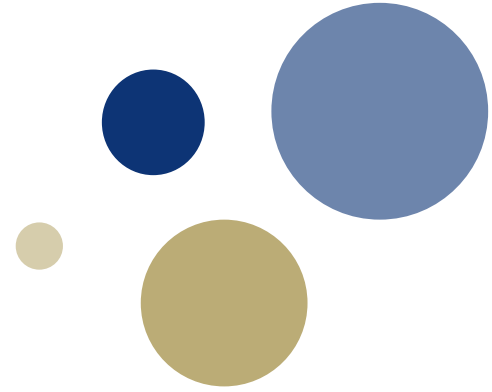
NTNU

Norwegian University of
Science and Technology



Intelligent Virtual Prototyping of Offshore Cranes

Robin T. Bye[1], Hans Georg Schaathun[1], Birger Skogeng Pedersen[1,2], Ibrahim A. Hameed[1], and Ottar L. Osen[1,2]

[1]Software and Intelligent Control Engineering (SoftICE) Laboratory,
Faculty of Engineering and Natural Sciences, NTNU in Ålesund, Norway
email: robin.t.bye@ntnu.no | web: blog.hials.no/softice
[2]ICD Software AS, Ålesund, Norway

MODPROD 2016, Linköping, Sweden, 2–3 February 2016

Background

About NTNU in Ålesund and virtual prototyping

- NTNU in Ålesund (formerly Aalesund University College) has close ties with the maritime cluster of Norway, relating to education, research, innovation, and dissemination
- many past and ongoing research and innovation projects in collaboration with the industry
- bachelor and master engineering programmes in automation, computer, and power systems engineering; product and system design; ship design; simulation and visualisation; management of demanding marine operations; and more
- virtual prototyping (VP) of maritime equipment currently has a strong research focus
- today's presentation: a computer-automated design solution for intelligent virtual prototyping of offshore cranes

What is virtual prototyping (VP)?

Many definitions exists; e.g. [1]:

> *. . . a virtual prototype, or digital mock-up, is a computer simulation of a physical product that can be presented, analyzed, and tested from concerned product life-cycle aspects such as design/engineering, manufacturing, service, and recycling as if on a real physical model. The construction and testing of a virtual prototype is called virtual prototyping (VP).*

or Wikipedia:

> *Virtual prototyping is a method in the process of product development. It involves using computer-aided design (CAD), computer-automated design (CAutoD) and computer-aided engineering (CAE) software to validate a design before committing to making a physical prototype.*

Key aspects of VP

- modelling, simulation, visualisation, analysis, testing, validation, optimisation, process planning, immersive collaborative design, etc.
- some relevant tools include virtual reality (VR), virtual environments (VE), computer-aided design (CAD), computer-aided engineering (CAE), computer-automated design (CautoD), hardware-in-the-loop (HiL) simulation, etc.
- better chance of reaching targets such as performance, revenue, cost, launch date, quality, bugs and flaws, etc.
- opens possibilities for new and innovative design, including improved performance

What is computer-automated design (CautoD)?

- first (?) occurrence in 1963 [2]: computer programme for design of logic circuits for character recognition
  - do the circuits satisfy hardware constraints?
  - how well do they perform character recognition?
- the general paradigm is optimisation
  $\Rightarrow$ minimise (maximise) a cost (fitness) function
- artificial intelligence (AI) highly suitable for optimisation, e.g., genetic algorithms (GAs), particle swarm optimisation (PSO), ant colony optimisation (ACO), simulated annealing, etc.
- trend: traditional CAD simulation transformed to CautoD by AI
- design problem: find best design within known range (i.e., through learning or optimisation) and find new and better design beyond existing ones (i.e., through creation and invention) (Wikipedia)
- Equivalent to a search problem in multidimensional (multivariate), multi-modal space with a single (or weighted) objective or multiple objectives (Wikipedia)

VP of offshore cranes at NTNU in Ålesund

- VP of offshore cranes active focus of research at NTNU in Ålesund
- ships, cranes, winches, crew, etc. in advanced maritime operations are complex systems (hydrodynamics, hydraulics, mechanics, electronics, control systems, human factors)
- workspace characteristics essential (2D load chart of lifting capacity)
  - depends on cylinders, links, sheaves, joints, etc.
  - often indirect consequence of a priori design choices
  - traditionally experience-based rules-of-thumb design
  - recent work use trial-and-error to improve design [3]
    $\Rightarrow$ cumbersome, suboptimal method; only a few design parameters are tuned; novelties may not be discovered

