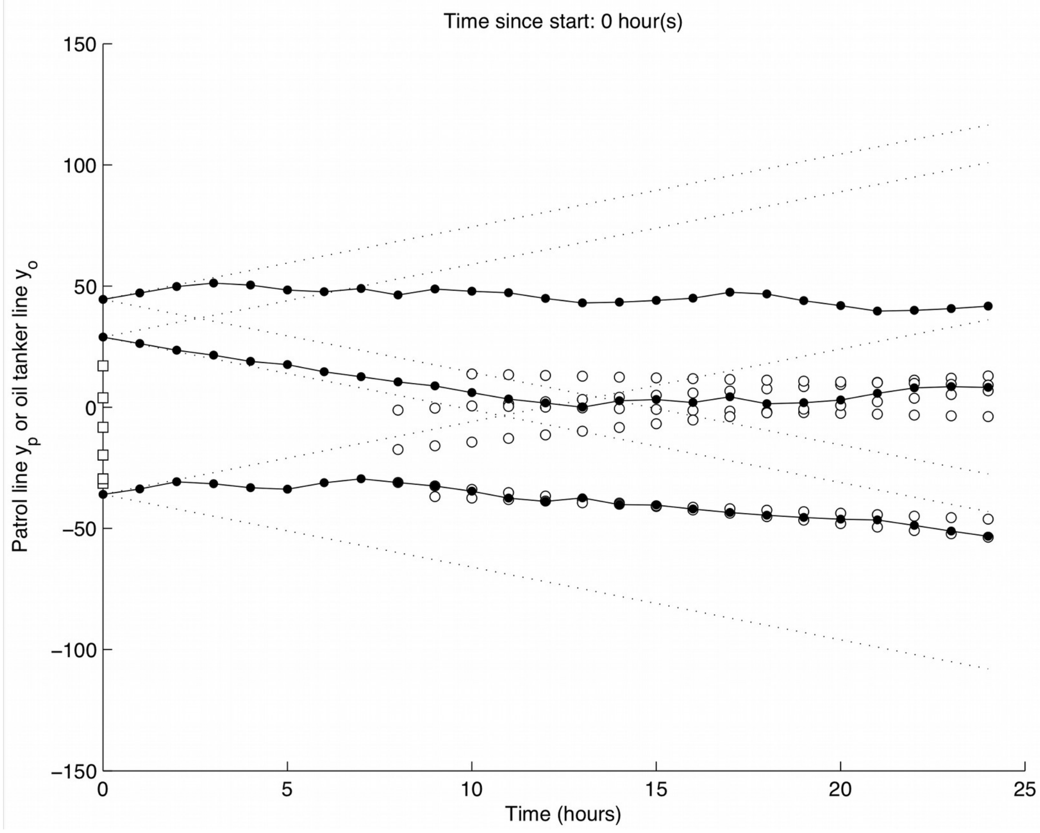
Robin T. Bye, roby@hials.no
Virtual Møre project,
www.virtualmore.org
Ålesund University College, www.hials.no

Results

Table 2: Simulation results.

Simulation run	f_{static}	f_{RHGA}	Performance (%)
1	1837.4	463.7	74.8
2	1552.2	1145.9	26.2
3	2278.0	675.1	70.4
4	3097.3	1314.0	57.6
5	2822.3	855.8	69.7
6	3929.4	1526.9	61.1
7	2431.7	633.5	73.9
8	2877.1	880.2	69.4
9	3174.7	794.0	75.0
10	1221.5	665.2	45.5
11	3839.0	1113.4	71.0
12	4356.1	914.3	79.0
13	1921.9	818.8	57.4
14	1536.1	583.4	62.0
15	1489.2	869.5	41.6
16	1546.6	575.5	62.8
17	1456.7	457.1	68.6
18	1836.8	445.8	75.7
19	950.8	559.9	41.1
20	3068.0	874.4	71.5
Mean	2361.2	808.3	65.8
Standard dev.	984.7	291.6	70.4
Best run: 12			79.0
Worst run: 2			26.2

