HydroCen
NORWEGIAN RESEARCH CENTRE
FOR HYDROPOWER TECHNOLOGY

CENTRE FOR ENVIRONMENT-FRIENDLY ENERGY RESEARCH

HydroCen Report nr. 39

Ramping Constraints in Hydropower Scheduling

With application in the SDDP-based ProdRisk model

Arild Helseth, Stefan Rex

NTNU SINTEF NINA

HydroCen

The main objective of HydroCen (Norwegian Research Centre for Hydropower Technology) is to enable the Norwegian hydropower sector to meet complex challenges and exploit new opportunities through innovative technological solutions.

The research areas include:

- Hydropower structures
- Turbine and generators
- Market and services
- Environmental design

The Norwegian University of Science and Technology (NTNU) is the host institution and is the main research partner together with SINTEF Energy Research and the Norwegian Institute for Nature Research (NINA).

HydroCen has about 50 national and international partners from industry, R&D institutes and universities.

HydroCen is a Centre for Environment-friendly Energy Research (FME). The FME scheme is established by the Norwegian Research Council.

The objective of the Research Council of Norway FME-scheme is to establish time-limited research centres, which conduct concentrated, focused and long-term research of high international calibre in order to solve specific challenges in the field.

The FME-centres can be established for a maximum period of eight years HydroCen was established in 2016.

Ramping Constraints in Hydropower Scheduling

With application in the SDDP-based ProdRisk model

Arild Helseth, Stefan Rex

HydroCen

Helseth, A., Rex, S. 2023. Ramping Constraints in Hydropower Scheduling. HydroCen report 39.
Norwegian Research Centre for Hydropower Technology

Trondheim, 2023

ISSN: 978-82-93602-40-8 (digital publikasjon)
ISBN: 2535-5392

© SINTEF Energi 2023
Publikasjonen kan siteres fritt med kildeangivelse

KVALITETSSIKRET AV
Birger Mo

FORSIDEBILDE
NN

NØKKELORD
Driftsplanlegging i vannkraftsystemer, miljørestriksjoner, rampebegrensninger

KONTAKTOPPLYSNINGER

HydroCen
Vannkraftlaboratoriet, NTNU
Alfred Getz vei 4
Gløshaugen,
Trondheim

www.HydroCen.no

Abstract

Ramping constraints on water flows in hydropower systems may, e.g., ensure the preservation of ecological conditions, safely operate river systems, and prevent unfavorable ice formation. In this work we study how ramping constraints on a specific type of water flow (discharge) can be included in the medium-term hydropower scheduling. There are numerous examples of ramping constraints on discharge in the Norwegian hydropower system. We focus on the stochastic dual dynamic programming (SDDP) method as applied in the ProdRisk computer model. The context is thus a price-taking hydropower producer operating a hydropower system in a liberalized market.

The accuracy of ramping constraints in optimization models depends on the time discretization in the models. While the optimization models treat changes in discharge as instantaneous changes between two consecutive time steps, the changes will in practice need to follow the rate of change provided by the ramping constraint. Finer time resolution makes it possible to control the accuracy to a desired level but comes at the cost of increased computation time.

As an alternative to refining time resolution, we developed a new technique for expressing the *transition cost* involved when changing the magnitude of discharge between time steps in the optimization model. This technique is based on the assumptions that an increase (resp. decrease) in discharge between consecutive time steps is motivated by an increase (resp. decrease) in power price, and that the change in discharge needs to follow the ramping constraint symmetrically around the shift in time step. The cost (or lost revenue) of respecting the constraint is expressed as a function of decision variables and added to the objective function of the model.

A research prototype of the ProdRisk model was developed, including the possibility to define ramping constraints on discharge and to approximate the transition costs. The prototype was tested on two test systems, and results are presented and discussed in this report.

Sammendrag

Rampebegrensninger på vannføringer i vannkraftsystemer kan for eksempel sikre bevaring av økologiske forhold, trygge drift av elvesystemer og forhindre ugunstig isdannelse. I dette arbeidet studerer vi hvordan rampebegrensninger på driftsvannføring kan inkluderes i vannkraftplanleggingen på mellomlang sikt. Det er mange eksempler på rampebegrensninger for utslipp i det norske vannkraftsystemet. Vi fokuserer på metoden stokastiske dual dynamisk programmering (SDDP) slik den brukes i ProdRisk. Konteksten er altså en vannkraftprodusent som driver et vannkraftsystem i et liberalisert marked.

Nøyaktigheten av rampebegrensninger i optimaliseringsmodeller avhenger av tidsdiskretiseringen i modellene. Mens optimaliseringsmodellene behandler endringer i utslipp som øyeblikkelige endringer mellom to påfølgende tidstrinn, vil endringene i praksis måtte følge endringshastigheten gitt av rampebegrensningen. Finere tidsoppløsning gjør det mulig å kontrollere nøyaktigheten til et ønsket nivå, men kommer på bekostning av økt beregningstid.

Som et alternativ til å avgrense tidsoppløsningen utviklet vi en ny teknikk for å uttrykke overgangskostnadene som er involvert når man endrer driftsvannføring mellom tidstrinn i optimaliseringsmodellen. Denne teknikken er basert på antakelsene om at en økning (resp. reduksjon) i driftsvannføring mellom påfølgende tidstrinn er motivert av en økning (resp. reduksjon) i kraftpris, og at endringen i vannføring må følge rampebegrensningen symmetrisk rundt skifte i tidstrinn. Kostnaden (eller tapt inntekt) ved å respektere begrensningen uttrykkes som en funksjon av beslutningsvariabler og legges til i modellens objektivfunksjon.

En forskningsprototype av ProdRisk ble utviklet, inkludert muligheten til å definere rampebegrensninger for driftsvannføring og å beregne tilnærmede overgangskostnader. Prototypen ble testet på to testsystemer, og resultater presenteres og diskuteres i denne rapporten.

Content

1 Introduction

1.1 Background

The term ramping refers to the rate of change of a measured quantity in a physical system. Ramping may be subject to constraints for reasons that are technical, environmental, or even jurisdictional.

In the context of hydropower systems, quantities such as power generation, water flows and reservoir levels may be subject to ramping constraints. While hydropower systems are known for fast and flexible technical ramping capability, environmental constraints may discourage rapid variations in water flows and reservoir levels. Environmental constraints on ramping may, e.g., serve to avoid ecological strain, increase the security along the river, and stabilize ice formation.

A user survey conducted within the HydroCen research center pointed to the need for better representation of ramping constraints on river flows in hydropower scheduling tools [1]. This work should be seen as an effort to meet this need.

1.2 Organization

This work focuses on ramping constraints on the discharge waterway. The possibility to model such constraints was implemented in experimental Julia/JuMP code as well as a research prototype of the ProdRisk model [2], which is based on the stochastic dual dynamic programming (SDDP) algorithm. The prototype was validated in two test cases.

2 Briefly on the Stochastic Dual Dynamic Programming

This chapter provides a brief introduction to the SDDP algorithm. A special variant of the SDDP algorithm is used in ProdRisk, where the exogenous market prices are treated in an outer layer based on stochastic dynamic programming (SDP). For early references on the hybrid SDDP/SDP algorithm used in ProdRisk, the reader is referred to [3] and [4]. Recent research related to this algorithm involves topics such as multi-market scheduling [5], maintenance scheduling [6], and environmental constraints [7].

2.1 Description of the SDDP Algorithm

The objective of the hydropower scheduling problem is to maximize the expected profit over the period of analyses, which typically covers 2-5 years. It is important to include uncertainty both in future inflow to the reservoirs and electricity prices over the period of analyses. We let the model take decisions in weekly stages. That is, for each week the value of the uncertain variables for that week are known to the model, and the optimal decisions can be made for that week.

