# NEWS IN SHORT-TERM SCHEDULING

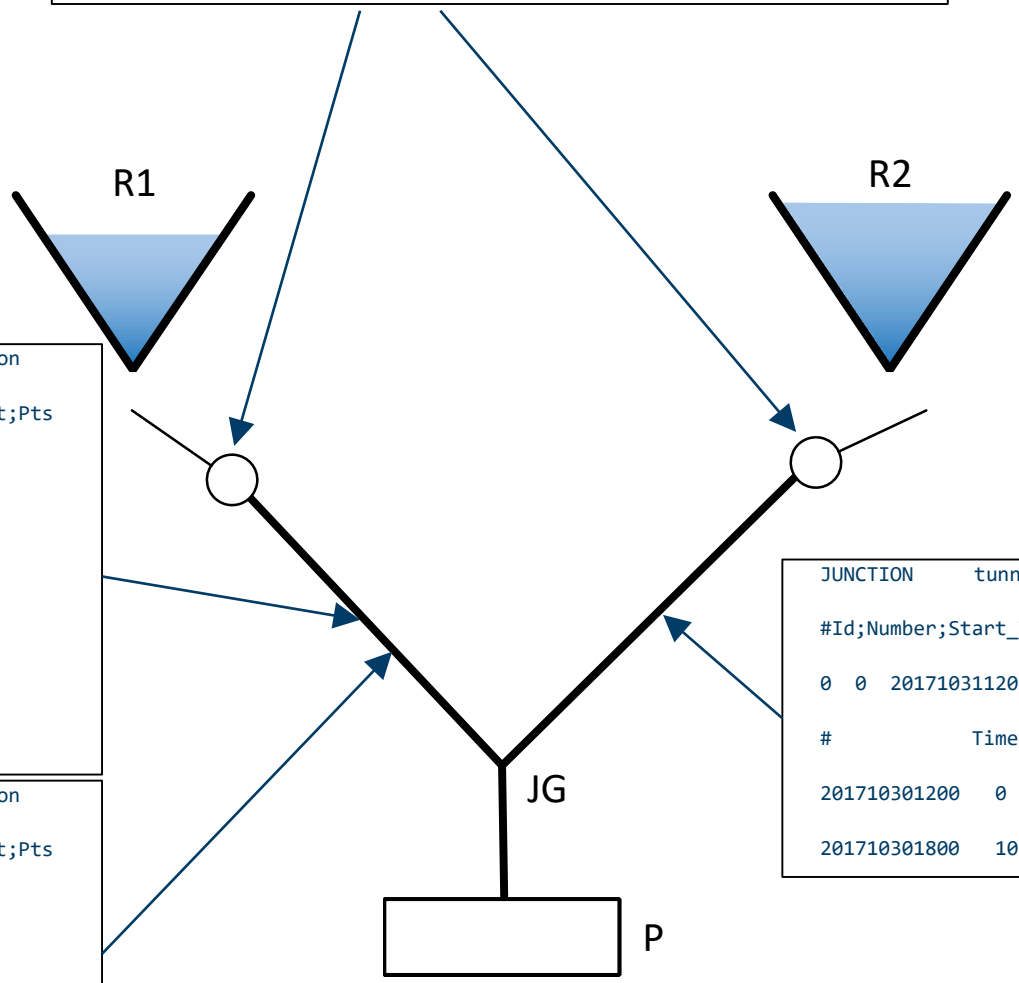
Hans Ivar Skjelbred, SINTEF Energi

Production planning User meeting

Hell, March 14, 2019

# Junction gate optimization

- JUNCTION tunnel_1_close_time MyJunction 360
- JUNCTION tunnel_2_close_time MyJunction 180
- JUNCTION tunnel_1_open_time MyJunction 120
- JUNCTION tunnel_2_open_time MyJunction 240




```
JUNCTION     tunnel_1_optimization_flag       MyJunction
#Id;Number;Start_Time;Time_unit;Period;Data_type;Y_unit;Pts
0  0  201710311200  MINUTE  525600  -1  NO_UNIT  4
#            Time;      f(t)
201710301200   0  #Closed
201710301800   1  #Open
201710310000   2  #Either fully open or closed
201710310600   3  #Throttling allowed
```

```
JUNCTION     tunnel_1_switch_cost             MyJunction
#Id;Number;Start_Time;Time_unit;Period;Data_type;Y_unit;Pts
0  0  201710311200  MINUTE  525600  -1  NOK  2
#            Time;      f(t)
201710301200   2000
201710301800   8000
```

```
JUNCTION     tunnel_2_min_flow  MyJunction
#Id;Number;Start_Time;Time_unit;Period;Data_type;Y_unit;Pts
0  0  201710311200  MINUTE  525600  -1  M3/S  2
#            Time;      f(t)
201710301200   0
201710301800   10
```

