# RTC-Tools 2.0: Waar staan we nu?

Jorn Baayen

17 maart 2017

# Outline

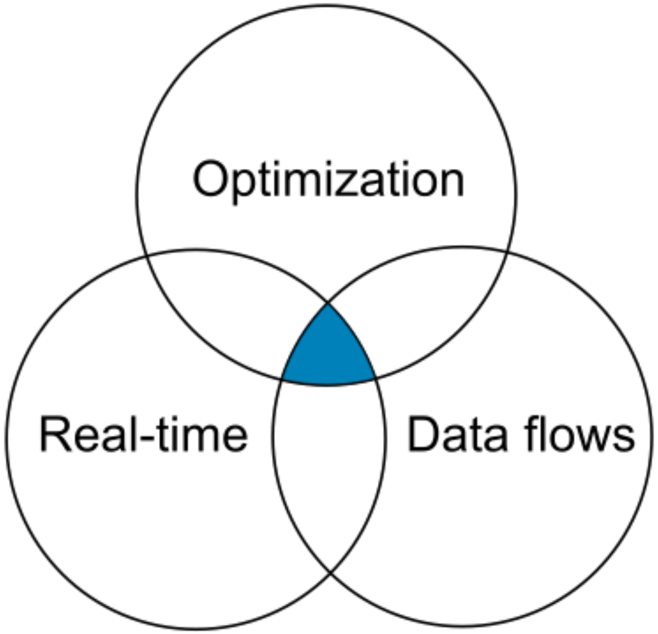
- RTC-Tools: Scope, history
- Convex optimization
- Hydraulic modelling
- Multiple objectives
- Mixed integer optimization

# RTC-Tools: Scope

RTC-Tools is the Deltares toolbox for control and optimization of environmental systems.

Delft-FEWS is an open data handling platform, used for the aggregation of (real-time) environmental data flows.

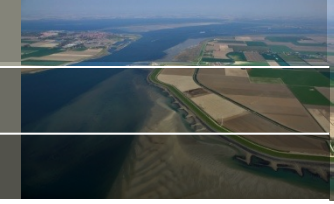
Together, they provide a platform for the development of decision support systems.




Deltares

# History

- 2005: Reservoir module for Delft-FEWS.
- 2012: Dirk Schwanenberg releases first version of RTC-Tools source code to the public. RTC-Tools 1.x connected non-linear hydraulic and reservoir models to the IPOPT optimizer.
  - Promising results, many scientific publications
  - High interest from reservoir operators
  - But challenging to operationalize, and hard to extend
- 2015: Work starts on new mathematically rigorous foundation, initially as an experiment of Jorn Baayen and Matthijs den Toom.
- 2016: First pilot project on new foundation. Peter Gijsbers develops water allocation tool for Rijkswaterstaat using new framework.
- 2016: RTC-Tools 2.0 released.