The structure of decision sequences can be represented in a scenario tree. For each decision node the tree further branches into a set of new decision nodes in each decision stage. This leads to an exponentially large problem: For example, if we plan for a period of 104 weeks and have 12 branches each week, the full scenario tree would have 1.72×10^{112} nodes, which is far beyond what can be stored in memory and be solved in reasonable computation times. For this reason, we rely on algorithms that:

a) Decompose the optimization problem into weekly decision problems that can be solved independently.
b) Apply sampling algorithms to avoid searching the entire tree.
c) Apply cut sharing to efficiently "collapse" the scenario tree.

A main iteration of the SDDP algorithm is illustrated in Figure 1. It consists of a forward and backward iteration, as briefly outlined below. Note that this illustration and the explanation below refer to the SDDP-part of the combined SDP/SDDP model used in ProdRisk. Stochastic price is treated by an outer SDP-loop, as described in detail in [5].

In the forward iteration of the SDDP algorithm, we sample a set of inflow scenarios {s1-s3} and simulate week-by-week along these scenarios by solving a linear programming (LP) problem for each week. All scenarios share the initial state, which is defined by a vector of initial reservoirs and inflows in the first week. The simulated state trajectory along the sampled scenarios is shown as the thick black line Figure 1. We keep track of the simulated reservoir levels and find the expected simulated profit over these scenarios, and that value serves as a lower bound.

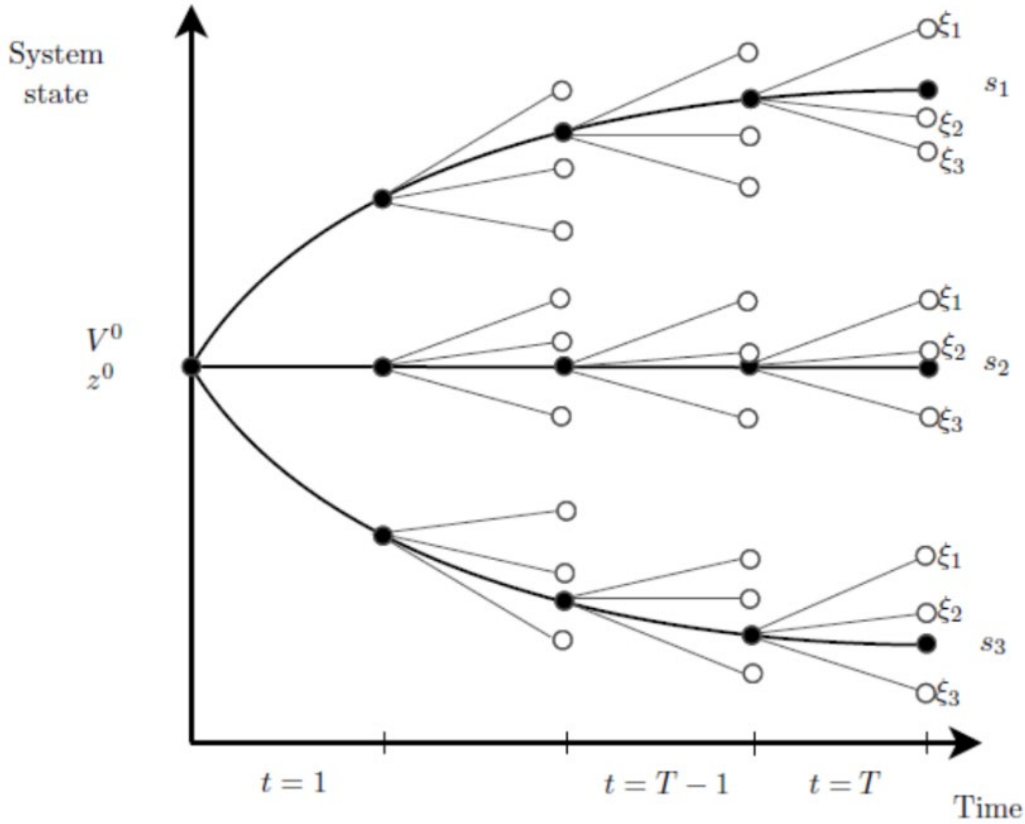

Figure 1 Illustration of a main iteration in the SDDP algorithm.

In the backward iteration, so-called Benders cuts (or just cuts) are built for each time stage. A cut represents a linear inequality constraining the future expected profit as a function of the state variables. Once created, cuts are added to a list of cuts representing that stage. Consider the state obtained in scenario 1 in stage *T* - 1 in Figure 1. The simulated state trajectory (reservoir level and inflow) up to this point is known from the forward iteration. We now sample 3 vectors of errors or "white noise", so that inflow samples for the coming week can be calculated. For each sample, we solve an LP problem. The average dual values on state variables are used to create a cut constraining the expected future profit seen from previous decision stage (*T* - 2). An important feature here is cut sharing; Cuts computed for a specific initial state at a given stage can be shared among all states for that stage. The expected future profit seen from the beginning of the planning horizon serves as an upper bound.

For maximization problems, uncertainty around the lower bound can be used as a convergence check. As discussed in the literature, this statistical convergence criterion has limitations, and for practical purposes the stabilization of the lower bound may serve as an alternative criterion [8].

2.2 The Weekly Decision Problem

As explained above, the overall optimization problem is decomposed into weekly decision problems formulated as LP problems. The basic set-up is as follows:

$$\max \lambda_t p_t + \alpha_{t+1}$$
S.t.:
- Water constraints (1)
- Production function
- Cuts constraining α_{t+1}

The objective in (1) is to maximize the 'here and now' profit, obtained as the product of market price λ_t and generation p_t, and the 'expected future profit' represented by α_{t+1}. The LP problem is constrained by technical constraints on the reservoir storages and waterways and a linear approximation of the transformation from water to power, often referred to as the production function. Finally, the expected future profit is constrained by cuts that are iteratively built and added to the LP problem in the backward iterations, as explained above.

Being a widely applied scheduling model, ProdRisk allows for a variety of technical and environmental constraints in the problem formulation. We will not go into details here but refer to the refer to [2] for more information.

The SDDP algorithm converges if the problem formulation is convex. This requirement translates into the following for a maximization problem:

The expected future profit (α_{t+1}) should be concave in all state variables.

In the SDDP problem formulated in ProdRisk, there are three types of state variables: reservoir volumes, average weekly inflow and average weekly power price. The expected future profit is concave in reservoir volume and inflow, but not in price. For this reason, price is discretized and treated in an outer SDP loop.

The cut takes the following form:

$$\alpha_{t+1} - \sum_{h=1}^{NH} \pi_h \left(v_h - v_h^* \right) - \sum_{s=1}^{NS} \mu_s \left(z_s - z_s^* \right) \le \alpha_{t+1}^* \quad (2)$$

Where *NH* is the number of hydropower modules, *NS* the number of inflow series, v_h the reservoir volume, z_s the normalized inflow, π_h and μ_s are the cut coefficients for reservoir volume and inflow, respectively. All starred variables are trial values.