Offshore cranes

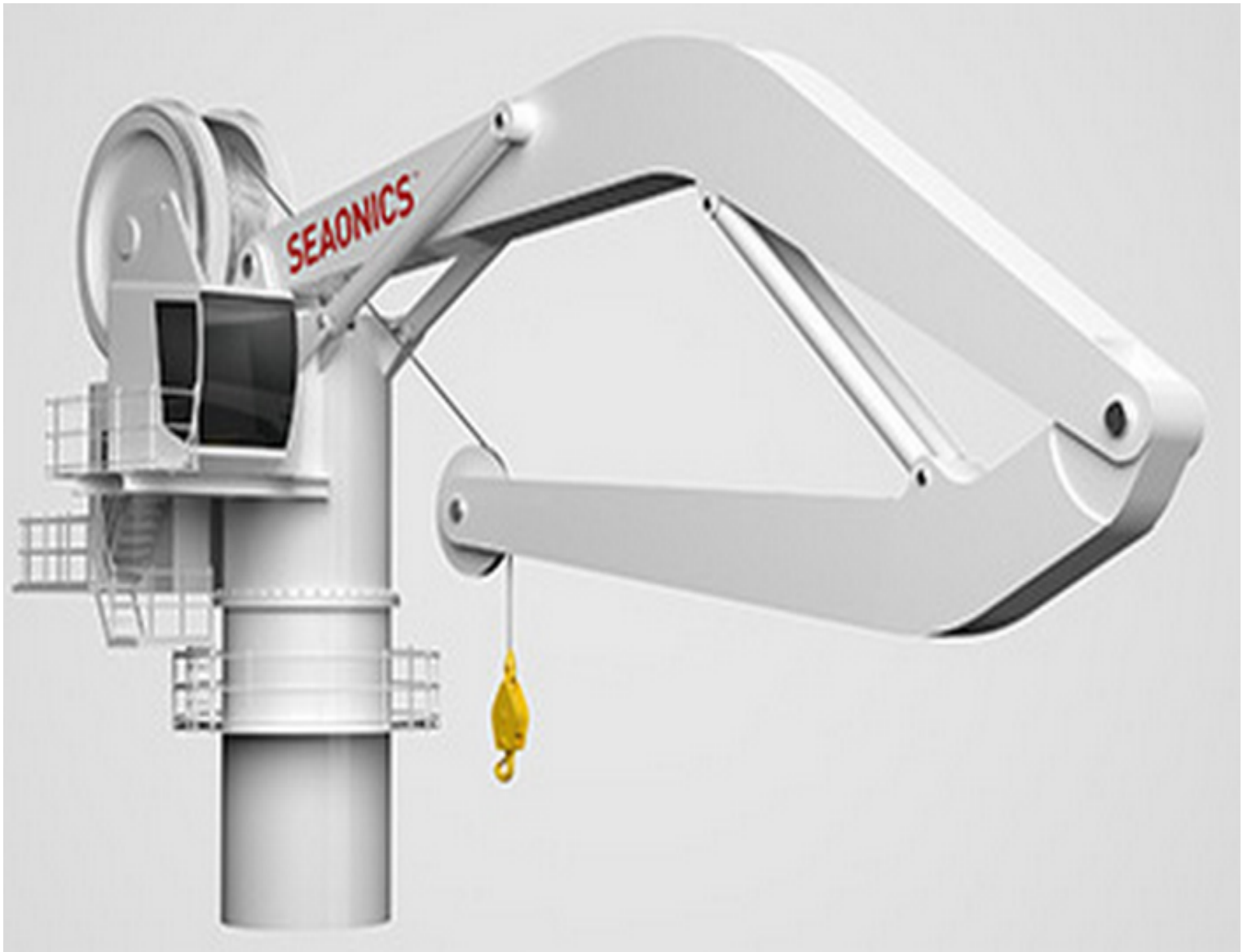
Seaonics and crane types

- Seaonics is industrial partner in our research project
- located in Ålesund, Norway and central to the maritime cluster
- designer and manufacturer of offshore handling equipment for critical lift and handling operations
- offshore/subsea cranes
  - 50T offshore/subsea crane been delivered (80T in 2-fall)
  - crane with 250T safe working load (SWL) been designed in a pilot project
  - drawings of various crane sizes up to 250T prepared
- marine cranes
  - cranes from 0.5–20T with various reach
  - ship-to-ship operations
  - handling of personnel in baskets

Typical Seaonics knuckleboom crane

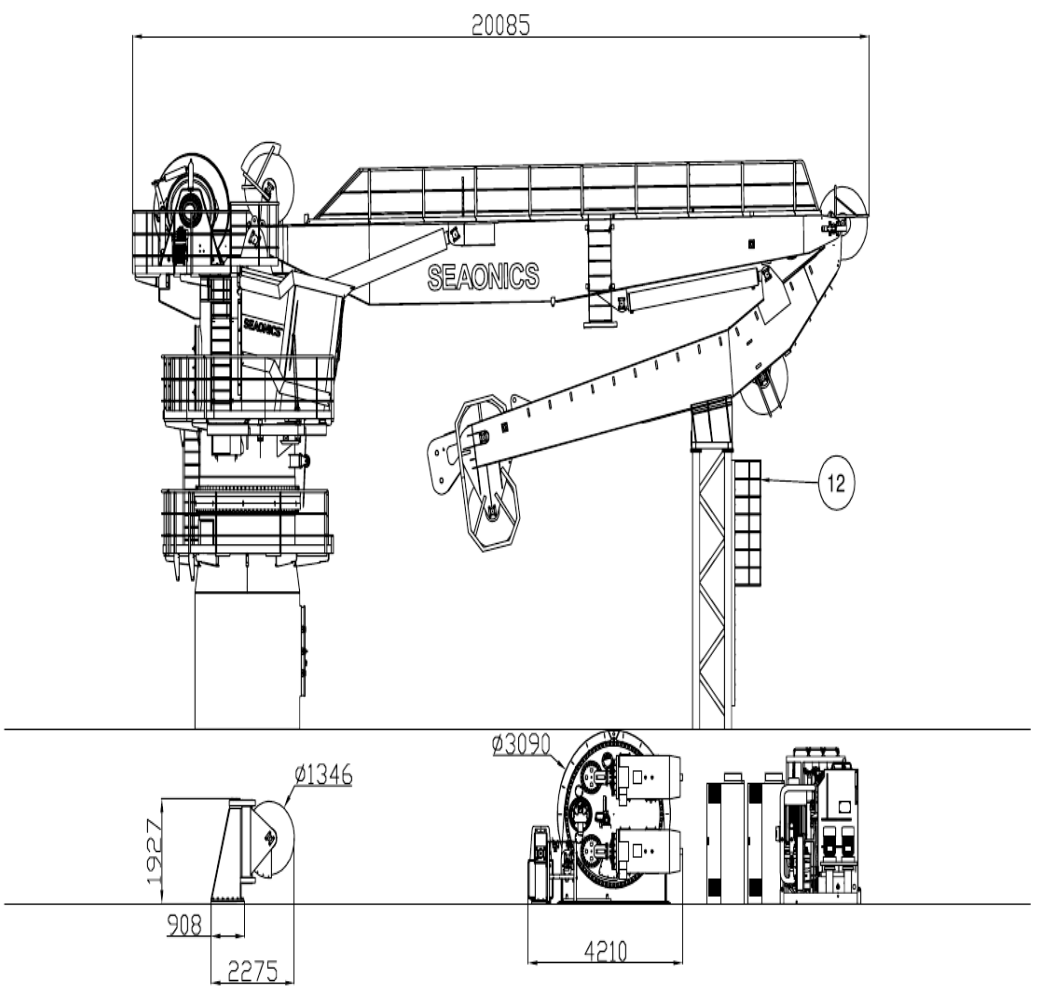
- winch
  - capacity up to 3000m of wire
  - designed according to DNV Standard for certification No. 2.22, June 2013
- operators cabin
  - innovative design, based one the highest quality standards
  - made in Germany
- machinery house
  - location of HPU, starter cabinets and operational valves
  - easy access for maintenance and service
- main boom cylinders lifted to improve sideways view for the crane operator
- walkways/ladders fitted for easy access to maintenance points
- hydraulic piping
  - walform fittings up to and including 42mm pipes
  - stainless steel pipes up to and including 42mm
- “standard” components from recognized suppliers located in Europe

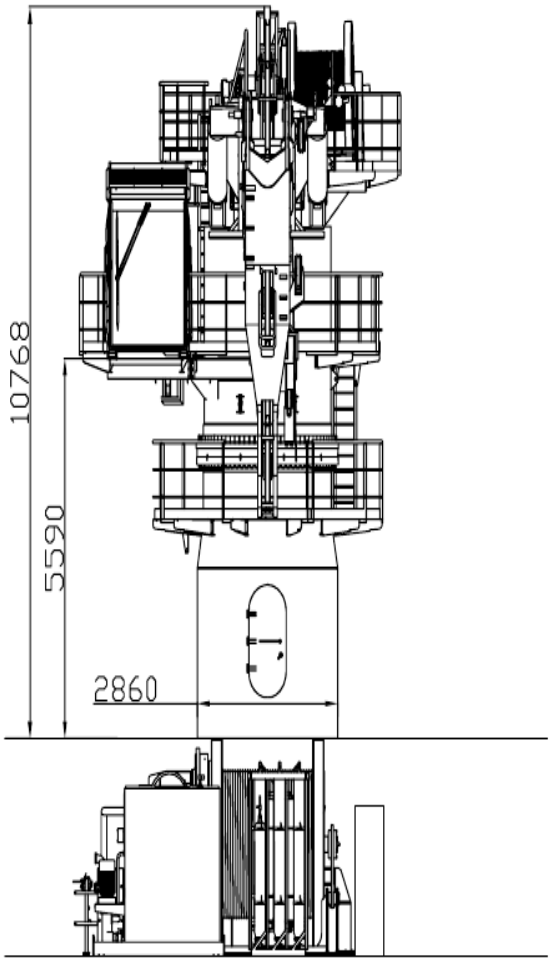
Example of a knuckleboom crane



50T knuckleboom crane delivered to Baku, Azerbaijan

Some engineering drawings for Baku crane





Some Baku crane facts

— delivery price: 28 MNOK (ca. 3.25 MUSD)
— estimated total crane weight: $\approx$ 50T
— maximum safe working load (SWL): 100T
— some important design parameters affecting weight and SWL:
  - boom length: 15.8 m
  - jib length: 10.3 m
  - max pressure of main cylinder: 315 bar
  - max pressure of jib cylinder: 215 bar

How can we optimise the design parameters to minimise total crane weight while maximising SWL?