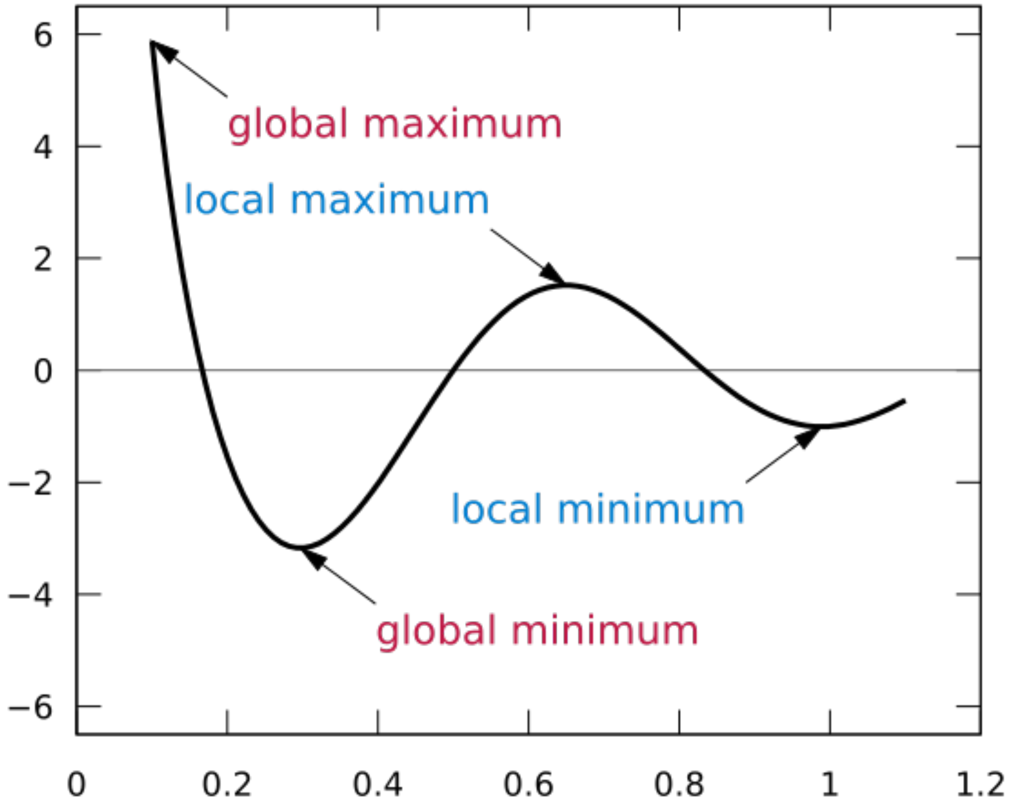



# Reliability axioms

A decision support system that is used day in, day out needs to be reliable. This need can be made precise with six axioms:

- *Robustness*: The solutions are robust in the sense that system constraints are satisfied regardless of the differences between model and reality.
- *Feasibility*: A feasible solution always exists.
- *Quality*: Any solution is a “good” solution.
- *Stability*: The solutions are stable in the sense that small perturbations in the configuration result in small changes in the solution.
- *Determinism*: Given the same initial solution guess and configuration, the solution is always identical.
- *Bounded solution time*: A solution is found within a predetermined amount of time.

# Local and global optima




Source: Wikipedia. GFDL 1.2.

Deltares

# From axioms to convexity

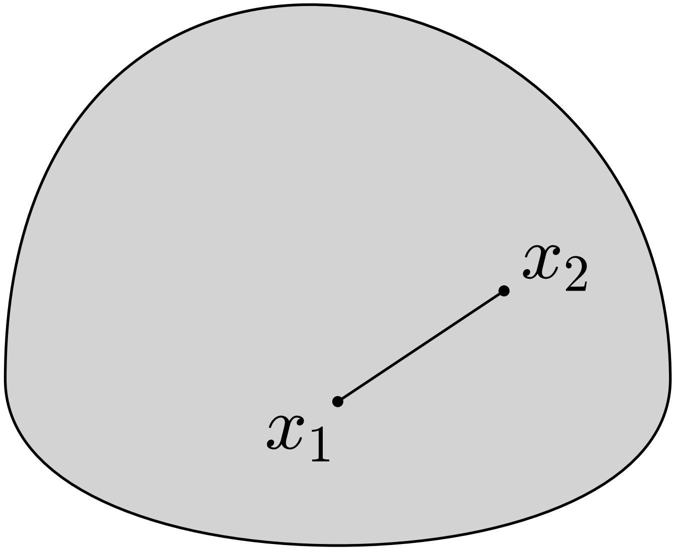
Suppose we had an optimization problem that would only have globally optimal solutions.

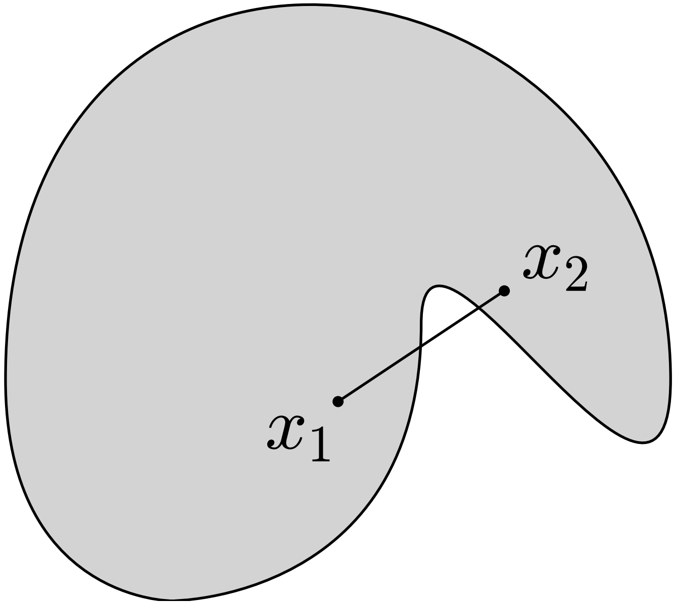
That would give us:

- *Quality*: Every solution is a globally optimal solution.
- *Stability*: Changing seed solutions or optimizer settings won't change the quality of the end solution.