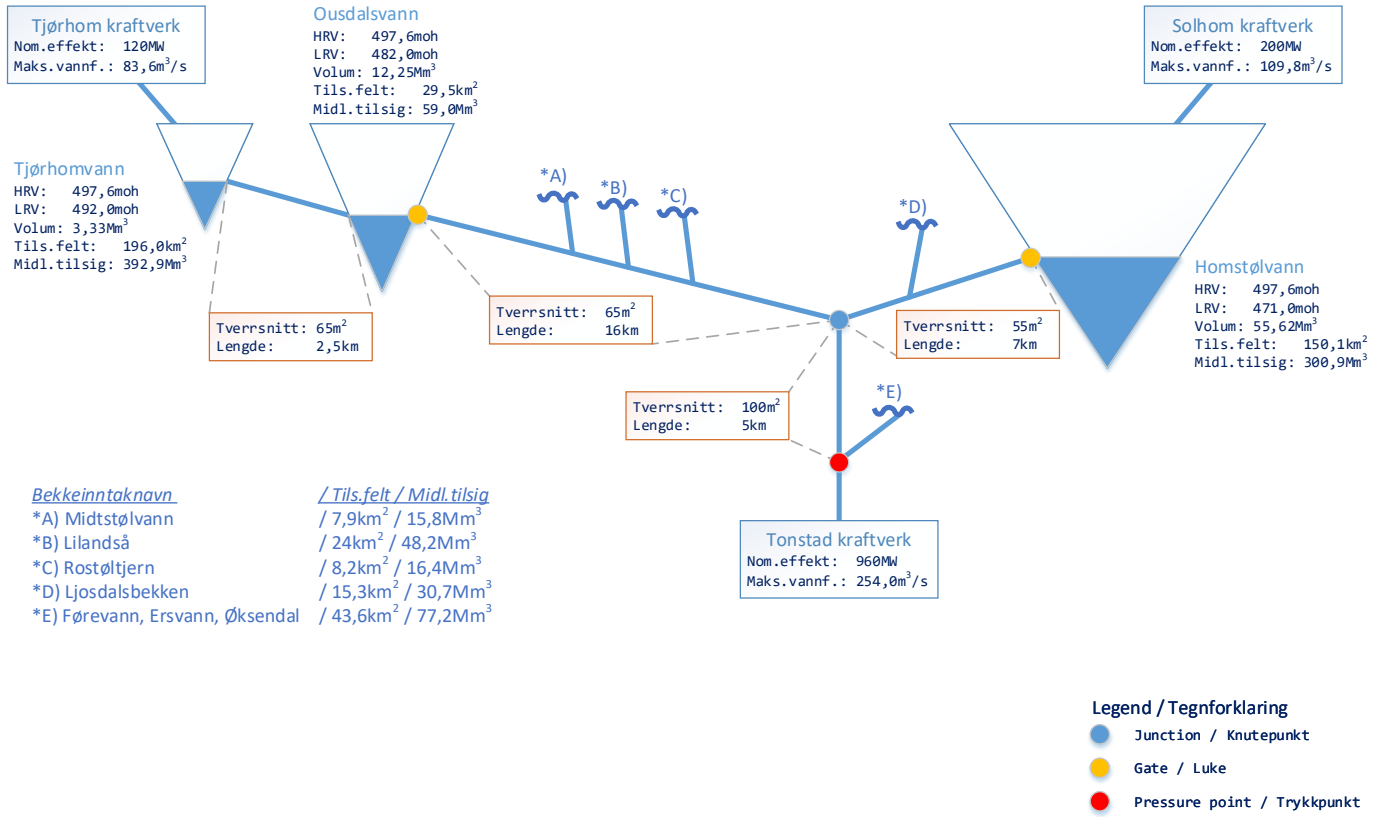


SINTEF

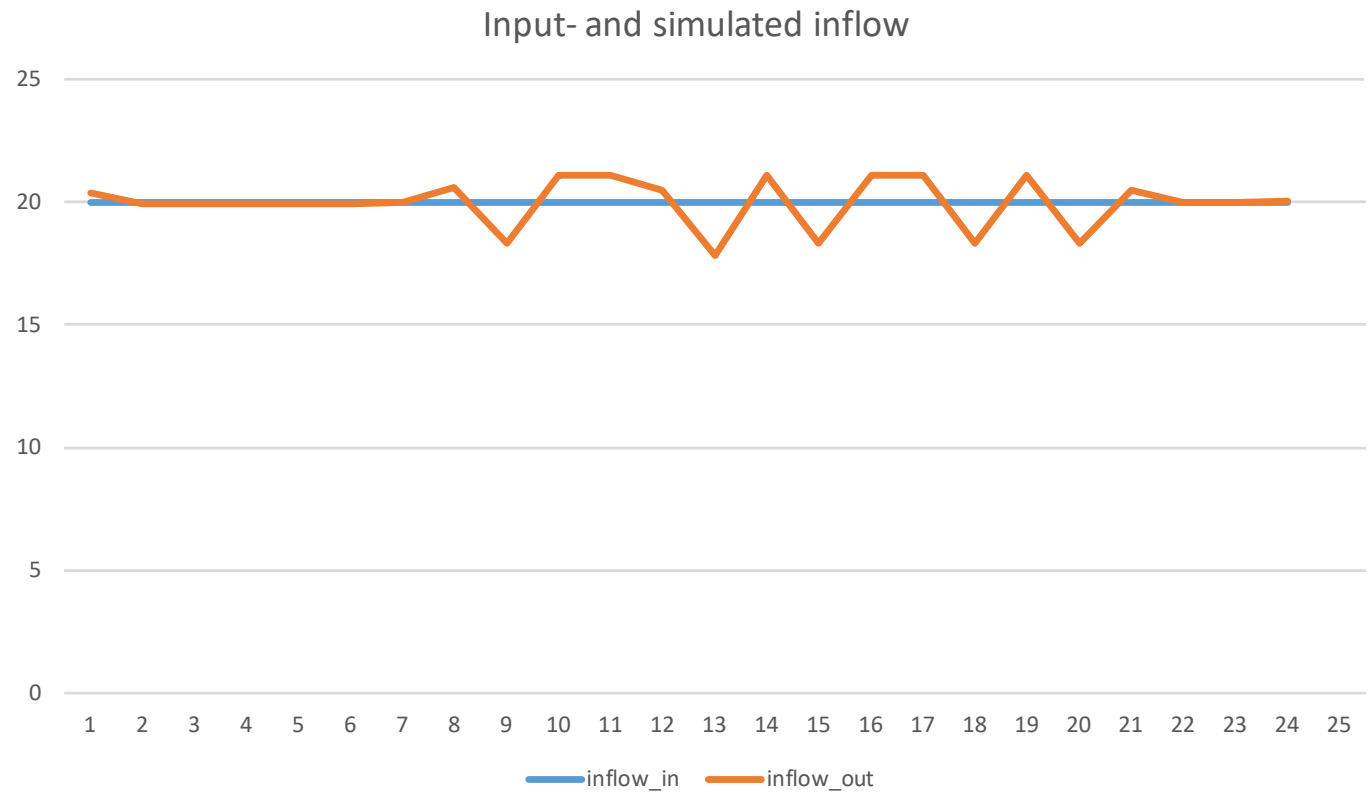
# Case study – Sira Kvina

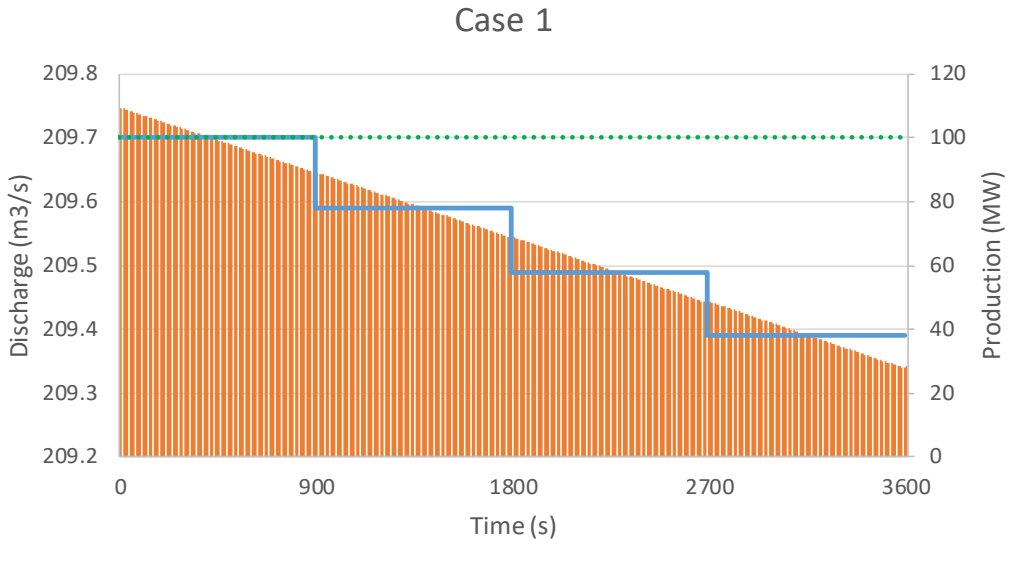
- Knutepunkt Ousdalsvann – Homstølvann er usymmetrisk både mht magasinvolum, kapasitet og tunnellengde.
- Høydeforskjell mellom Tjørhomvann og Homstøl på 3-6 meter ved normalt tilsig.
- For å trekke nok vann fra Ousdalsvann ved høyt tilsig må Homstølvann senkes for fortsatt kunne kjøre Tjørhom kraftverk uten overløp i Tjørhomvann.
- Ved stort eller overraskende tilsig er det ikke nok i senke Homstølvann. Da kan luken til Homstølvann lukkes helt eller delvis.