Motivation and aim

Motivation

— traditional methods use “calculators” to find crane properties and behaviour based on pre-determined design parameters
$\Rightarrow$ analogous to “forwards kinematics” in robotics

— the inverse problem is much harder and analytical solutions are generally infeasible

— the research question becomes:

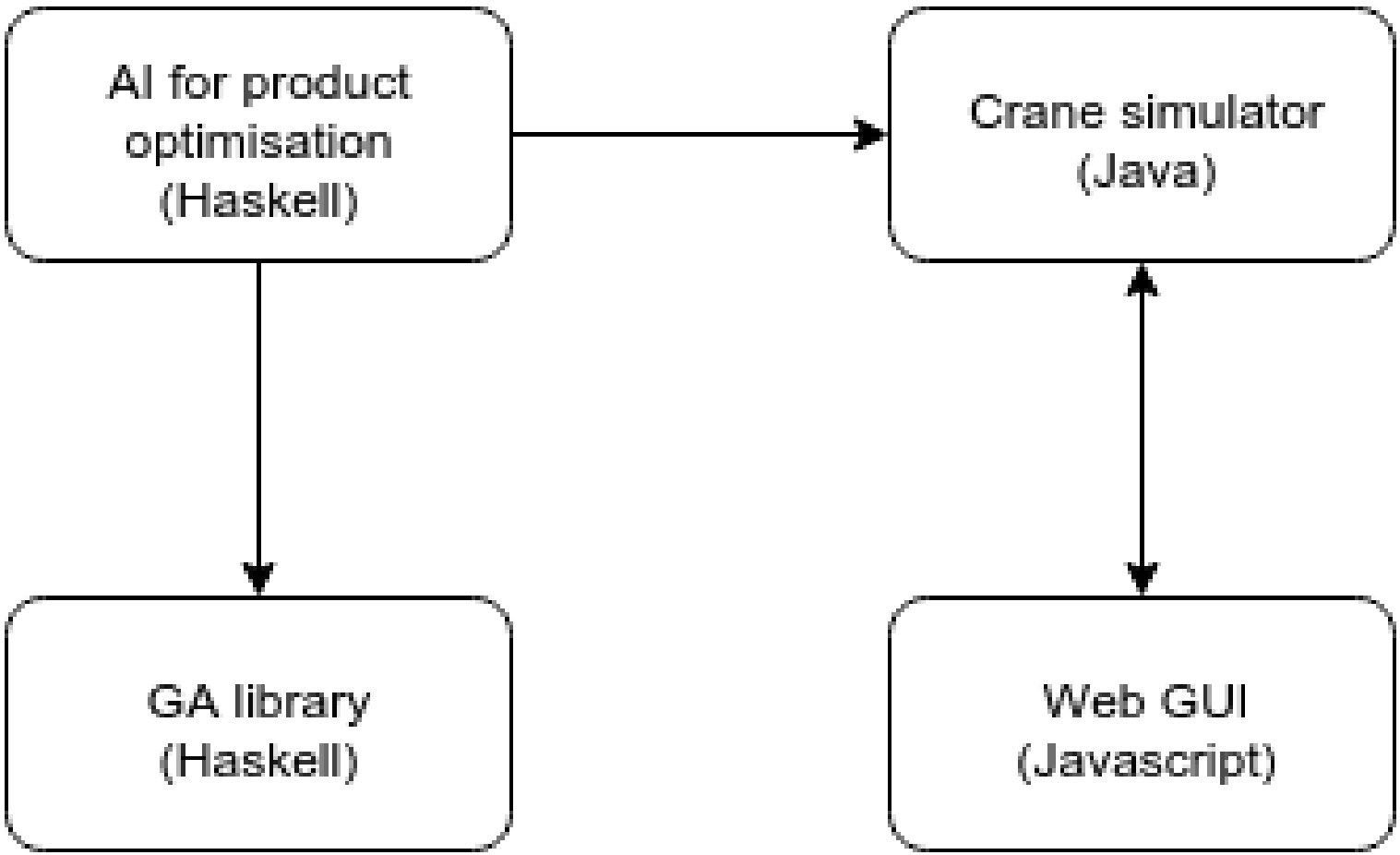
How can we choose appropriate, and possibly conflicting, values for numerous, offshore crane design parameters such that the resulting cranes have the desired properties and behaviour that we want?

Aim

Create a CautoD tool whose main components include

1. a black box crane simulator implemented in Java that calculates a crane's properties for a given set of design parameters or specifications
2. a web graphical user interface (GUI) implemented in Javascript that enables a crane designer to manually input design parameters and calculate the corresponding crane and its properties
3. an AI for product optimisation (AIPO) module implemented in Haskell that employs a GA library, also implemented in Haskell, to feed sample design solutions to the crane simulator in order to optimise some objective function, which is specified such that the optimal design solution yields the required specifications for a crane to achieve certain desired design criteria
4. communication interfaces between the crane simulator and the web GUI and AIPO module