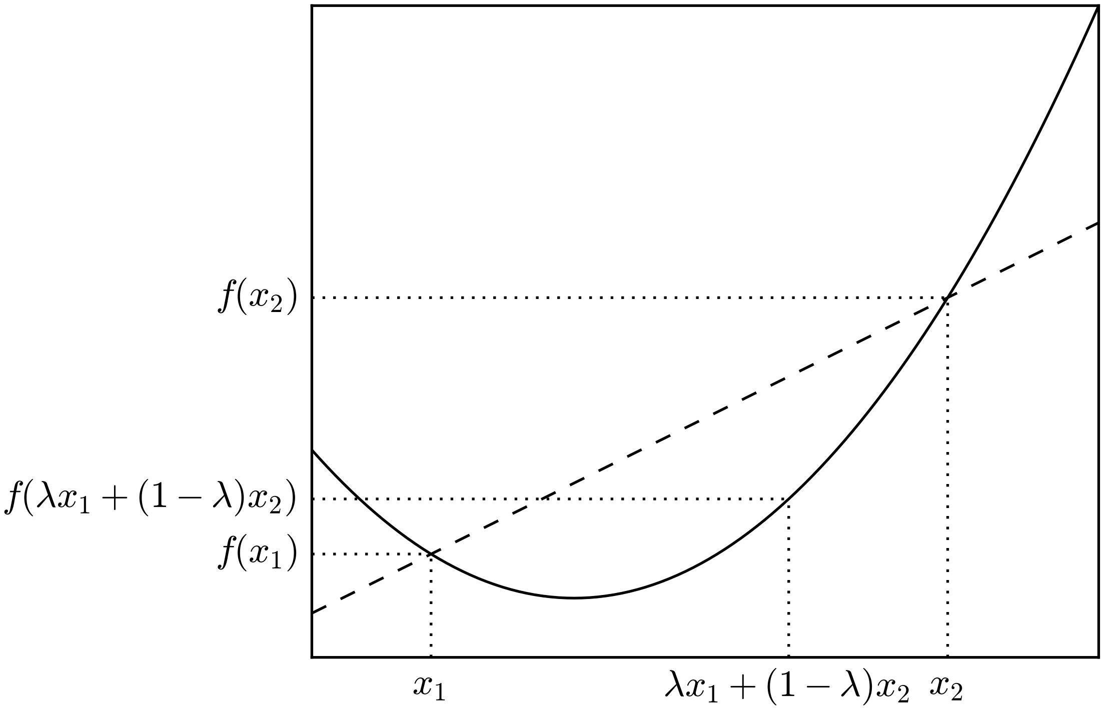
So-called *convex* optimization problems only admit globally optimal solutions. Convex problems can be solved efficiently using deterministic methods.

# Convex sets



# Convex functions




$$f(\lambda x_1 + (1-\lambda)x_2) \leq \lambda f(x_1) + (1-\lambda) f(x_2)$$

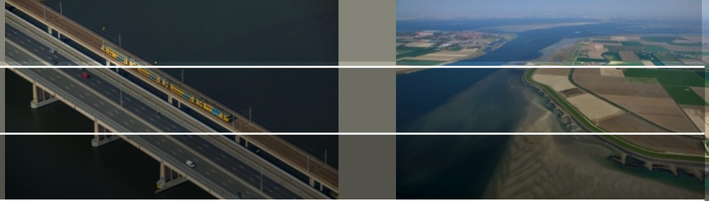
# Convex optimization

$$\min f(x) \quad \text{subject to}$$
$$g(x) \leq 0$$
$$h(x) = 0$$

Problem is called convex when:

- $f$ is a convex function
- $g$ is a convex function
- $h$ is an affine function:
  - $h(x) = 0 \Leftrightarrow h(x) \leq 0$ and $-h(x) \leq 0$ .
  - $h$ must be both convex and concave, i.e., affine: $h(x) = ax + b$.
  - **This is quite restrictive**

Convex problems only admit global optima.