3 Modelling of Ramping Constraints

3.1 Mathematical Formulation

Ramping constraints on discharge define a maximum allowed change in discharge over a defined time interval. The constraint may relate to both upward and downward ramping. Mathematically, the constraint in each time interval *k* (Δk hours) can be formulated as:

$$-\Delta Q^{-} \leq \frac{q_k - q_{k-1}}{\Delta k} - s^{+} + s^{-} \leq \Delta Q^{+} \tag{3}$$

Where parameters ΔQ^{-} and ΔQ^{+} limit the down- and up-ramping in m3/s/h. Variables q_k and q_{k-1} describe the flow rate in m3/s in time step *k* and *k-1*, and parameter Δk the length of each time step in hours. Slack variables s^{+} and s^{-} in m3/s/h are included to ensure feasibility and are penalized in the objective function.

For each weekly decision problem, constraints of type (3) are included for each hydropower module with ramping constraints and for each time step. The time-linking of discharge variables introduced in (3) makes the discharge variable a state variable. That is, the expected future profit will depend on the discharge in the last time step within the week.

The cut formulated in (2) can be extended as follows:

$$\alpha_{t+1} - \sum_{h=1}^{NH} \pi_h \left(v_h - v_h^{*} \right) - \sum_{s=1}^{NS} \mu_s \left(z_s - z_s^{*} \right) - \sum_{r=1}^{NR} \gamma_e \left(q_r - q_r^{*} \right) \leq \alpha_{t+1}^{*} \tag{4}$$

Where *NR* is the number of modules with ramping constraints, γ_e the cut coefficient for the discharge variable, and q_r is the discharge in the last time step of the week.

3.2 Challenges with Time-Discretization

While the formulation in Section 3.1 has the capability to limit ramping at a desired precision level, there are practical and computational challenges associated with it. Some concessions define constraints on ramping that are on a finer time resolution than the model naturally can handle. For example, some concessions describe ramping limitations measured on 15-minute intervals, whereas the finest time discretization in ProdRisk is currently one hour. Moreover, input data such as inflows (weekly or daily) or market prices (minimum hourly) do not motivate a finer time resolution in the model. Exact modelling of ramping limits measured in finer time intervals is therefore challenging, at best.

Another aspect is the computational burden when applying fine time resolution and adding ramping constraints of type (3). The weekly decision problems grow leading to prohibitive computation times. The trade-off between improved results and added computational complexity will be further elaborated on in the case studies.

Below we list three possible measures for dealing with the above-mentioned challenges with ramping constraints, based on continuous-time formulation, constraint relaxation, and transition costs.

A different take on ramping than equation (3) is presented in [9], where the scheduling is made ***continuous time***. Concepts from the continuous time methodology may be worthwhile further exploring in this context but were not further considered within the scope of this work.

The constraint of type (3) is formulated as an inequality and is therefore not necessarily binding. ***Relaxation*** of equation (3) provides a possibility for computational speed-up. By relaxation we refer to the process of first solving the LP problem without the constraints (they are 'relaxed'). Subsequently, from the solution of the relaxed LP problem, we find the equations being violated, and add these before the LP problem is re-solved. This procedure is repeated until no more constraint violations are detected. Relaxation is already used when treating cuts in the weekly decision problem. Relaxation can be combined with sophisticated methods for predicting which constraints to include. For the problem at hand, maximizing the use of water against an exogenous power price, challenges with ramping will typically occur between intervals with rapid price changes. Thus, it may be possible to predict whether the constraints will be binding or not based on the exogenous price profile. For this purpose, one could apply statistical methods or machine learning.

With discrete time, the transition from a discharge rate in one time step to the next is assumed to be instantaneous, as a step-function. In reality, this transition requires some time due to the laws of physics. Regardless of the existence of ramping constraints, we can therefore argue that the time discretization contributes to overestimating the capability of adjusting generation to price. Since the price is treated exogenously, it is possible to estimate a ***transition cost*** that covers the adaptation of discharge from one time step to the next. This concept is described in detail next.

3.3 Transition Cost

Below follows a description of the transition cost functionality. A similar description is provided in [10].

3.3.1 Methodology

To supplement ramping constraints in limiting the ramping according to the provided requirements, we introduce the concept of transition cost, as illustrated in Figure 2. The figure shows two time-steps with low (π_L) and high (π_H) prices in [€/MWh] and corresponding low and high discharge rates (q_L and q_H) in [m3/s]. The real discharge will not change instantaneously from the one time-step to the next, as in a discrete model (solid line). Rather, it will follow a maximum ramp rate which is limited by rules and recommendations motivated by physical and/or legislative factors. This ramp rate is illustrated in Figure 2 as the orange dotted line.

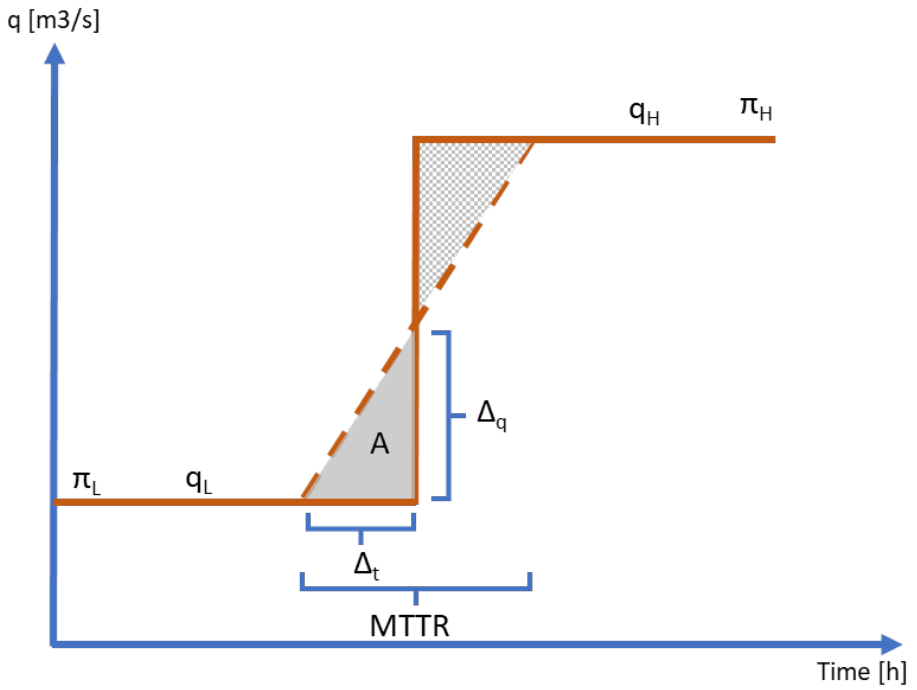

Figure 2 Illustrating transition cost of ramping.

We assume that the change of discharge rate will start Δt_1 hours before time step 2 and will complete Δt_2 hours after step 2 has started, and that $\Delta t = \Delta t_1 = \Delta t_2$. The mean time to ramp (MTTR) is 2Δt. An additional volume A will be discharged at price 1, while the discharge at price 2 is reduced by volume A', where A = A'. Since the prices are known a priori, the economic loss (or the transition cost) of changing discharge rate between the two time-steps can be estimated.

First, we express the Δt [hours] based on the discharge:

$$\Delta t = \frac{MMTR}{2} = \frac{q_H - q_L}{2\Delta Q} \tag{5}$$

Where ΔQ is the maximum allowed ramp rate in [m3/s/h]. Next, Δq can be expressed:

$$\Delta q = \frac{q_H - q_L}{2} \tag{6}$$

The re-allocated volume labelled A in [m3] in Figure 2 can then be expressed as:

$$A = \frac{3600\Delta q \Delta t}{2} = \frac{3600\frac{(q_H - q_L)}{2}\frac{(q_H - q_L)}{2\Delta Q}}{2} = \frac{450(q_H - q_L)^2}{\Delta Q} \tag{7}$$

The factor 3600 converts m3/s to m3/h.