Software dependencies

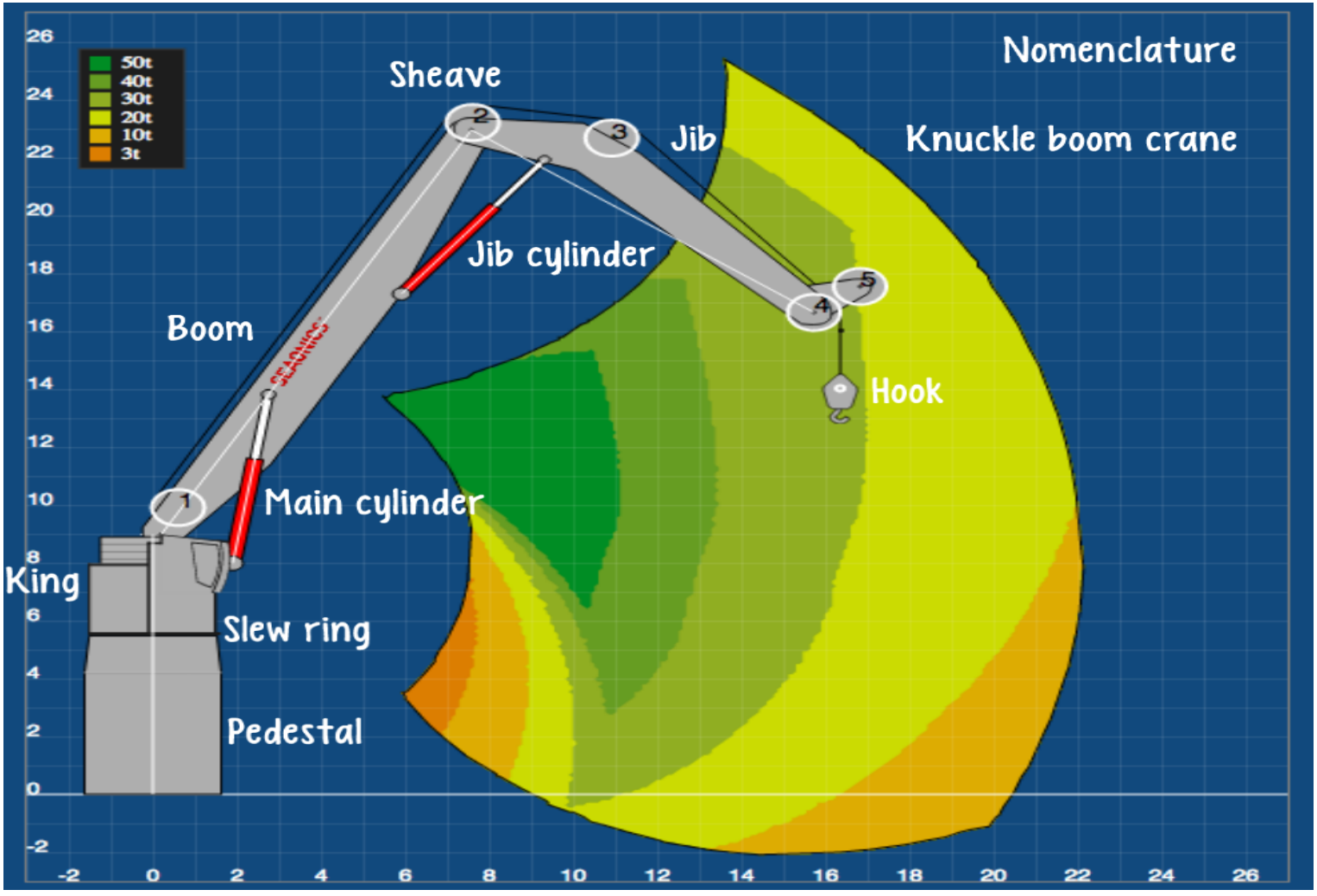


Method

Components, characteristics, key performance indicators (KPIs), constraints

- design parameters are mainly the crane components
- components include hooks, winches, slewing rings, cylinders, booms, hinges, sheaves, pedestals, etc.
- characteristics and key performance indicators (KPIs) are affected by placements, types, capacities, materials, and abilities of components
- KPIs include desired workspace, working load limit (WLL), safe working load (SWL), total weight, control system characteristics, durability, installation and operating costs, safety concerns (e.g., wind impact), etc.
- design constraints may be derived from physical and financial restrictions and laws, regulations, standards, design codes (by DNV-GL, Lloyd's, etc.)

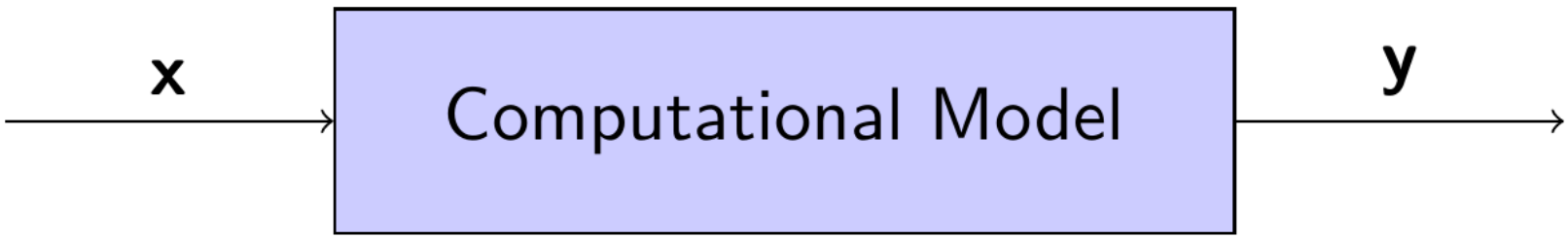
Main components and 2D load chart



Computational model and simulator

— cannot include all parameters in computational model (CM)
$\Rightarrow$ reduce to 120 parameters for feasability

— CM is implemented in software as a simulator that calculates outputs **y** dependent on parameter inputs **x**



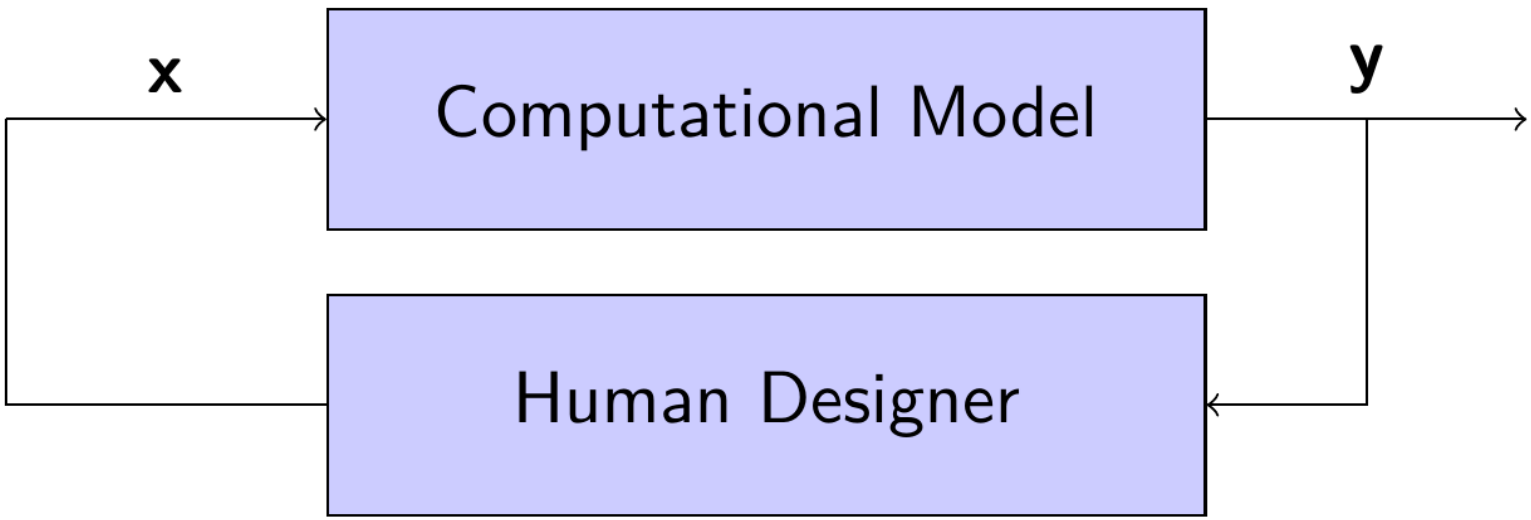
— outputs are a set of design criteria (max SWL, load chart, weight, etc.)