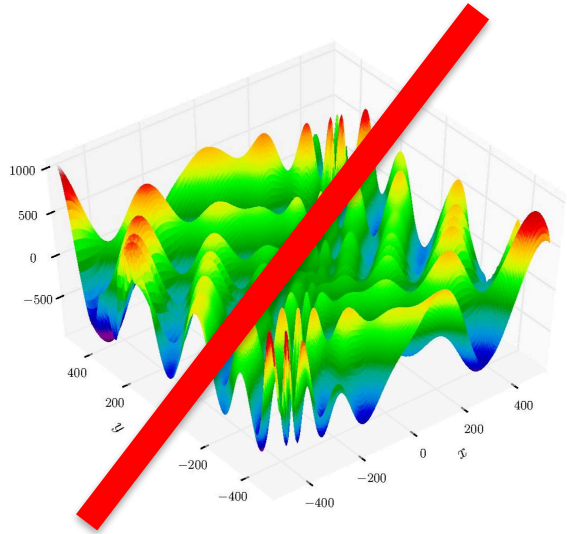


# Hydraulic modelling

- Highly nonlinear friction term in diffusive wave equation:

$$\frac{\partial H}{\partial x} + \frac{C}{R} Q^2 = 0$$

- When using many diffusive wave branches, large numbers of local minima are created. What to do?
- Linearization results in large errors; piecewise linearization results in large numbers of integer variables.




Deltares

# Homotopy

Idea: **Interpolate between linearized and non-linear model.**

$$M = (1-\theta)M_L + \theta M_{NL}$$

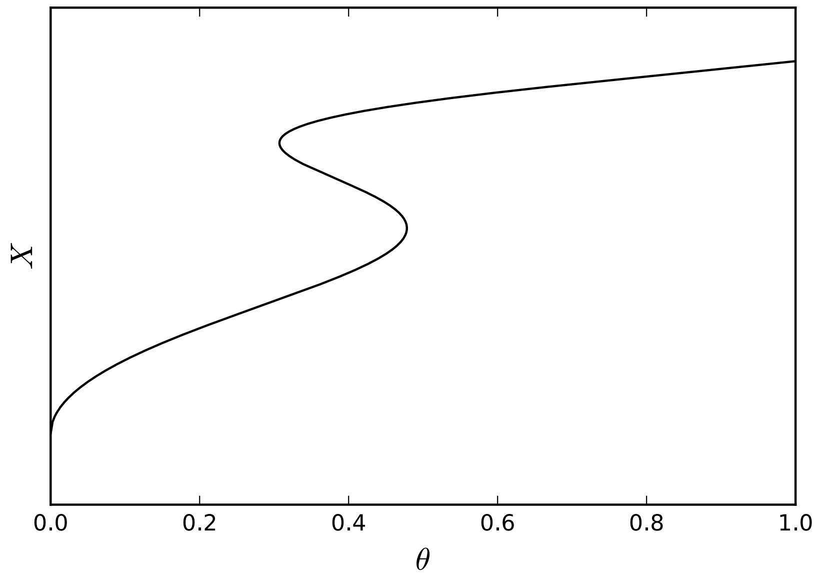
With $\theta \in [0, 1]$.

.