The cost (or lost revenue) associated with the re-allocated volume can be found (in [€]) by multiplying the volume with the price difference and the water-to-energy efficiency η in [MWh/m3]. We assume that η corresponds to the best efficiency point so that the cost (y) is an upper bound on the "true" cost.

$$y = A\eta \cdot (\pi_H - \pi_L) = C \cdot (q_H - q_L)^2 \tag{8}$$

Where C is a constant:

$$C = \frac{450\eta(\pi_H - \pi_L)}{\Delta Q} \tag{9}$$

The cost y is a quadratic function of the difference between variables q_H and q_L can be added to the objective function. Moreover, the cost is inversely proportional to the maximum ramping rate ΔQ and proportional to the price difference $\pi_H - \pi_L$.

So far, we have assumed that $\pi_H \geq \pi_L$. The expressions presented above are also valid in the opposite case if $\pi_H - \pi_L$ is replaced by $|\pi_H - \pi_L|$, as long as high/low price and discharge levels remain perfectly correlated in the optimal solution.

Note that we do not account for the misplaced volume in the reservoir balances and in other constraints related to water. The estimated TC is purely an additional cost element reflecting the lost revenue due to ramping limitations in between time steps. Since the TC is expressed as a function of decision variables, both primal and dual variables are directly impacted by this cost. In particular, since the cost is a quadratic function of change in discharge, it is clear that the TC functionality will limit discharge ramping.

3.3.2 Example

We illustrate the computation of transition cost in a fictitious example of a single plant operated over two time-steps, as described below. It is assumed that the optimization model has already computed the optimal discharges. The time step length is 1 hour and the prices, discharge decisions, efficiency and maximum allowed ramp rate is provided in Table 1.

Table 1 Example parameters.

π_L (øre/kWh)	π_H (øre/kWh)	q_L (m3/s)	q_L (m3/s)	η (kWh/m3)	ΔQ (m3/s/h)
40	60	10	20	1.0	10

According to (7) and with reference to Figure 2, the additional volume A that needs to be moved from the high-price to the low-price time step is 4500 m3. The plant delivers 36 MW in the low-price period and 72 MW in the high-price period, which involves a revenue stream of 14.40+43.20 = 57.60 kNOK for the two time-steps. The transition cost estimated by (8) amounts to 0.90 kNOK in this example, reducing the profit to 57.60-0.90 = 56.70 kNOK.

The relationship between transition cost and maximum allowed ramp rate is plotted in Figure 3. Note that the ramp from q_L = 10 m3/s to q_L = 20 m3/s in this example feasible only if maximum allowed ramping is at least 10 m3/s/h.

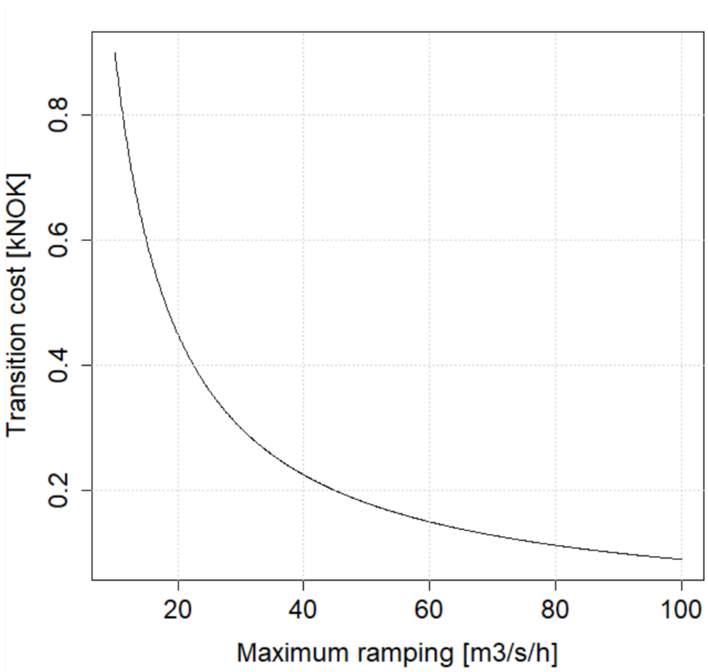

Figure 3 Transition cost as a function of maximum ramping rate.

3.3.3 Linearization

Adding the cost element in (8) to the objective leads to a quadratic optimization problem. Since it is crucial for the scheduling model to maintain the weekly decision problems as LP problems, the cost component was linearized as follows.

First the range of *N* possible discrete Δq values are defined, as illustrated by the orange dots along the horizontal axis in Figure 4. In this work we simply assume that the N points are evenly distributed over the range of feasible ramping values. For each discrete point Δq^* we compute both the exact transition cost y*, according to (8), and the derivative $\left.\frac{\partial y}{\partial \Delta q}\right|_{\Delta q^*}$. By forming a first-order Taylor expansion around this operating point, a linear constraint can be formulated:

$$y \geq y^* + \left.\frac{\partial y}{\partial \Delta q}\right|_{\Delta q^*} \left(\Delta q - \Delta q^*\right) \tag{10}$$

We refer to (10) as transition cost cuts (TC-cuts) in the following. A total of N TC-cuts can be computed for each price period prior to solving the LP problems.

Note that all TC-cuts were included when solving the weekly decision problems. Most likely, sophisticated cut management could improve the computational efficiency, but was not considered in this work.

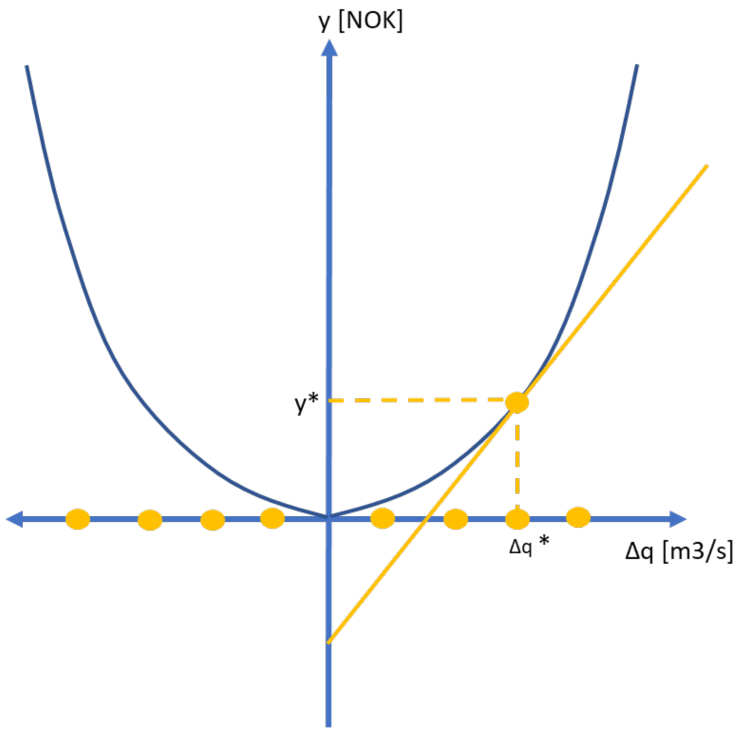

Figure 4 Linearizing the transition cost component.

3.4 Validating Transition Cost Functionality

Before implementing the TC functionality as TC-cuts in ProdRisk, we validated the functionality in a simple experiment. A computer code in Julia/JuMP was prepared, as documented in Appendix A, and the CPLEX optimization solver was applied. A single hydropower module was considered, to be optimized against a deterministic power price and inflow. A planning horizon of one week with 3-hour time steps was considered. The power station allows a maximum discharge of 30 m3/s with the best efficiency point at discharge 20 m3/s. The challenge is thus to use the limited initially stored water optimally to maximize profit over the planning horizon.