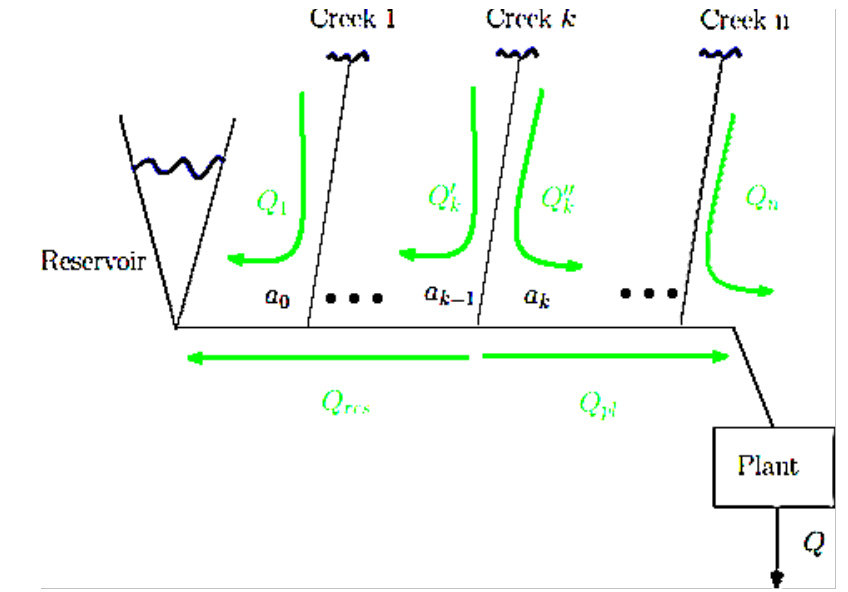
## Case – Junction gate



# Water budgeting with SHOP-SIM






SINTEF

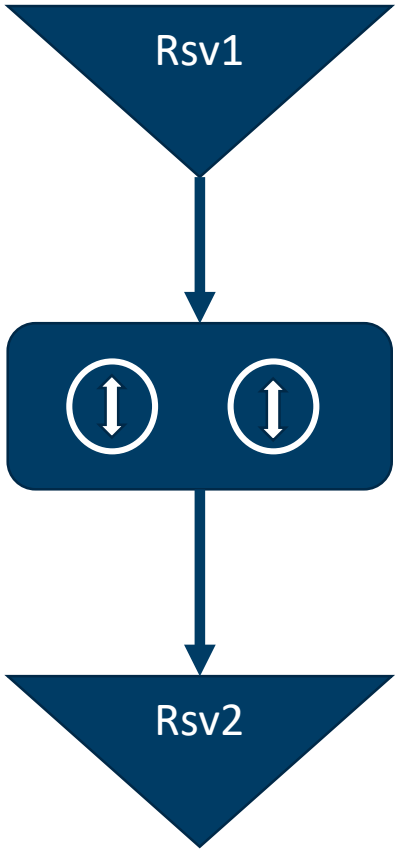
# Head optimization

Gen: $p_{out} = \eta(h,q) \cdot g \cdot h(v,q) \cdot q$

Pump: $p_{in} = 1/\eta(h,q) \cdot g \cdot h(v,q) \cdot q$

$$p_{min} \leq p \leq p_{max}$$

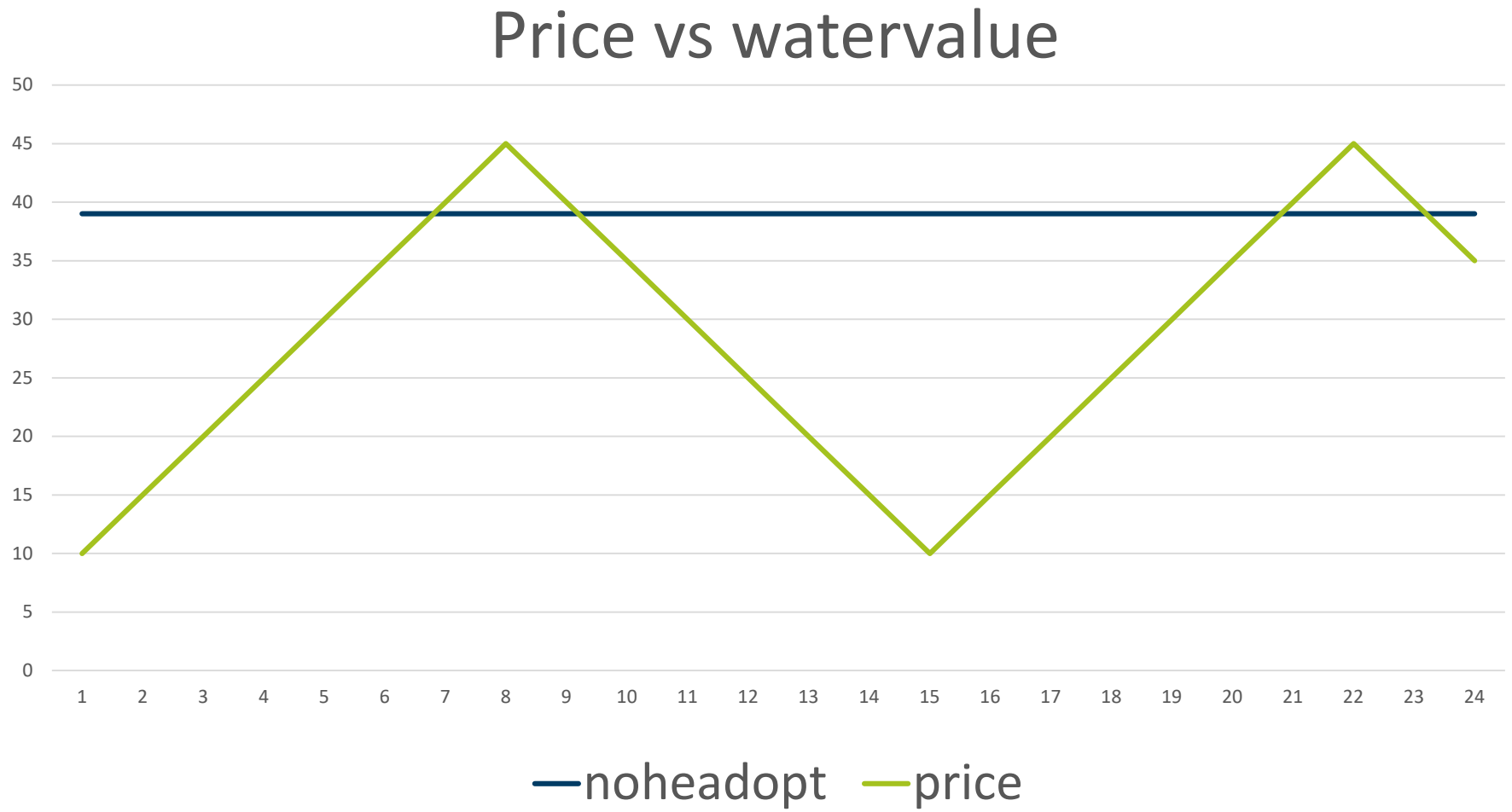
$$q_{min}(h(v,q)) \leq q \leq q_{max}(h(v,q))$$