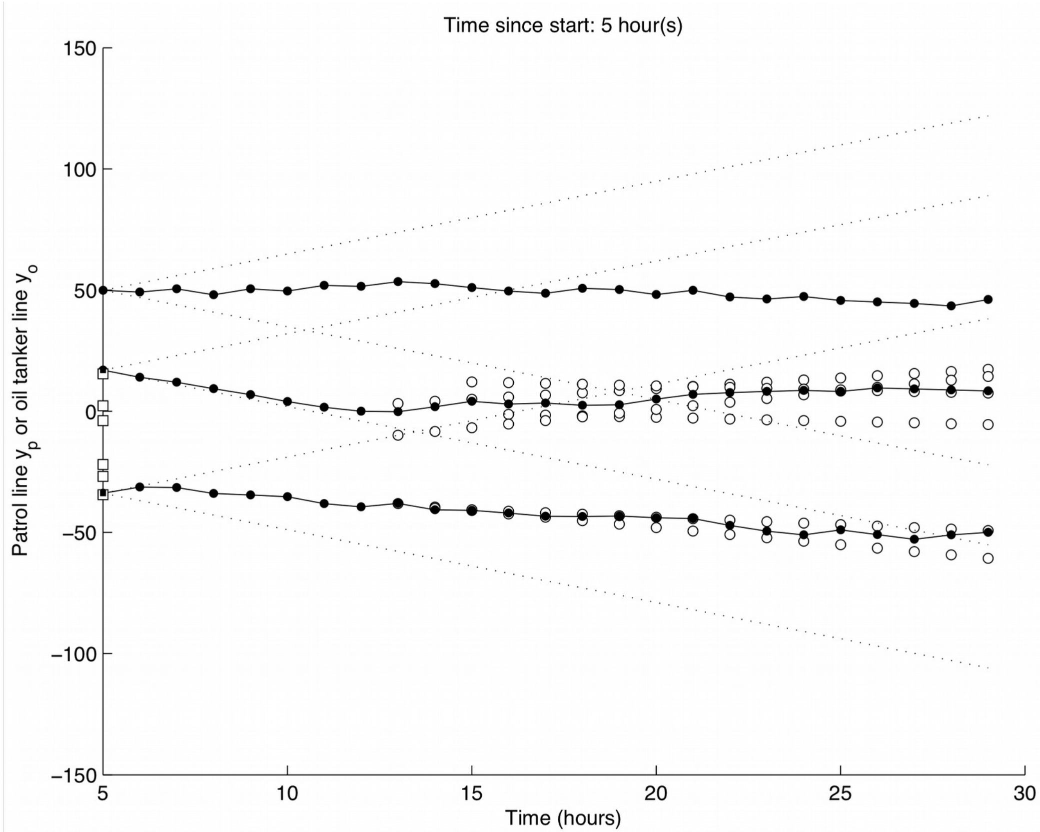
Simulation example, t_d=0 hours



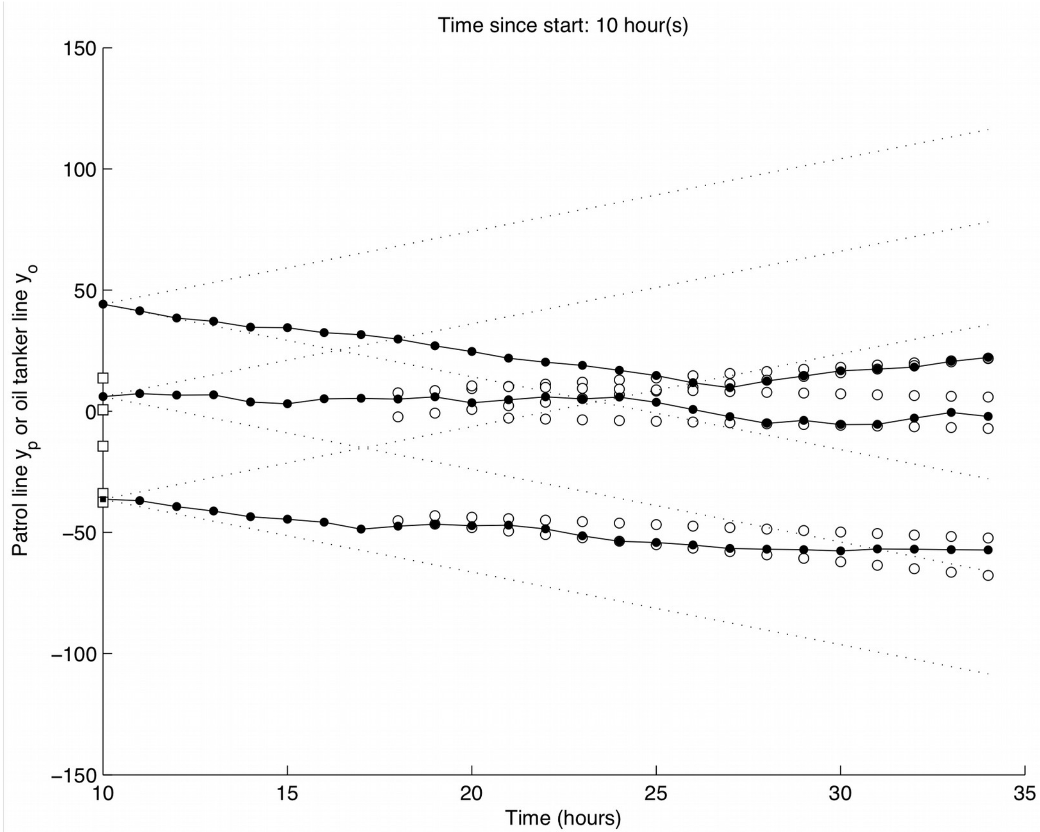
HØGSKOLEN I ÅLESUND

Ingeniør