A maximum ramping of 10 m3/s/h was allowed. This amounts to 30 m3/s for the 3-hour time-steps, and thus ramping constraints in its original form will not be binding. We ran the code with 3 different formulations:

a) **Quadratic** – With TC functionality formulated as a quadratic problem, as in (8).
b) **Linearized** – With TC functionality represented by TC-cuts, as in (10).
c) **Original** - Without transition cost.

The economic results (objective) are shown in Table 2. We observe that with increasing number of TC-cuts, the linearized formulations gradually approach the objective of the quadratic. This demonstrates that the linearization approach can approximate the quadratic transition costs at an arbitrarily small tolerance. As expected, we see that the objective from the original formulation is higher than from the two others. This indicates the overestimated ramping capability in the original formulation. Moreover, it is interesting to see that, even with a low number of TC-cuts, the linearized version is significantly closer in objective function value to the quadratic than the original.

Table 2 Objective [NOK].

NCut	4	7	13	30	1000
Quadratic	271254.96	271254.96	271254.96	271254.96	271254.96
Linearized	273362.59	271755.24	271404.31	271274.30	271254.97
Original	294230.25	294230.25	294230.25	294230.25	294230.25

If the time resolution is finer than 3 hours, the ramping constraint of 10 m3/s/h will gradually bind the solution, leading to a lower objective function value (profit). With infinitesimal time steps, the ramping capability would be perfectly approximated. To compute the optimal solution with perfect consideration of the ramping constraint, we allowed the model to make discharge decisions according to a much finer time resolution while reservoir balances are on 3-hour time resolution. Figure 5 shows the reduction in profit as the time resolution gets finer. With finer time resolution we observe that the marginal reduction in optimal objective function value gradually approaches zero. With 3-hour time resolution (the green point in Figure 5), the profit is similar with and without ramping constraints. This is because the time resolution is too coarse to allow the ramping constraint to be binding.

The two red points in Figure 5 indicate the extremal solutions from the linearized formulation, as documented in Table 2. We find that the linearized formulation leads to a profit that is lower than the case with ramping constraints at a fine time discretization. We identified two reasons for this underestimation provided by the linearized formulation:

1) Applying ramping constraints at a fine time resolution provides a more flexible approach to ramping than the proposed TC functionality. While the latter assumes that ramping is symmetrically allocated around the shift from one time step to the next, ramping constraints allow the up/down ramping to be asymmetric around, and the model will choose such trajectories if economically more efficient. This is shown in Figure 6.
2) The cost computed in (9) is approximated since we need to choose a value for η. We chose to use the best efficiency point which will provide an upper cost bound and thus tend to overestimate the cost associated with volume A in Figure 4.

In more complex model formulations, it is likely that other technicalities contribute to deviations between the cost estimated by TC-cuts and what is the actual cost of the ramping constraints if evaluated at a fine time scale. Thus, we cannot without further investigations claim that the TC functionality will provide an overestimation of costs when implemented in complex scheduling models.

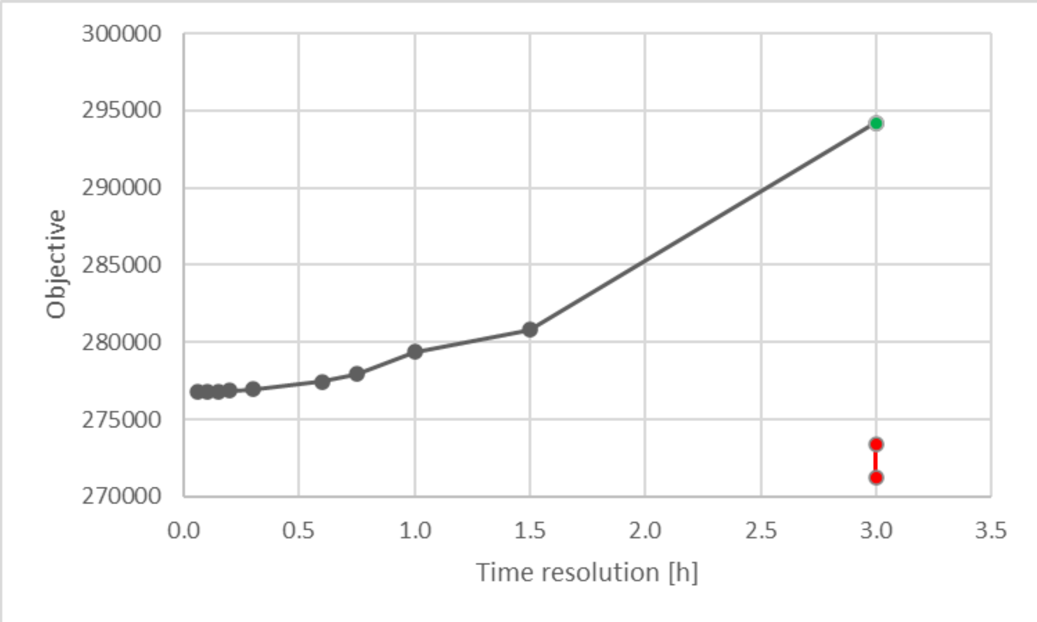

Figure 5 Objective function value as function of ramping constraint time resolution.

Discharge solutions along the simulated scenario are shown in Figure 6, for the fine time-discretization (blue), the original without TC functionality (grey) and with Quadratic TC (red). Simulations with TC-cuts will be identical to the Quadratic results if the number of TC-cuts are high enough. We can observe from the figure that the discharges from the Quadratic formulation are closer to the ones with fine time resolution.