— simulator accuracy been verified with current in-use industry crane calculators

Traditional design by trial-and-error

— finding suitable design **x** that yields desired **y** analytically is not possible (inverse solution)

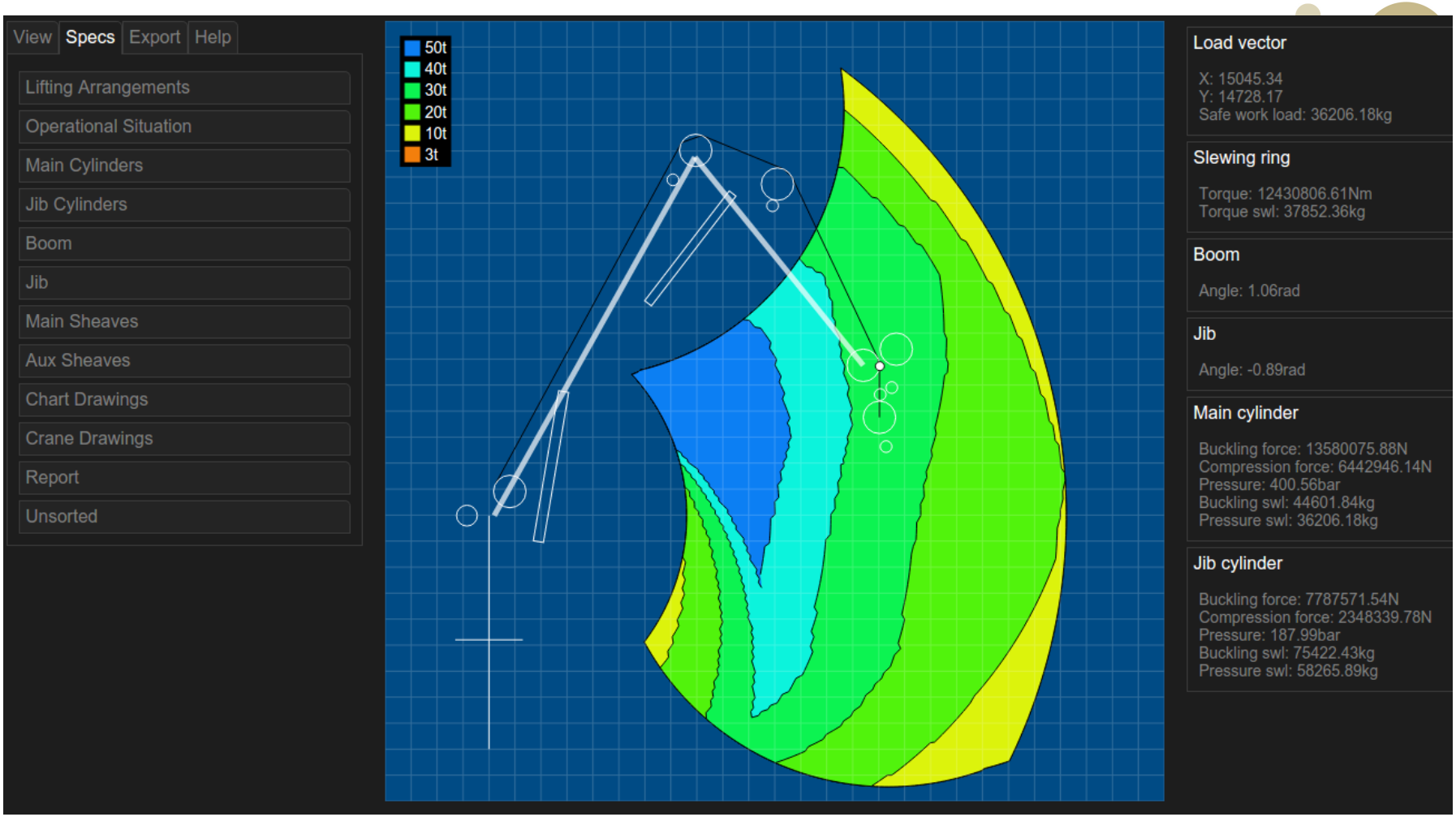
— can manually tune 120 design parameters and observe effects on design criteria (forward solution)



— improve design by repeated trial-and-error using graphical web interface

$\Rightarrow$ time-consuming, suboptimal, may miss novelties

Graphical web interface



Genetic algorithms

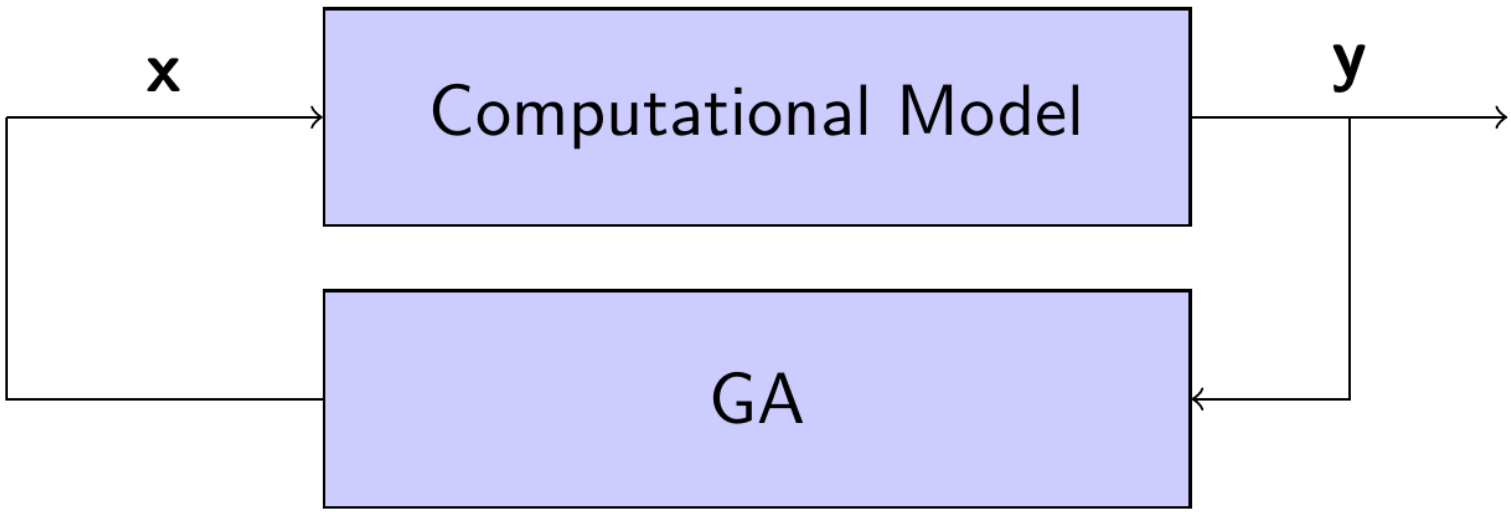
- bio-inspired stochastic search heuristic for search and optimisation problems
- inspired by natural evolution and uses inheritance, mutation, selection, crossover
- usually attributed to Holland and popularised by Goldberg
- several applications at AAUC, including dynamic resource allocation, adaptive locomotion of caterpillar robots, universal control architecture for maritime cranes/robots, machine learning, optimisation of boid swarms, etc.
- requires suitable objective function incorporating design criteria we wish to optimise
- suitable for parallel computing