# Continuation method

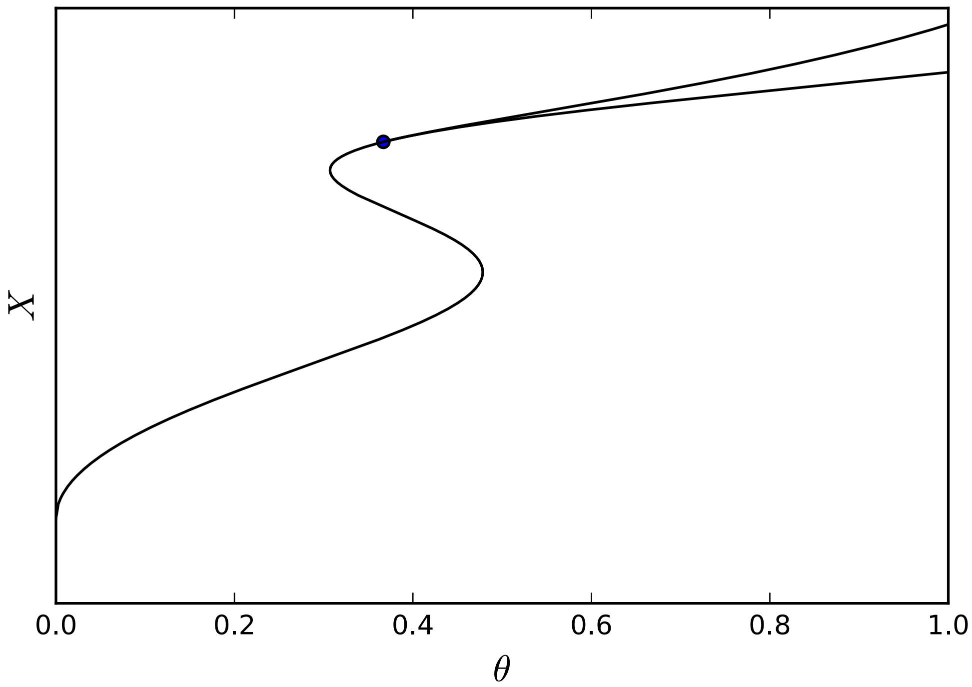
Start solving at $\theta = 0$ continuing step–by-step until $\theta = 1$.

At $\theta = 0$, the linearized problem only has global optima. We find a global optimum, and trace this through to a nonlinear solution at $\theta = 1$: Solution to nonlinear problem can be traced back to globally optimal solution of linearized problem.

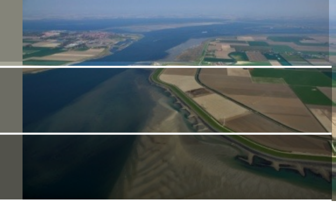


# Continuation method

Ongoing research challenge: Bifurcations



# Multi-objective optimization



Suppose we have the following goals:

- Keep water levels within bounds as much as possible
- Maintain minimum spill flows for fish migration, if possible
- Apply best effort to track the generation request

Let $\{f_i : i \in I\}$ denote the set of functions encoding these goals. We have:

$$\min f_i \ \ \forall i \in I \text{ subject to}$$
$$g(x) \leq 0$$
$$h(x) = 0$$

How to solve this?

Deltares

# Pareto optimality

A solution $x^*$ of the problem

$$\min f_i \;\; \forall i \in I \text{ subject to}$$
$$g(x) \leq 0$$
$$h(x) = 0$$

Is *Pareto-optimal* if there is no $x^{**}$ such that for a $j$

$$f_j(x^{**}) < f_j(x^{**})$$

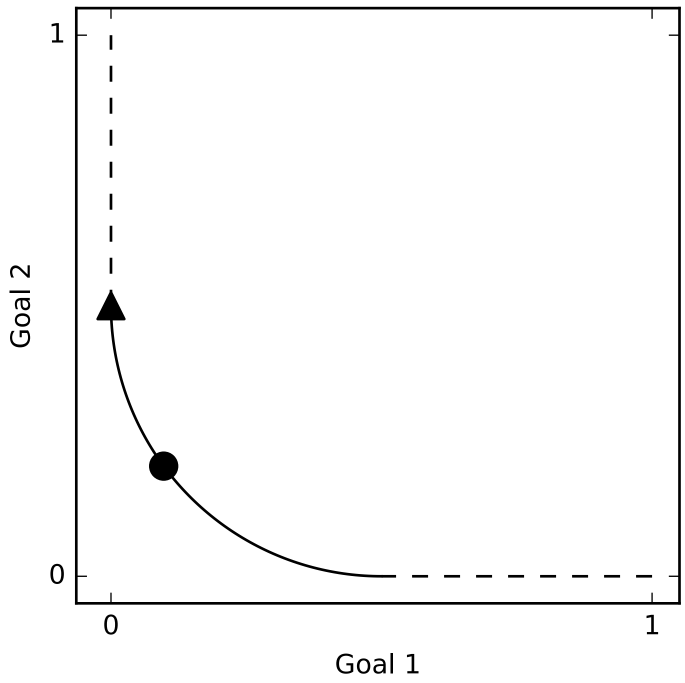
and for all $i \neq j$

$$f_i(x^{**}) \leq f_i(x^{**})$$

In words: Pareto optimality implies that no goal can be improved without making another one worse.