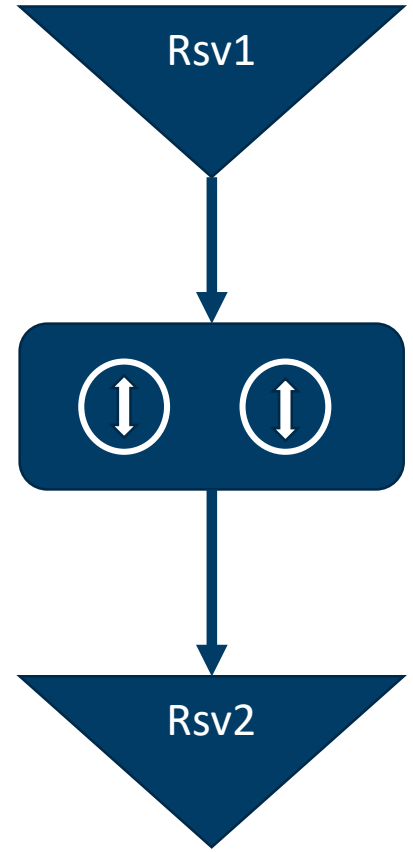





SINTEF

# Head optimization for pumps

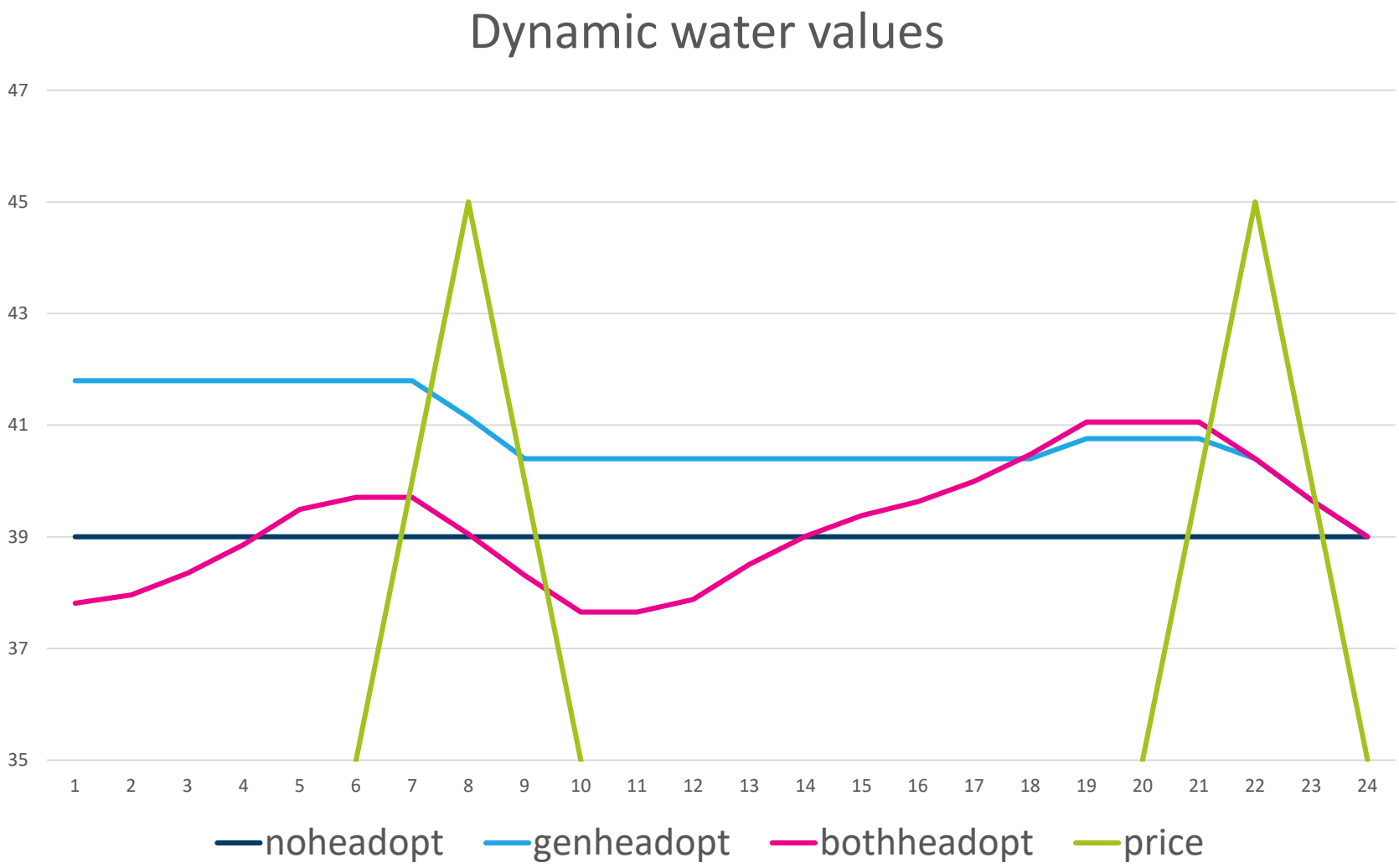


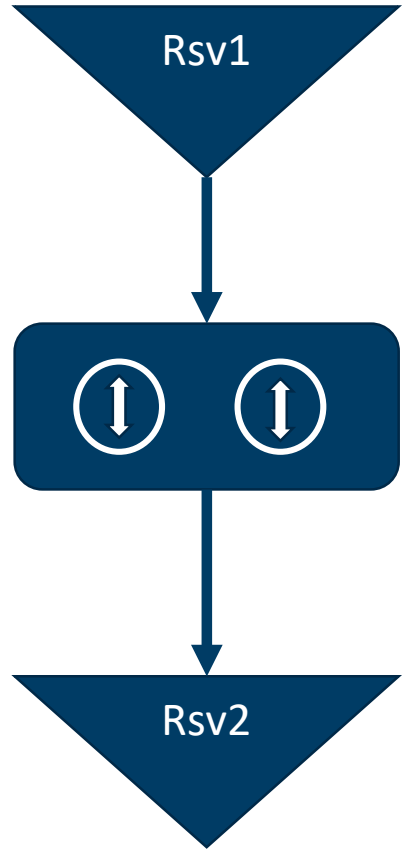






SINTEF

# Head optimization for pumps

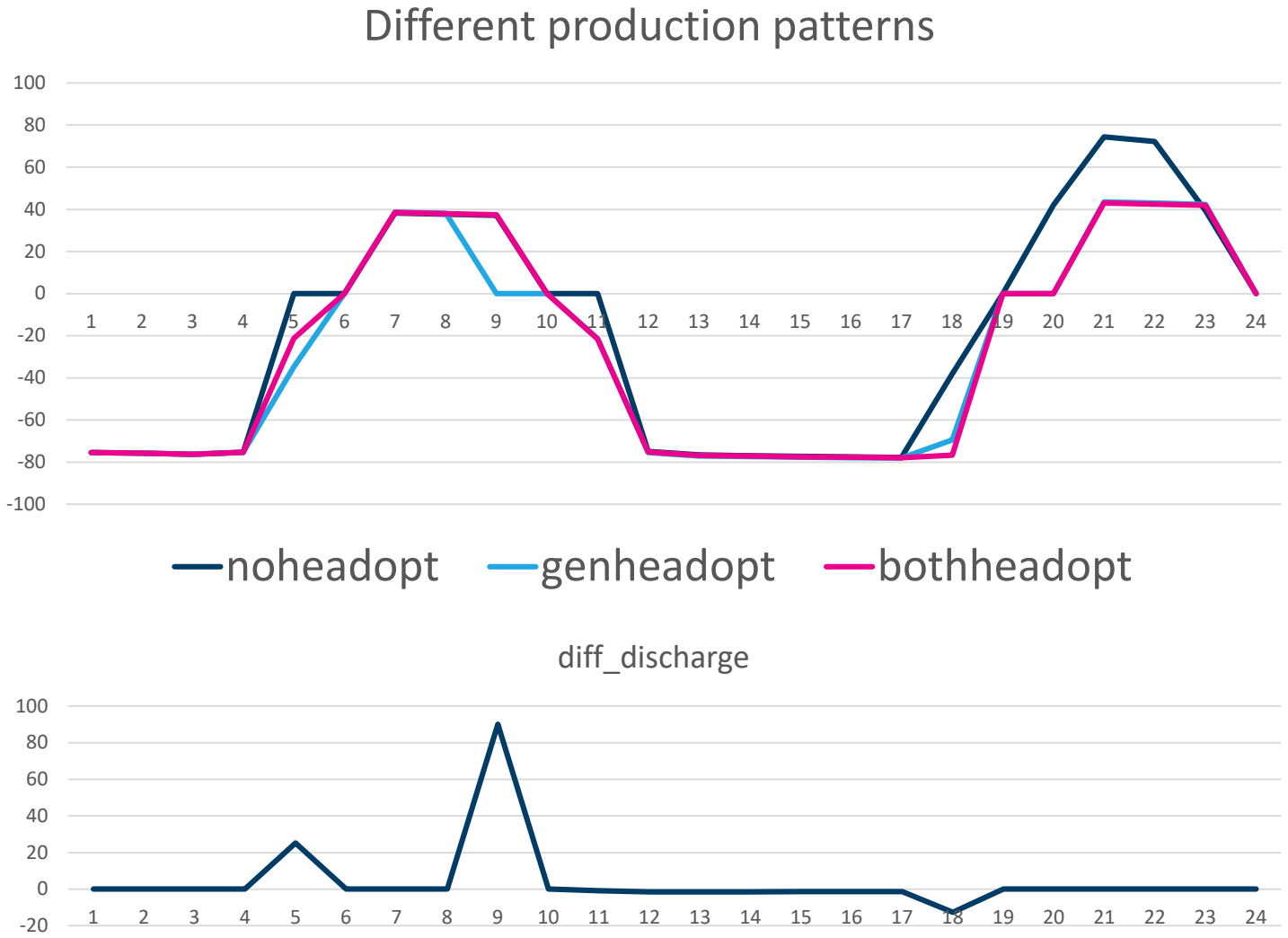


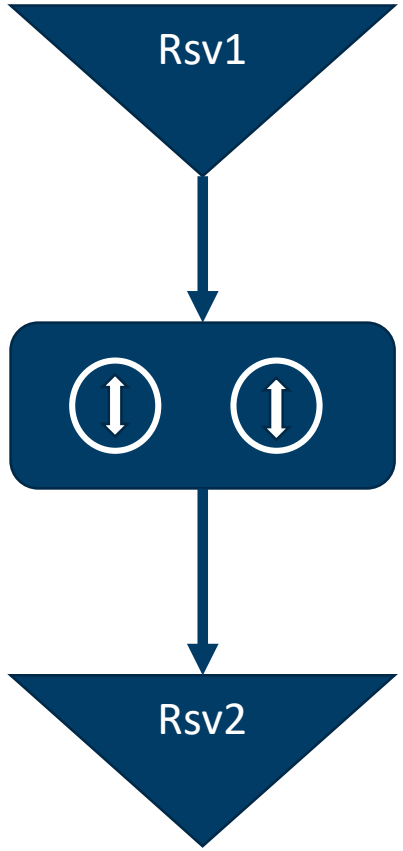






SINTEF