Simulation example, t_d=5 hours



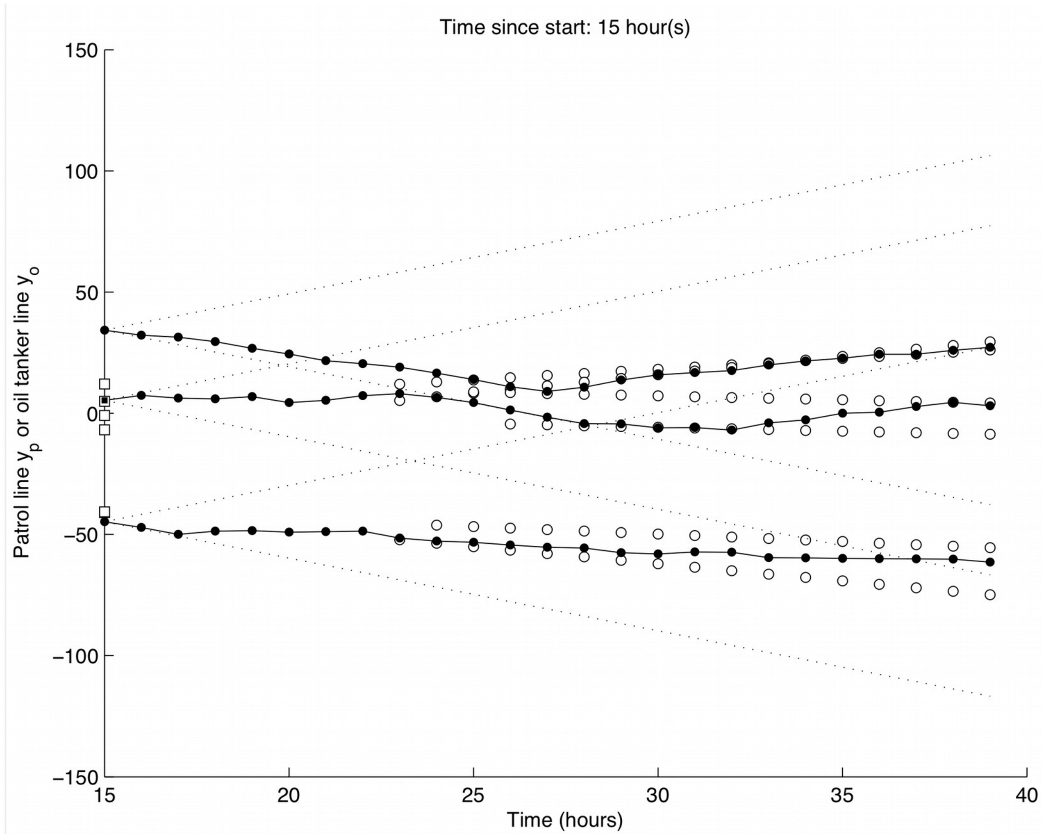
Simulation example, t_d=10 hours



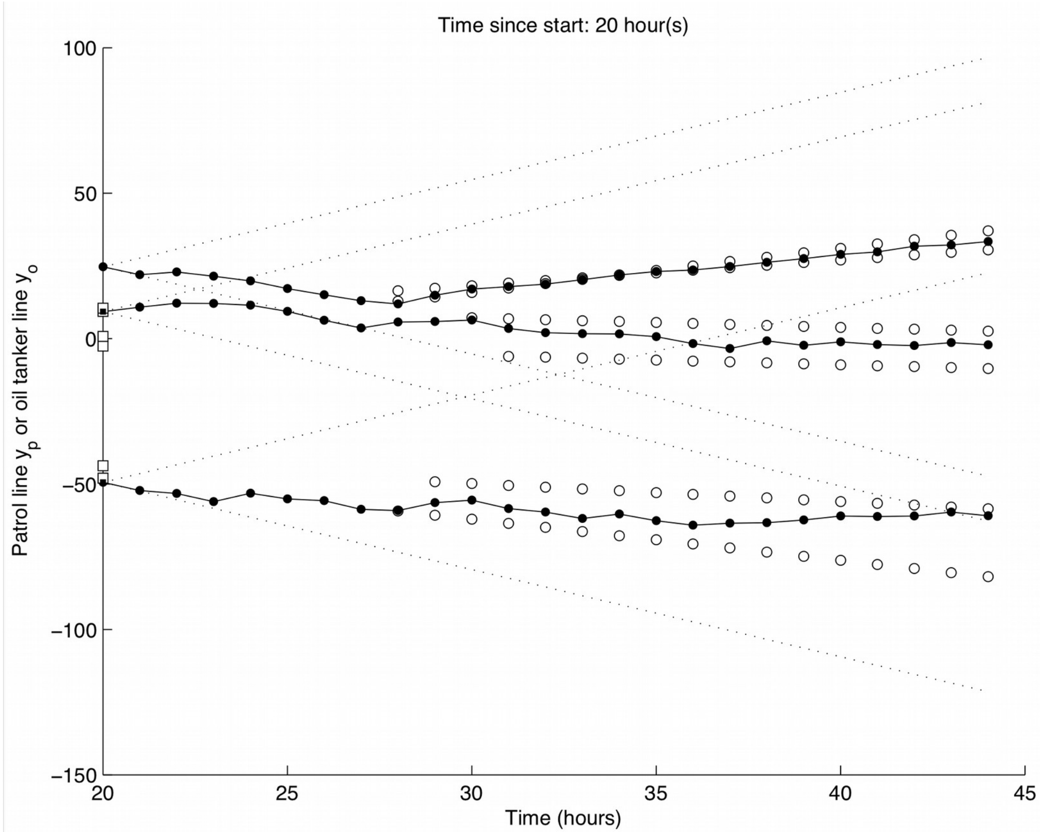
HØGSKOLEN I ÅLESUND

Ingeniør

Simulation example, t_d=15 hours



Simulation example, t_d=20 hours



HØGSKOLEN I ÅLESUND

Ingeniør

Simulation example, t_d=25 hours