Objective functions

- GA finds $\mathbf{x}$ that optimises objective function $f(\mathbf{x})$
- $f(\mathbf{x})$ called cost (fitness) function if minimised (maximised)
- 120 input parameters means search space is huge
- difficult to find appropriate objective function
- tradeoff in optimising conflicting design criteria
  $\Rightarrow$ may require multiobjective optimisation (MOO) with a set of several objective functions
- MOO returns a set of Pareto optimal solutions
  $\Rightarrow$ improving one solution degrades another

Automated design solution

— replace human designer by automated solution using GA



— initial work: let GA optimise subset of 120 parameters

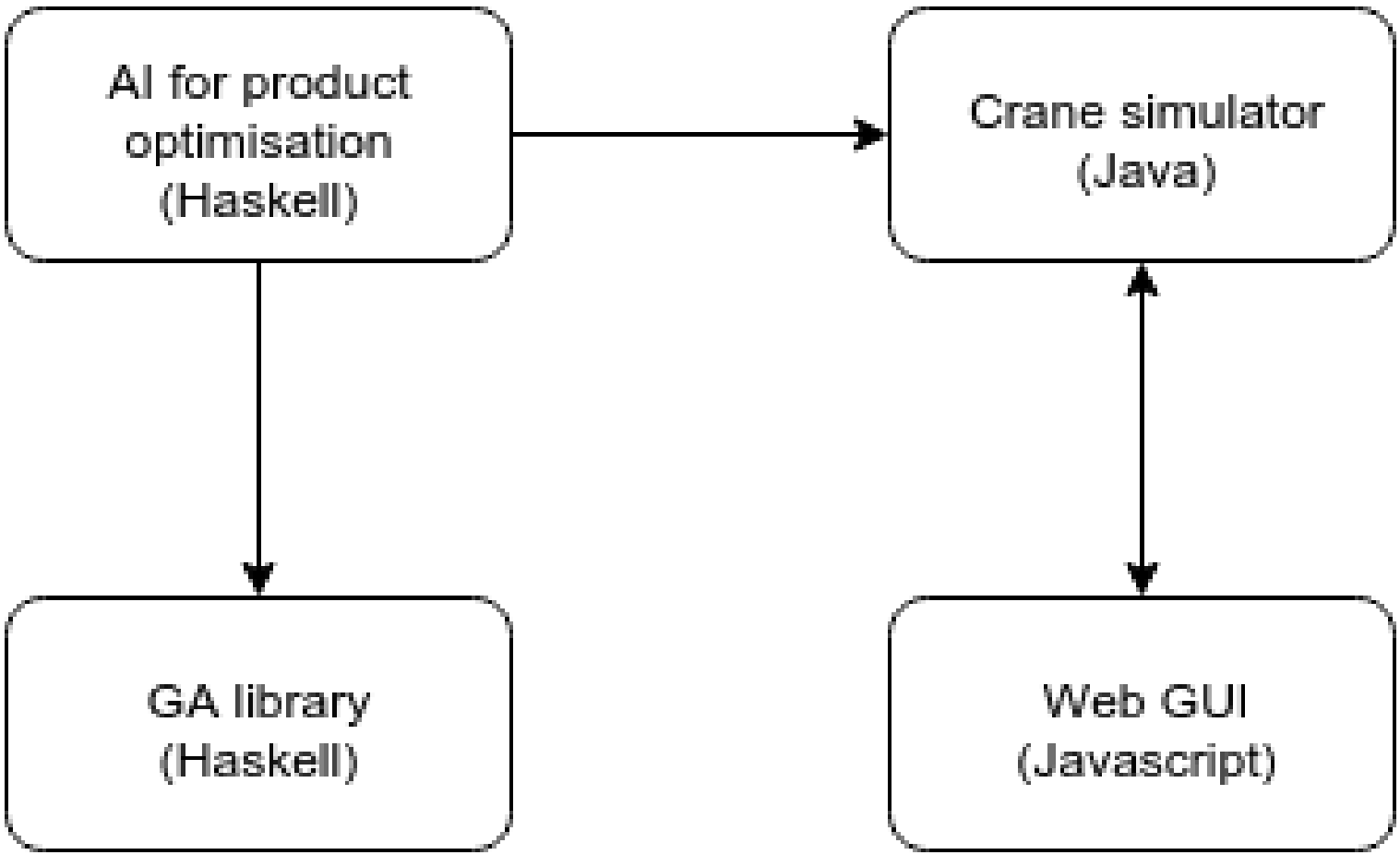
— easy to extend later when we know more about computational overhead and feasibility of approach

— important design criteria: load chart, weight, cost

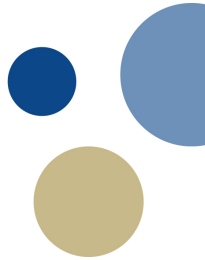
— use relative weighting of design criteria in objective function

Results

Complete system implemented and working (alpha version)



Crane Prototyping Tool (CPT), AIPO, and GA



- CPT consists of CM, simulator, and GUI for crane design
- highly detailed CM of offshore knuckleboom cranes
  - mathematical models provided by partner Seaonics AS
  - accuracy verified with industry crane calculators
  - adheres to laws, regulations, standards, design codes
- simulator implemented in Java by ICD Software AS
- two online server versions (graphical + WebSockets/HTTP)
- Haskell client (AIPO) + Haskell GA that use WebSocket and JSON for communication
- server-side generation of 1 full set of crane data (120 parameters) for a candidate design takes less than a second
- typical total processing time for population of 100 GA candidate solutions for 50 generations: $\approx$6000 sec (100 min)
- GUI version already adopted for professional use by Seaonics