# Head Optimization – Production comparison

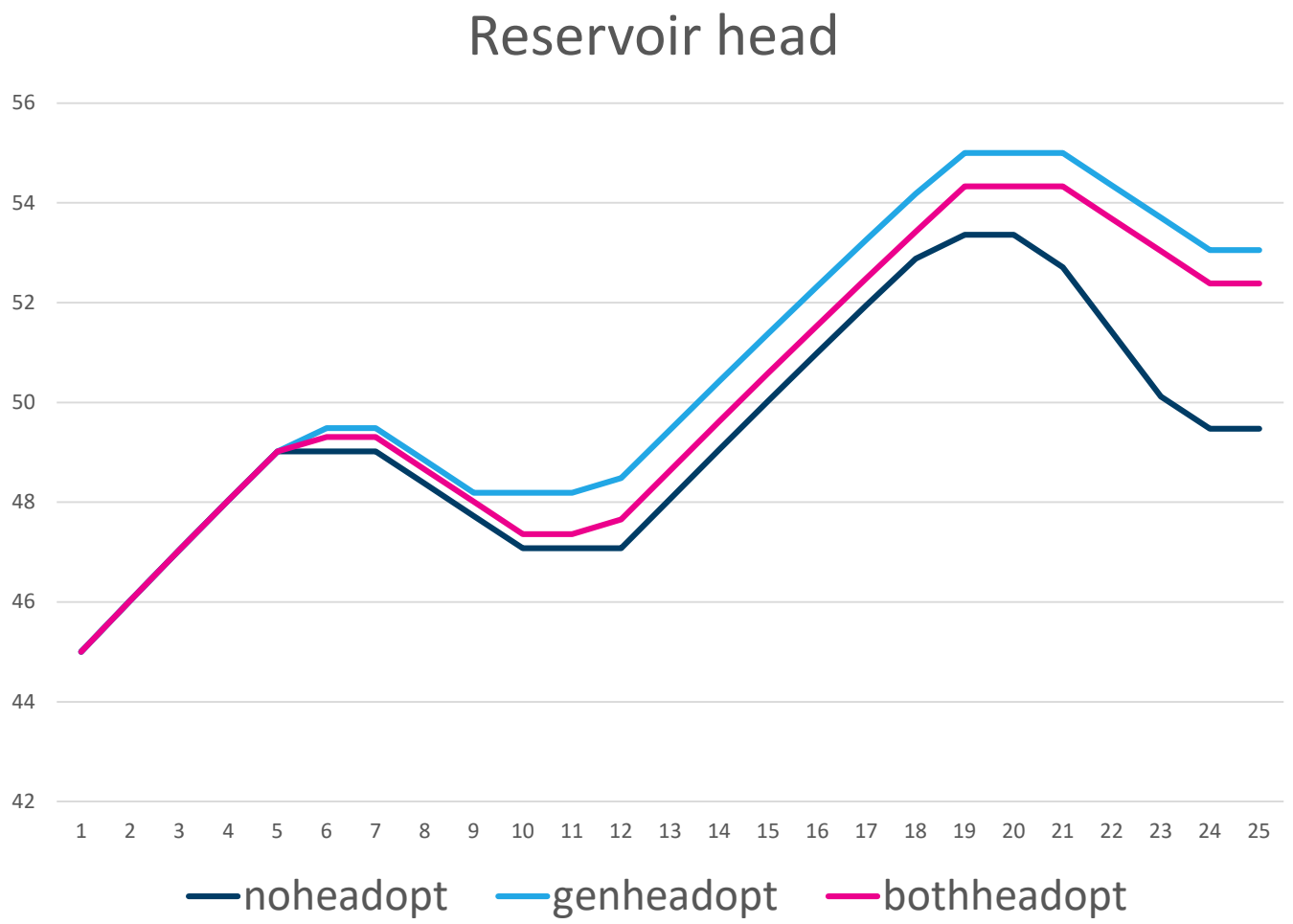


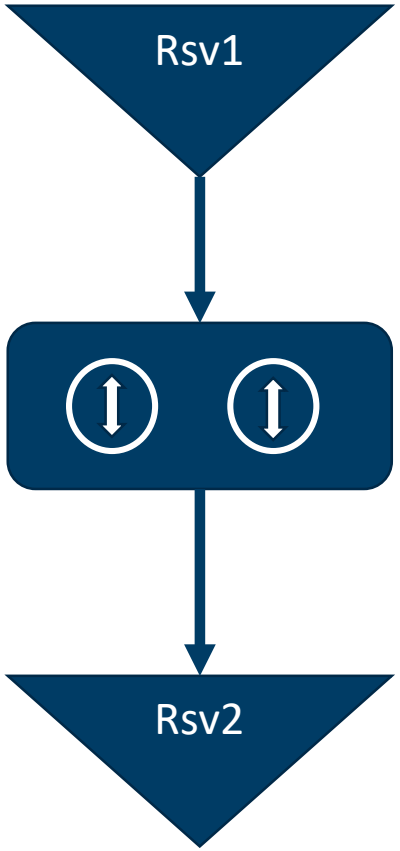





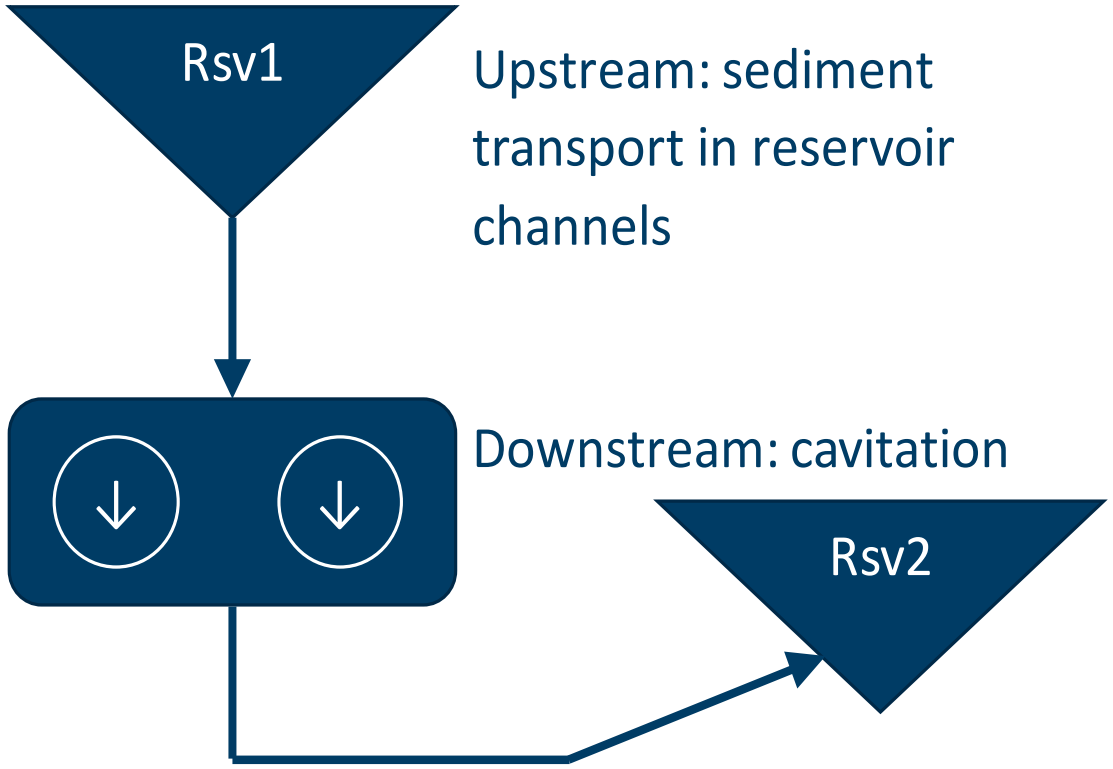

SINTEF

# Head Optimization – Reservoir trajectory comparison

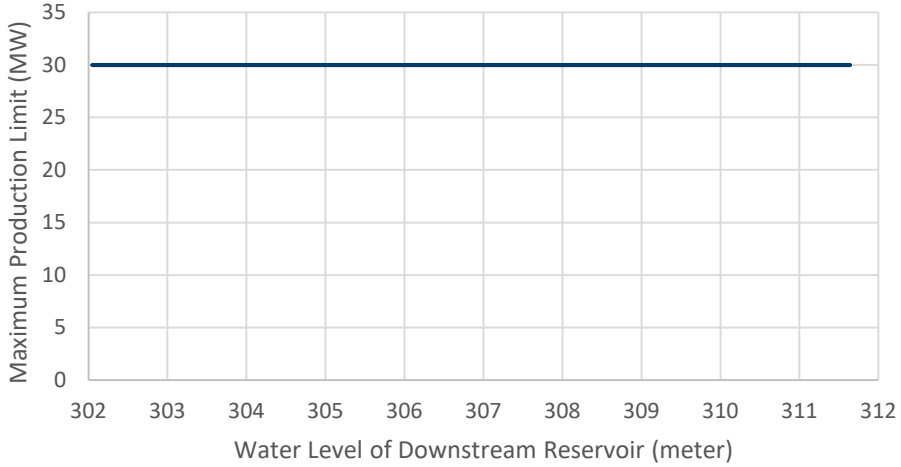


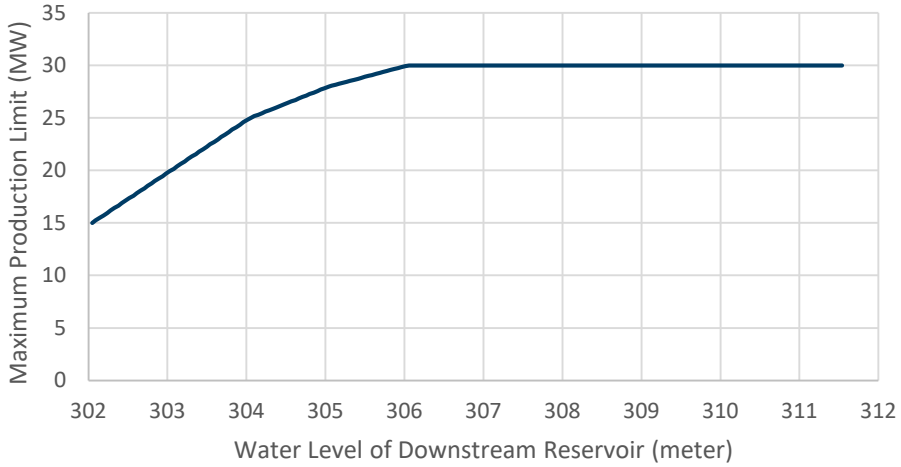
# Plant discharge limits depending on water level of reservoirs



Without reservoir level production limits




With reservoir level production limits (linear)




GENERATOR **p_max_limit_rsv_down** Junsterforsen 1

#Id; Number; Reference; Pts; X_unit; Y_unit

42325 0 0 4 meter MW

#x_value; y_value;