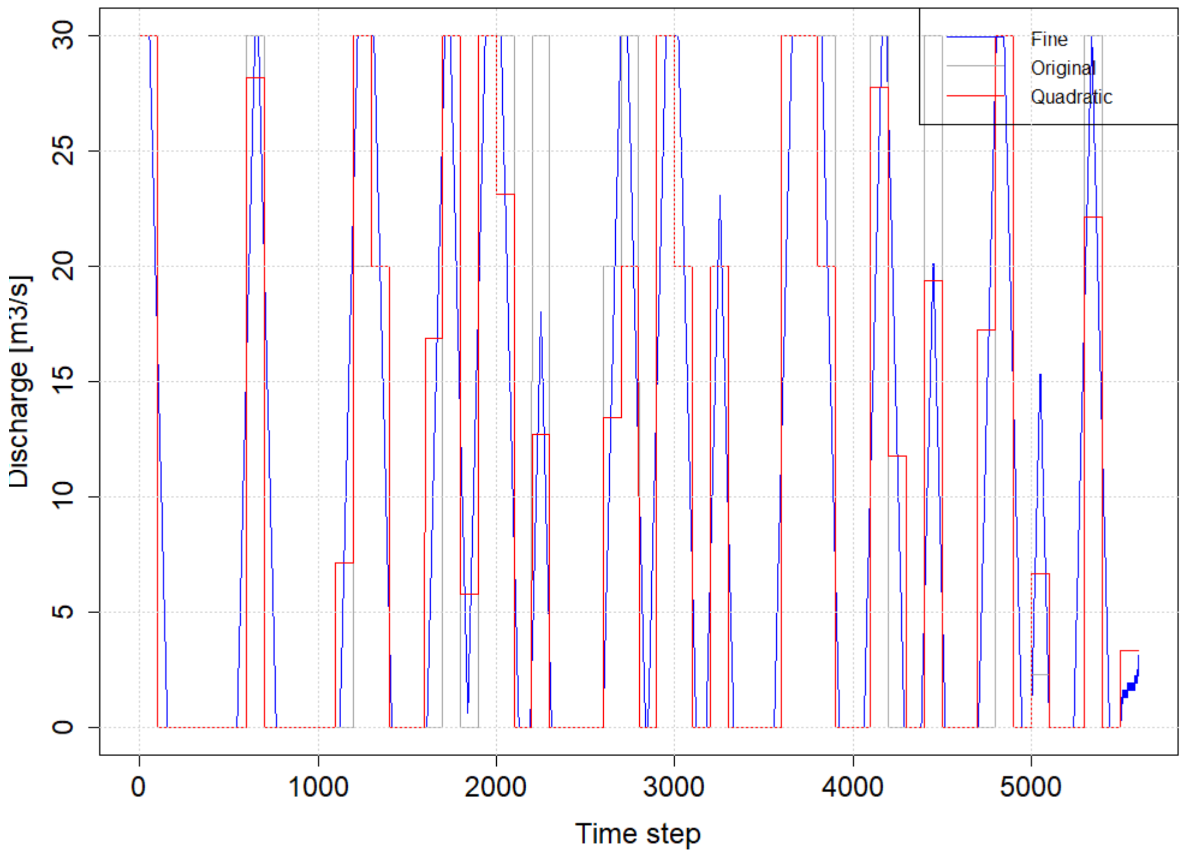

Figure 6 Discharge solutions.

4 Computational Experiments

4.1 Prototype Implementation and Experiment Setup

A prototype model of ProdRisk was prepared with the implementation of ramping constraints and TC-cuts. The ramping constraints can be included for a selected set of hydropower stations. The inclusion of the discharge variable as a state in the cuts was implemented as an optional feature. Ramping can be specified in various units and for specific intervals (weeks).

In case the TC functionality is used, the user must specify the number of TC-cuts to be used. By design, it is not possible to run the prototype with TC-cuts but without ramping constraints.

All tests were carried out on an Intel Core i7-9850H processor with maximum frequency of 4.60 GHz and 64 GB RAM. CPLEX version 12.2 was used to solve the LP problems, using the dual Simplex algorithm. All tests are run using 4 parallel processes.

The discharge variable is defined as a state variable for hydropower modules/stations with ramping constraints. We do not consider head dependency in our tests.

4.2 Test Case 1

The first system is a fictitious system with one hydropower module (reservoir plus plant), as shown in Figure 7. The system was optimized for a period of 104 weeks, with 46 time-steps within the week, of length varying between 1-5 hours. The maximum discharge is 65 m3/s, allowing a ramp rate of 65/5=13 m3/s/h to cover the whole range from 0 to 65 m3/s for the longest time steps.

We will not try to analyze results such as economic performance and reservoir utilization here, but rather focus on verification of the functionality.

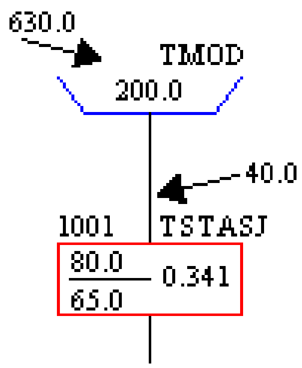

Figure 7 Technical description for Test system 1.

First, we simulated three cases with ramping constraints on discharge:

1) **NoRamp** – With no ramping constraints
2) **Ramp5** – With maximum allowed ramping up and down of 5 m3/s/h
3) **Ramp2** – With maximum allowed ramping up of 2 m3/s/h

The optimal objectives obtained by ProdRisk at convergence are reported in Table 3. As expected, the objective decreases with stricter ramping constraints.

Table 3 ProdRisk objective (measured as the so-called K-Kost) at convergence.

Case	K-Kost [MEUR]
NoRamp	14.54
Ramp5	14.53
Ramp2	14.52

Figure 8 shows the duration curves of the simulated ramping (lag-1 difference in discharge) for the three cases. The figure shows that the imposed ramping constraints are met for all cases.

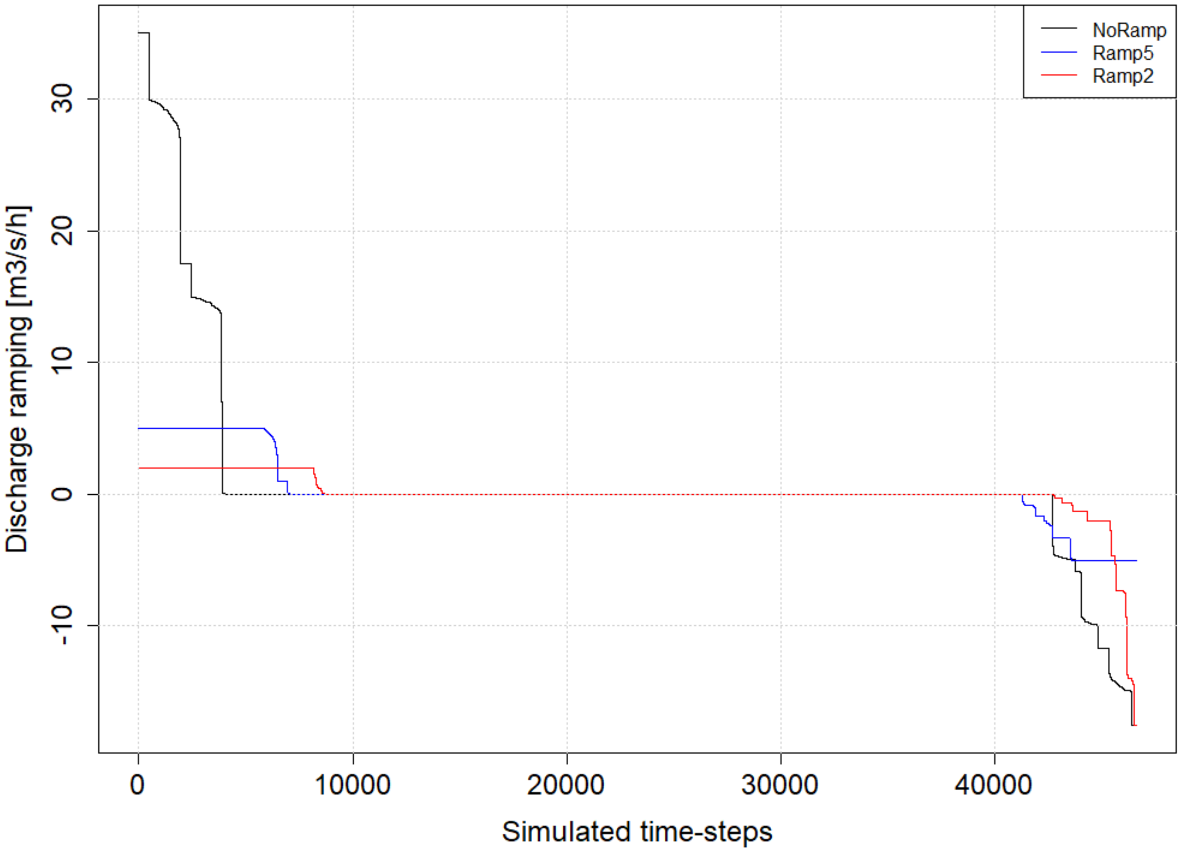

Figure 8 Ramping, sorted by magnitude, for the three cases.