Proof-of-concept experiment

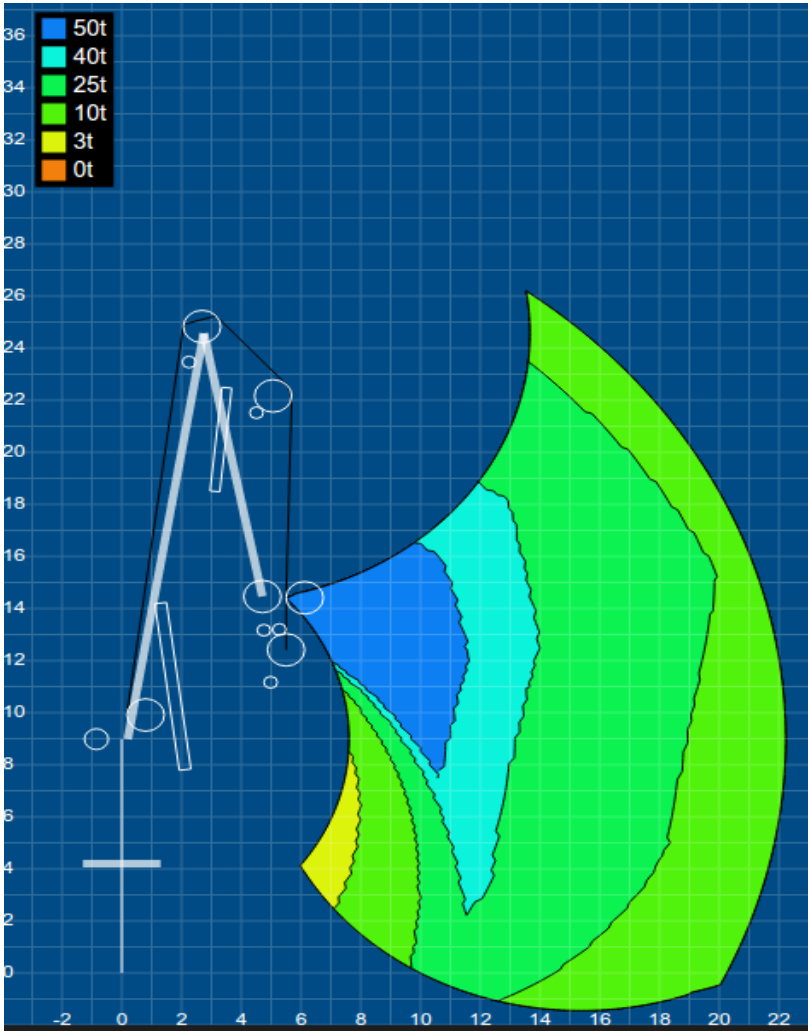
- use designed and delivered Baku crane as benchmark
- optimise 4 design parameters: boom length, jib length, max pressure of main cylinder, max pressure of jib cylinder
- remaining design parameters kept the same as for Baku crane
- objective functions based on 2 KPIs: total weight of crane and maximum SWL
- intelligent virtual prototyping crane design outperforms Baku crane for chosen objective functions

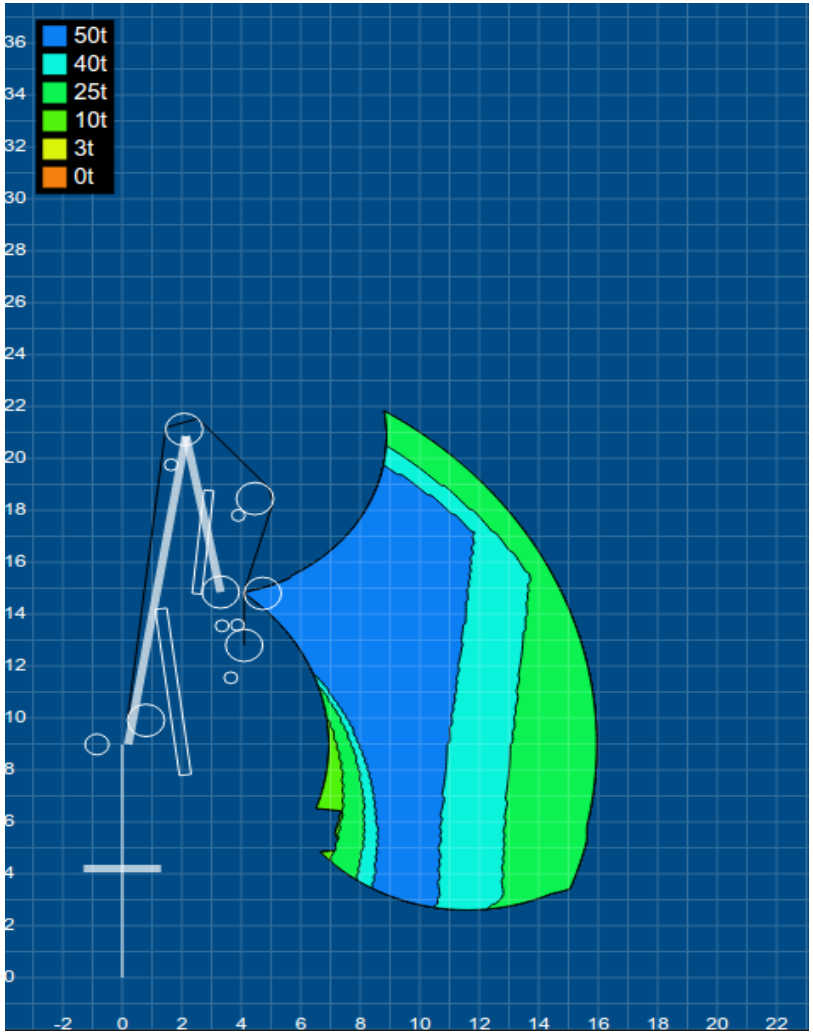
Objective function f_1

- objective: maximise $f_1 = SWL_{max}/W$
- rationale: increase max SWL while reducing weight of crane (weight may serve as surrogate for price of crane)
- note: currently only have an estimate of crane weight
- results:
  - max SWL improved from 100T to 142T (+42.2%)
  - weight of crane reduced from 50.8T to 44.0T (-13.5%)
  - 64.3% improvement of f_1
- load chart: higher SWL values in workspace but size of workspace is reduced

objective: maximise f1	units	nominal	limits (min, max)	optimised	difference	improvement
boomLength	mm	15800	(12000, 26000)	12038	-3762	-23.8%
jibLength	mm	10300	(6000, 16000)	6124	-4176	-40.5%
mainCylinderMaxPressure	bar	315	(100, 400)	383	68	21.6%
jibCylinderMaxPressure	bar	215	(50, 300)	262	47	21.8%
SWL_{max}	kg	99978	N/A	142138	42160	42.2%
W	kg	50856	N/A	44014	-6842	-13.5%
$f_1 = SWL_{max}/W$	-	1.97	N/A	3.23	1.26	64.3%

Load charts of Baku crane and f_1-optimised crane





Other potential objective functions

- Objective function f_2
  - maximise $f_2 = \mathrm{SWL}_{\max} \cdot w_1 + 1/W \cdot w_2$
  - w_1, w_2 are scaling factors (function weights)
  - rationale: maximise max SWL while punishing crane weight
- Objective function f_{3a}
  - minimise $f_{3a} = 1/\mathrm{SWL}_{\max} \cdot w_1 + (W_{\text{target}} - W) \cdot w_2$
  - w_1, w_2 are scaling factors (function weights)
  - W_{target} is set to the weight of the Baku crane
  - rationale: maximise max SWL while punishing deviation from the Baku crane weight
- Objective function f_{3b}
  - minimise $f_{3b} = W \cdot w_1 + (\mathrm{SWL}_{\max,\text{target}} - \mathrm{SWL}_{\max}) \cdot w_2$
  - w_1, w_2 are scaling factors (function weights)
  - $\mathrm{SWL}_{\max,\text{target}}$ is set to the max SWL of the Baku crane
  - rationale: minimise crane weight while punishing deviation from the max SWL of the Baku crane

Future work

- implement more and refine existing objective functions
  - incorporate functions of entire workspace/load chart
  - obtain exact crane weight measures
  - obtain delivery price estimate of crane designs
  - involve Seaonics designers in process for realistic objectives and quality assurance
- develop easy-to-use high-level graphical prototype for end-users without domain expertise in AI, programming, etc.
- develop new back-end simulators for other products
  - plug'n'play with existing AIPO + GA modules
  - current ongoing project for winch design $\Rightarrow$ great synergy with existing crane project
- much more testing and refinements
- publication at ECMS 2016 and later renowned journal

References

* Early work and more details can be found in [4].