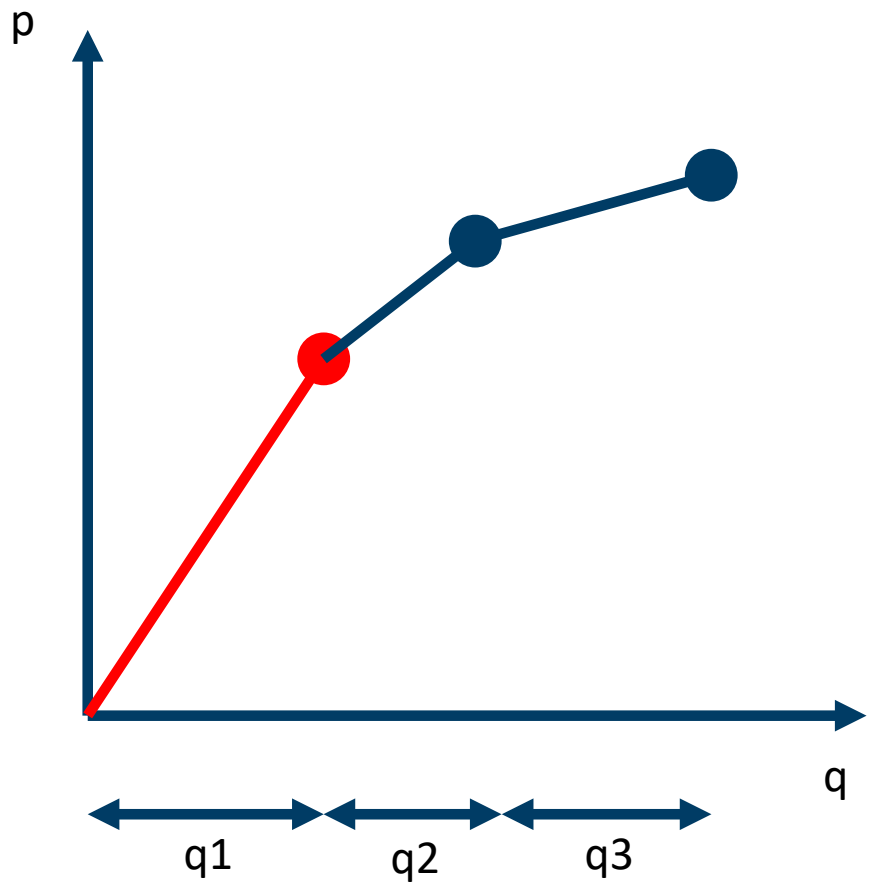
303 20

304 25

305 28

306 30

# Discharge cost below best point - Hydrocen

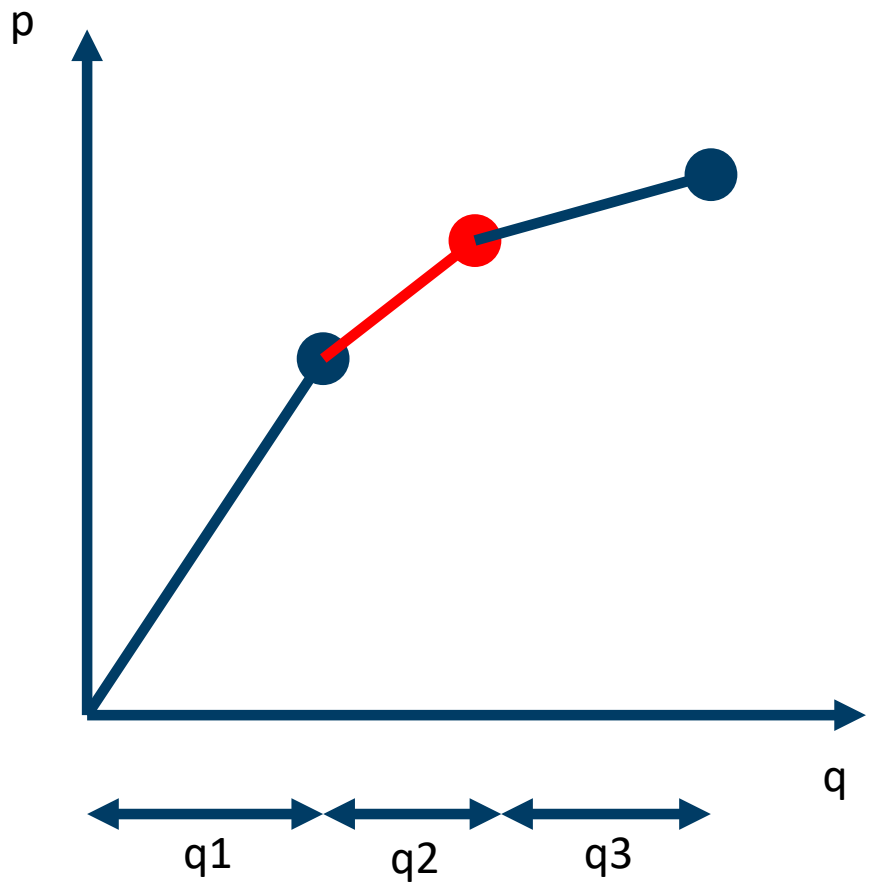





SINTEF

# Discharge cost below best point in SHOP

- New situation: segment 1 (q1) has an additional discharge cost due to cavitation, and the optimization chooses to use segment 2 (q2) before segment 1. This is not correct, since it is not the first part of the p/q-curve.






SINTEF

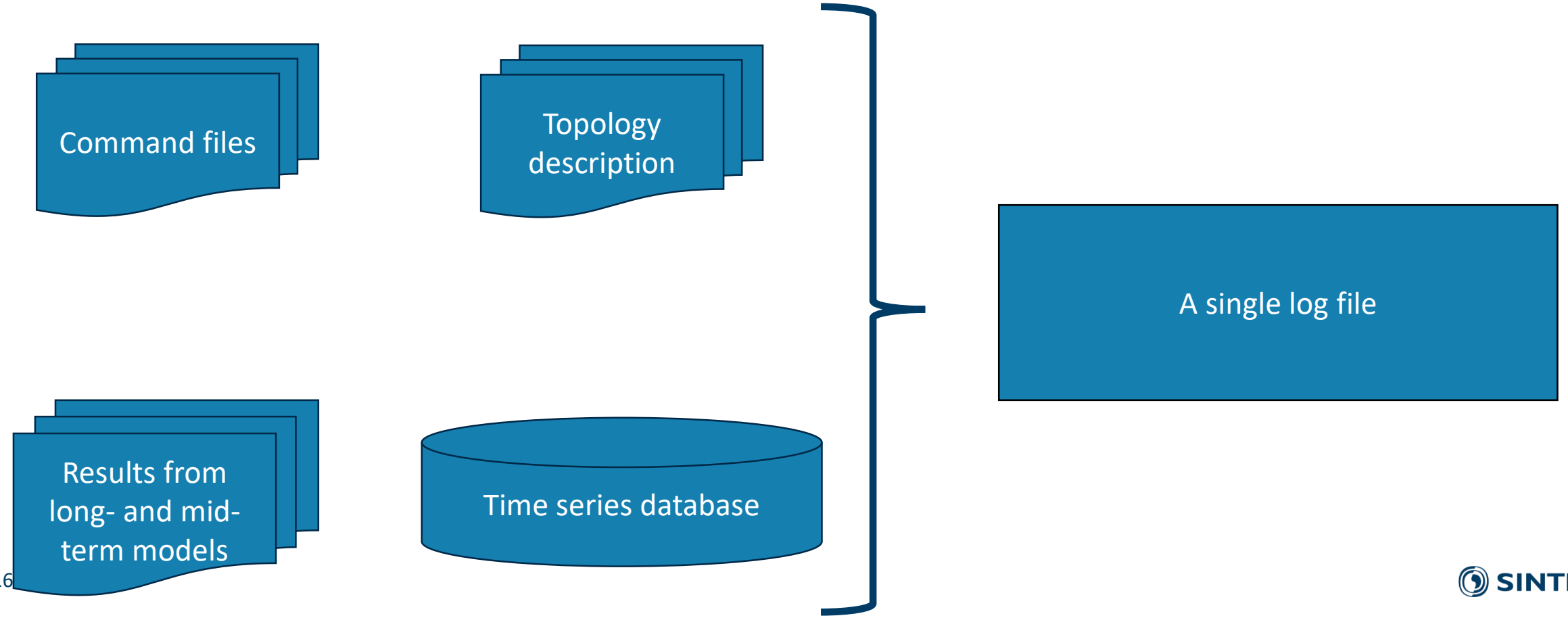
# Discharge cost below best point in SHOP

- Tested possible solution: model segments 1, 2 and 3 as different production zones on the same generator. Binary variables are added automatically by SHOP to make sure the zones follow the correct order.
- Problem found when testing this: The discharge cost curve uses its own set of segments, independent of the production zones. Although the zones are used correctly, the discharge cost segments are not.