Next, we studied the importance of including TC-Cuts. We selected the Ramp5 case and tested with a varying number of TC-cuts (NCuts), as reported in Table 4. Compared to the K-Kost[1] reductions in Table 3, the reductions in Table 4 are higher. Recall that an artificial cost component is added, as described in Section 3.3, to embed the cost of changing the discharge rate between time steps. We do not have the possibility to further assess the magnitude of these cost components, since ProdRisk does not allow a finer time resolution than 1 hour.

[1] K-Kost is a measure of the upper bound in ProdRisk.

Table 4 Performance with TC functionality.

Case	NCuts	Comp. time	K-Kost
1	0	99	14.53
2	6	168	14.47
3	20	260	14.45
4	100	909	14.44

Figure 9 shows how the TC-cuts serve to reduce the ramping.

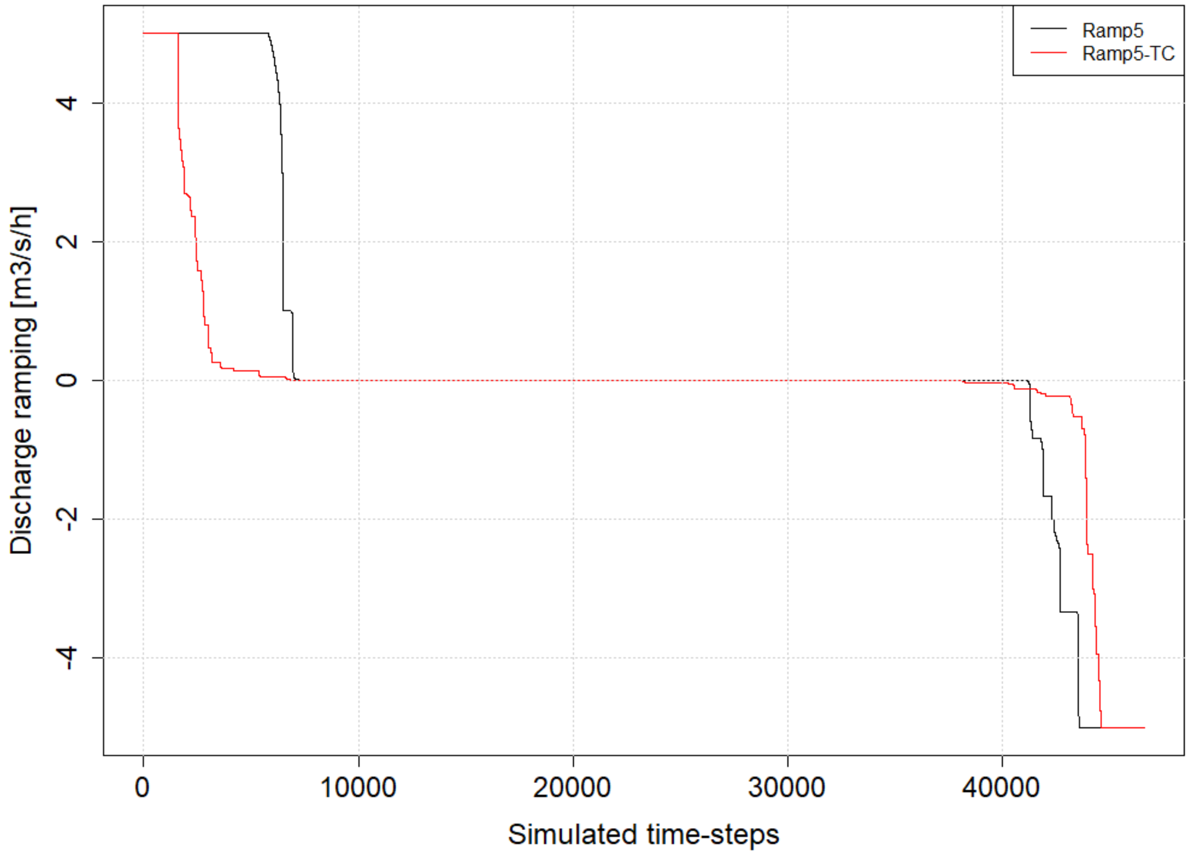

Figure 9 Ramping, sorted by magnitude, for case Ramp5 with (black) and without (red) TC-cuts.

Figure 10 shows a selected sequence of simulated discharge for cases 1 (NCuts=0) and 3 (NCuts = 20). The rapid fluctuations between zero, best efficiency point (30 m3/s) and maximum (65 m3/s) discharge observed in case 1 are significantly reduced in case 3.

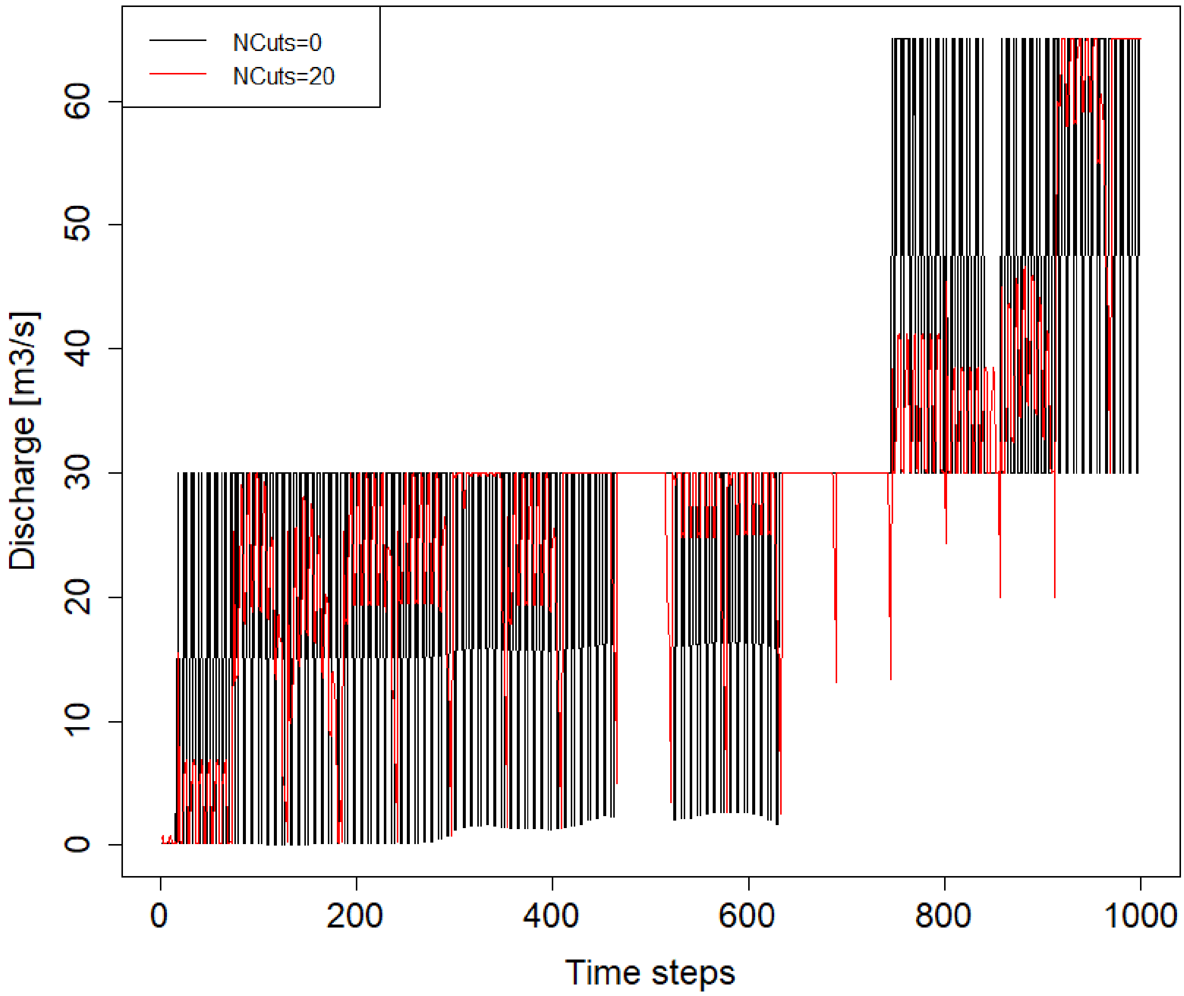

Figure 10 Discharge for a selected time sequence for case 1 (black) and case 3 (red).