[1] G. Gary Wang. Definition and review of virtual prototyping. *Journal of Computing and Information Science in engineering*, 2(3):232–236, 2002.

[2] LA Kamentsky and CN Liu. Computer-automated design of multifont print recognition logic. *IBM Journal of Research and Development*, 7 (1):2–13, 1963.

[3] L I Hatledal, Filippo Sanfilippo, Yingguang Chu, and Houxiang Zhang. A Voxel Based Numerical Method for Computing and Visualizing the Workspace of Offshore Cranes. In *Proceedings of the ASME 2015 34th International Conference on Ocean, Offshore and Arctic Engineering (OMAE2015)*. St. John's, Newfoundland, Canada, 2015. Accepted for publication.

[4] Robin T. Bye, Ottar L. Osen, and Birger Skogeng Pedersen. A computer-automated design tool for intelligent virtual prototyping of offshore cranes. In *Proceedings of the 29th European Conference on Modelling and Simulation (ECMS'15)*, pages 147–156, 2015.

Acknowledgements

The SoftICE lab at NTNU in Ålesund wishes to thank ICD Software AS for their contribution towards the implementation of the simulator, and Seaonics AS for providing documentation and insight into the design and manufacturing process of offshore cranes.
We are also grateful for the support provided by Regionalt Forskningsfond Midt-Norge and the Research Council of Norway through through the VRI projects *Artificial Intelligence for Crane Design (Kunstig intelligens for krandesign (KIK))*, grant no. 241238. and *Artificial Intelligence for Winch Design (Kunstig intelligens for vinsjdesign (KIV))*, grant no. 249171.

Thank you for listening!

Questions?

Supplementary material

Example design calculations I

- hydraulic cylinders key component for “muscle power”
- must avoid buckling = sudden sideways deformation due to compressive stress
- design constraint: buckling load P/max cylinder force $F \geq 2.3$
- design calculations (to show complexity only!):

$$\frac{P}{A} + \left(\frac{P \cdot f_0}{W} + \frac{P \cdot \mu \cdot r}{W} + \frac{m_{\mathrm{Cyl}} \cdot g \cdot L}{W} \right) \cdot \frac{P_E}{P_E - P} \leq \sigma_y \tag{1}$$

Example design calculations II

Simplified, the form of deflection for the cylinder is assumed to be:

$$y(x) = C_1 \cdot \sin\left(\frac{\pi \cdot x}{L}\right) + C_2 \cdot \sin\left(\frac{2 \cdot \pi \cdot x}{L}\right) \tag{2}$$

To find the acceptable load P, the following equations are used (note that some variables such as f_0, AA, BB, etc. are not given in the nomenclature as they are mainly intermediate auxiliary calculations):

$$\alpha = \frac{\pi \cdot L_2}{L} \tag{3}$$

$$AA = \frac{L_1}{2 \cdot I_1} + \frac{L_2}{2 \cdot I_2} + \frac{L}{4 \cdot \pi} \cdot \left(\frac{1}{I_1} - \frac{1}{I_2}\right) \cdot \sin(2 \cdot \alpha) \tag{4}$$

$$BB = \frac{4 \cdot L}{3 \cdot \pi} \cdot \left(\frac{1}{I_2} - \frac{1}{I_1}\right) \cdot \sin^3(\alpha) \tag{5}$$

Example design calculations III

$$CC = \frac{L_1}{2 \cdot l_1} + \frac{L_2}{2 \cdot l_2} + \frac{L}{8 \cdot \pi} \cdot \left(\frac{1}{l_1} - \frac{1}{l_2} \right) \cdot \sin(4 \cdot \alpha) \tag{6}$$

$$DD = \frac{\pi^2 \cdot E}{2 \cdot L} \tag{7}$$

$$a = 4 \cdot AA \cdot CC - BB^2 \tag{8}$$

$$b = -4 \cdot DD \cdot (4 \cdot AA + CC) \tag{9}$$

$$c = 16 \cdot DD^2 \tag{10}$$

$$P_E = \frac{-b - \sqrt{b^2 - 4 \cdot a \cdot c}}{2 \cdot a} \tag{11}$$

Example design calculations IV

$$f_0 = \frac{L_1 \cdot \blacksquare}{L \cdot L_3 \cdot 2} \cdot \sqrt{\frac{\pi^2 \cdot E \cdot I_2}{P_E}} \tag{12}$$

$$FF = \frac{d}{2 \cdot I_2} \cdot (\mu \cdot r + f_0) \tag{13}$$

$$GG = \frac{m_{\mathrm{Cyl}} \cdot g \cdot L \cdot d}{16 \cdot I_2} \tag{14}$$

$$HH = \left[1 + 2 \cdot A \cdot FF - \frac{2 \cdot \sigma_y \cdot A}{P_E} + A^2 \cdot FF^2 + \frac{2 \cdot A^2 \cdot FF \cdot \sigma_y}{P_E} + \left(\frac{\sigma_y \cdot A}{P_E}\right)^2 + \frac{4 \cdot A \cdot GG}{P_E}\right]^{\frac{1}{2}} \tag{15}$$

We can then find P as

$$P = \frac{\sigma_y \cdot A}{2} + \frac{P_E}{2} \cdot (1 + A \cdot FF - HH) \tag{16}$$

Example design calculations V

Finally, the safety factor P/F against buckling must be at least 2.3 as given by the following formula:

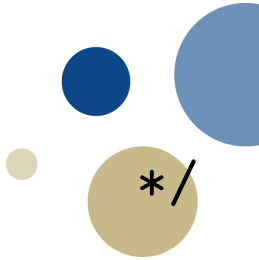
$$\frac{P}{F} \geq 2.3 \tag{17}$$

Design choice must satisfy constraint: Given some desired lifting capability F, must choose parameters such that P becomes large enough to satisfy safety factor!

Pseudo-code for a basic GA I

```
/* INITIALISATION                                                    */
```
define encoding scheme for chromosomes c;
define cost function $f(c)$;
set criteria for selection, crossover, mutation, elitism;
generate initial *population* of chromosomes;
sort *population* in increasing order of cost;
bestChrom $\leftarrow$ *population*[0];
set *minCost*, *maxIterations*;
$i \leftarrow 1$;

Pseudo-code for a basic GA II



```
/* LOOP
while i < maxIterations OR bestCost > minCost do
    evaluate cost for each chromosome;
    select chromosomes for mating;
    perform mating, crossover, mutation, elitism;
    update population;
    sort population in increasing order of cost;
    bestChrom ← population[0];
    bestCost ← f(bestChrom);
    i ← i + 1;
end
return i, bestChrom, bestCost;
decode bestChrom to original domain;
```