# Pareto front

The *Pareto front* is the set of all Pareto-optimal solutions.



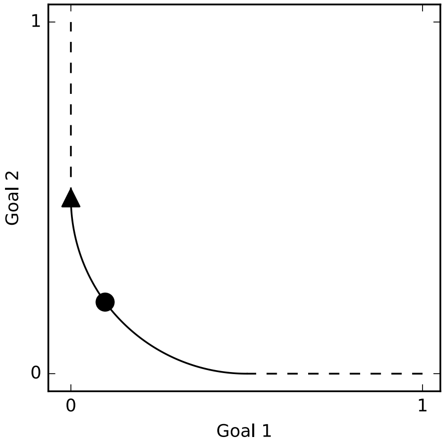
# Weighting method

The weighting method transforms the multi-objective problem to the scalar problem

$$\min \sum_i \lambda_i f_i \text{ subject to}$$
$$g(x) \leq 0$$
$$h(x) = 0$$

- Problem: How to pick the weighting factors $\lambda_i$.
- And if the weighting factors are arbitrary to a degree, then so is the solution!

Solution on Pareto front shown with a circle.



# Lexicographic goal programming

In lexicographic goal programming, we transform the multi-objective problem to a sequence of scalar optimization problems.

First, we order our goals. For example:

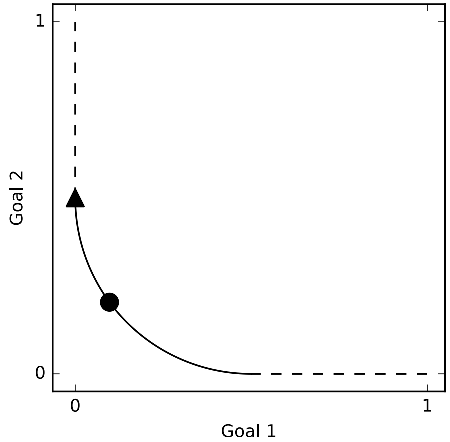
1. Keep water levels within bounds as much as possible
2. Maintain minimum spill flows for fish migration, if possible
3. Apply best effort to track the generation request

# Lexicographic goal programming

The <u>idea</u> of the algorithm is:

1. Minimize $f_1$ to yield a minimum objective value of $\varepsilon_1$.
2. Minimize $f_2$ to yield $\varepsilon_2$ subject to the additional constraints
   - $f_1(x) = \varepsilon_1$
3. Minimize $f_3$ subject to the additional constraints
   - $f_1(x) = \varepsilon_1$
   - $f_2(x) = \varepsilon_2$
4. …

Solution on Pareto front shown with an arrow.




Deltares

# Mixed integer: Minimum on-time pump

$$0 \le Q_t \le Q_{max}\delta_t$$

$$\delta_t - \delta_{t-1} \le 2(1 - x_t) - 1$$

$$\delta_t - \delta_{t-1} \ge -x_t$$

$$\delta_{t-1} - \delta_t \le 2(1 - y_t) - 1$$

$$\delta_{t-1} - \delta_t \ge -y_t$$

$$\boldsymbol{\sum_{i\in\{t,\ldots,t+N-1\}}(x_i+y_i) \le 1}$$

$$\delta_t, x_t, y_t \in \{0, 1\}$$

| | $\delta_{t-1}$ | $\delta_t$ | $\delta_t - \delta_{t-1}$ | $x_i$ | $y_i$ |
|---|---|---|---|---|---|
| Pump stays off | 0 | 0 | 0 | 0 | 0 |
| Pump switched on | 0 | 1 | 1 | 1 | 0 |
| Pump switched off | 1 | 0 | -1 | 0 | 1 |
| Pump stays on | 1 | 1 | 0 | 0 | 0 |

Idea: Encoding logical tables using linear constraints.