# Discharge cost below best point in SHOP

- The conclusion: Need to add binary variables to the discharge cost curve as well. This has not been done in SHOP before, and some more work is required before it can be finished.

# Case reporting format






SINTEF

# Example of code generating log file

```
bool SetIntValue(char* objectType, char* objectName, char* attributeName, int value)
 {
   lock(ThisLock)
   {
     std::string isoObjectName = UTF8toISO8859_1(objectName);
     if (pyLog)
       fprintf(pyFile, "shop.SetIntValue(\"%s\", \"%s\", \"%s\", %d)\n", objectType,
isoObjectName.c_str(), attributeName, value);
```

# Original file

```
import pandas as pd
import matplotlib.pyplot as plt

from pyshop import ShopSession

#Create a new SHOP session.
shop = ShopSession('', '', False, 'case.log')

#Set time resolution
starttime=pd.Timestamp('2018-02-27')
endtime=pd.Timestamp('2018-02-28')
shop.set_time_resolution(starttime=starttime, endtime=endtime, timeunit='hour')

#Add topology
rsv1=shop.model.reservoir.add_object('Reservoir1')
rsv1.max_vol.set(12)
rsv1.lrl.set(90)
rsv1.hrl.set(100)
rsv1.vol_head.set(pd.Series([90, 100, 101], index=[0, 12, 14], name=0))
rsv1.flow_descr.set(pd.Series([0, 1000], index=[100, 101], name=0))

plant1=shop.model.plant.add_object('Plant1')
plant1.outlet_line.set(40)
plant1.main_loss.set([0.0002])
plant1.penstock_loss.set([0.0001])

p1g1=shop.model.generator.add_object('Plant1_G1')
plant1.connect().generator.Plant1_G1.add()
```

# Resulting log file

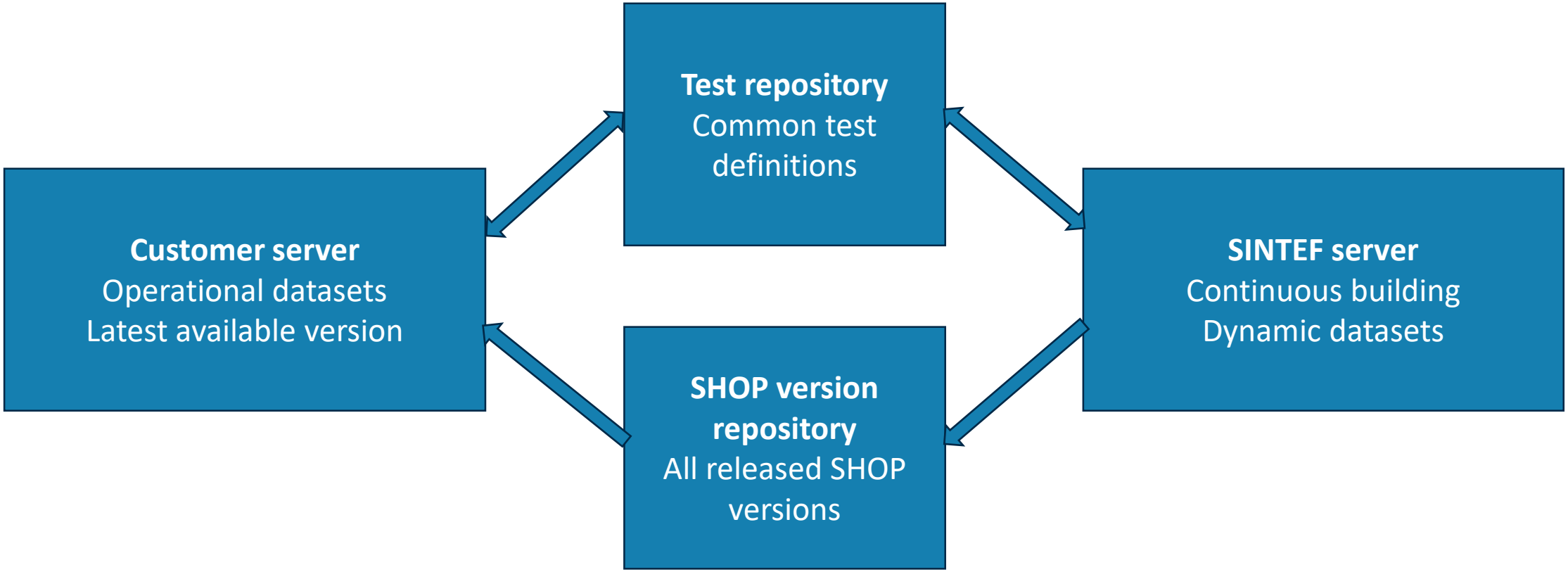
```
#  -*- coding: iso-8859-1 -*-
import pyshop
import numpy as np
import os

def init_shop_api(dll_path, license_path):
    import shop_pybind as pb
    return pb.ShopCore(False)


shop = init_shop_api("", "")

shop.GetObjectNamesInSystem()
shop.GetObjectTypesInSystem()
shop.GetCommandTypesInSystem()
shop.SetTimeResolution("20180227000000", "20180228000000", "hour")
shop.AddObject("reservoir", "Reservoir1")
shop.GetObjectNamesInSystem()
shop.GetObjectTypeAttributeNames("reservoir")
shop.GetObjectTypeAttributeDatatypes("reservoir")
shop.SetDoubleValue("reservoir", "Reservoir1", "max_vol", 12)
shop.SetDoubleValue("reservoir", "Reservoir1", "lrl", 90)
shop.SetDoubleValue("reservoir", "Reservoir1", "hrl", 100)
shop.SetXyCurve("reservoir", "Reservoir1", "vol_head", 0.000000, [0,12,14], [90,100,101])
shop.SetXyCurve("reservoir", "Reservoir1", "flow_descr", 0.000000, [100,101], [0,1000])
shop.AddObject("plant", "Plant1")
shop.GetObjectNamesInSystem()
shop.GetObjectTypeAttributeNames("plant")
shop.GetObjectTypeAttributeDatatypes("plant")
shop.SetDoubleValue("plant", "Plant1", "outlet_line", 40)
shop.SetDoubleArray("plant", "Plant1", "main_loss", [0.00020000000000000001])
shop.SetDoubleArray("plant", "Plant1", "penstock_loss", [0.0001])
shop.AddObject("generator", "Plant1_G1")
```

# ~~Test~~ User driven development






SINTEF

SINTEF

Teknologi for et bedre samfunn