4.3 Test Case 2

The hydropower system in Røssåga is located in Northern Norway and comprises two large power plants, 180+353 MW in our case study, with significant upstream reservoir capacity. An illustration of the system topology is provided in Figure 11. A minimum discharge constraint of 30 m3/s applies downstream to the power plants. A slightly different variant of this case study was presented in [10].

The system is optimized for a horizon of 156 weeks, using 56 3-hour time steps within the week.

We consider a ramping requirement associated with the outlet of the Nedre Røssåga power plant. As originally stated in the operating license, the constraint limits ramping to be at most 7.5 m3/s per 15 min when operating in the interval between the minimum discharge of 30 m3/s to 60 m3/s. In other words, the ramping from minimum discharge to 60 m3/s should not be done faster than one hour. This constraint is difficult to capture for two reasons. First, we do not allow for state-dependent ramping constraints in this work. That is, we currently have no functionality that allows conditioning the ramping requirement on the prevailing discharge rate. Secondly, the dataset is set up with a 3-hour time resolution. With the requirement above, the model can ramp 90 m3/s in a 3-hour time step, and thus the original purpose of the constraint is not captured by the model. With some data adjustments, we could facilitate cases with hourly time resolution, but at a high computational cost. If we assume a linear increase in computation time with problem dimensionality, which is typically an underestimation, the computation time is likely to increase with a factor of 3 when going from 3 to 1-hour time steps. We decided to conduct all experiments with the assumption that the ramping constraint applies for all discharge ranges and with the defined time resolution (3 hours per time step).

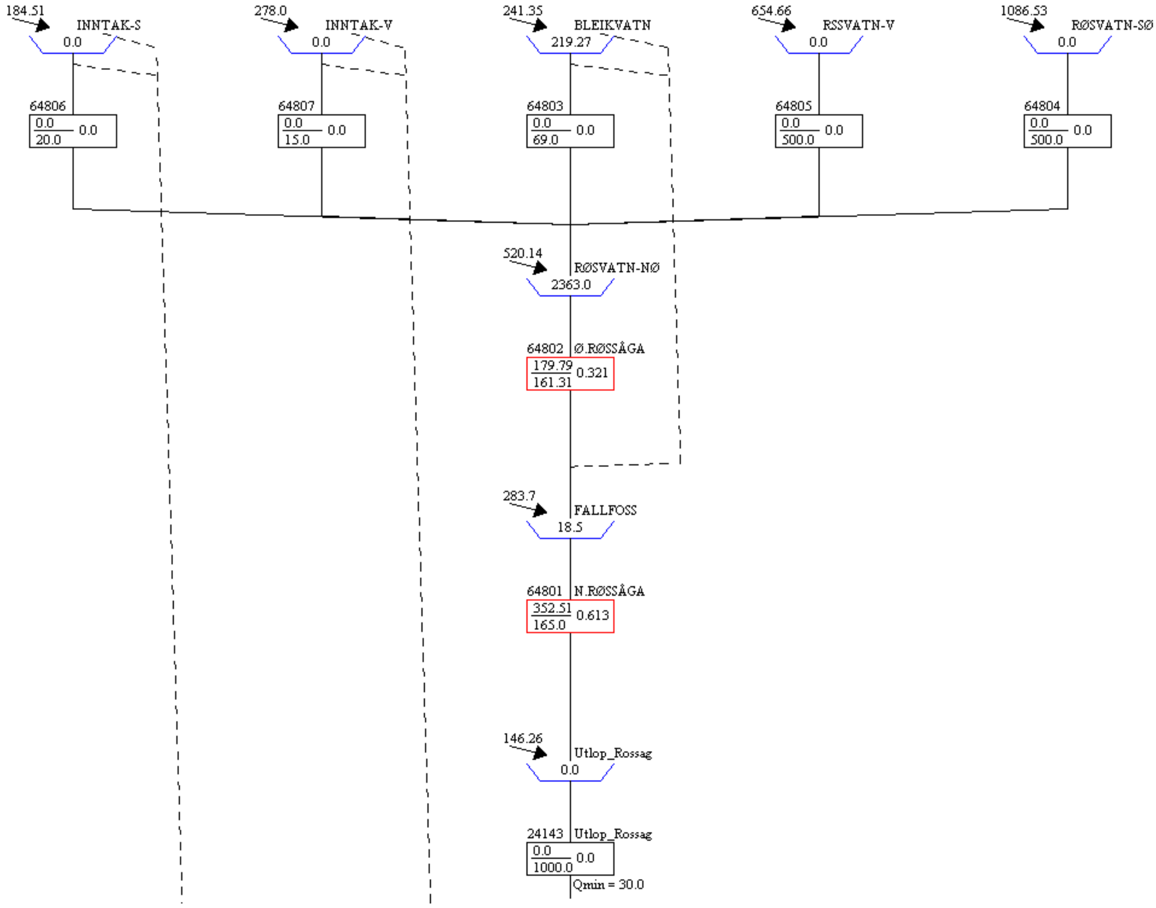

Figure 11 Technical description and topology of Røssåga hydropower system.

The following cases were run:

- **Rmp30** – With ramping constraint 30 m3/h/s
- **Rmp30TC-6** – With ramping constraint 30 m3/h/s and TC-Cuts, NCuts = 6
- **Rmp30TC-10** – With ramping constraint 30 m3/h/s and TC-Cuts, NCuts = 10
- **Rmp30TC-20** – With ramping constraint 30 m3/h/s and TC-Cuts, NCuts = 20
- **Rmp30TC-60** – With ramping constraint 30 m3/h/s and TC-Cuts, NCuts = 60
- **Rmp30TC-100** – With ramping constraint 30 m3/h/s and TC-Cuts, NCuts = 100

Computation times and objective function value (upper bound, K-Kost) are shown in Table 5. All cases were run using 4 parallel processes. The lower K-Kost indicates that the Rmp30TC- cases are more constrained than the Rmp30 case.

Table 5 Economic and computational performance.

Case	K-Kost [MEUR]	Comp. time [hr:min]
NoRmp	7221.27	2:14
Rmp30	7220.20	3:11
Rmp30TC-6	7186.73	3:39
Rmp30TC-10	7178.49	3:45
Rmp30TC-20	7169.51	4:14
Rmp30TC-60	7160.09	6:13
Rmp30TC-100	7158.79	7:34

Figure 12 shows the duration curves for simulated discharge ramping (lag-1 difference in discharge). The curves show that discharge ramping is more of a continuous process when the number of TC cuts increases. A significant reduction in ramping is observed already with 6 TC cuts (blue curve), but with a pronounced discretization error allowing a ramping of approximately 2 m3/s/h at no cost. This error could be treated by more sophisticated discretization procedure in the cut creation phase described in Section 3.3.

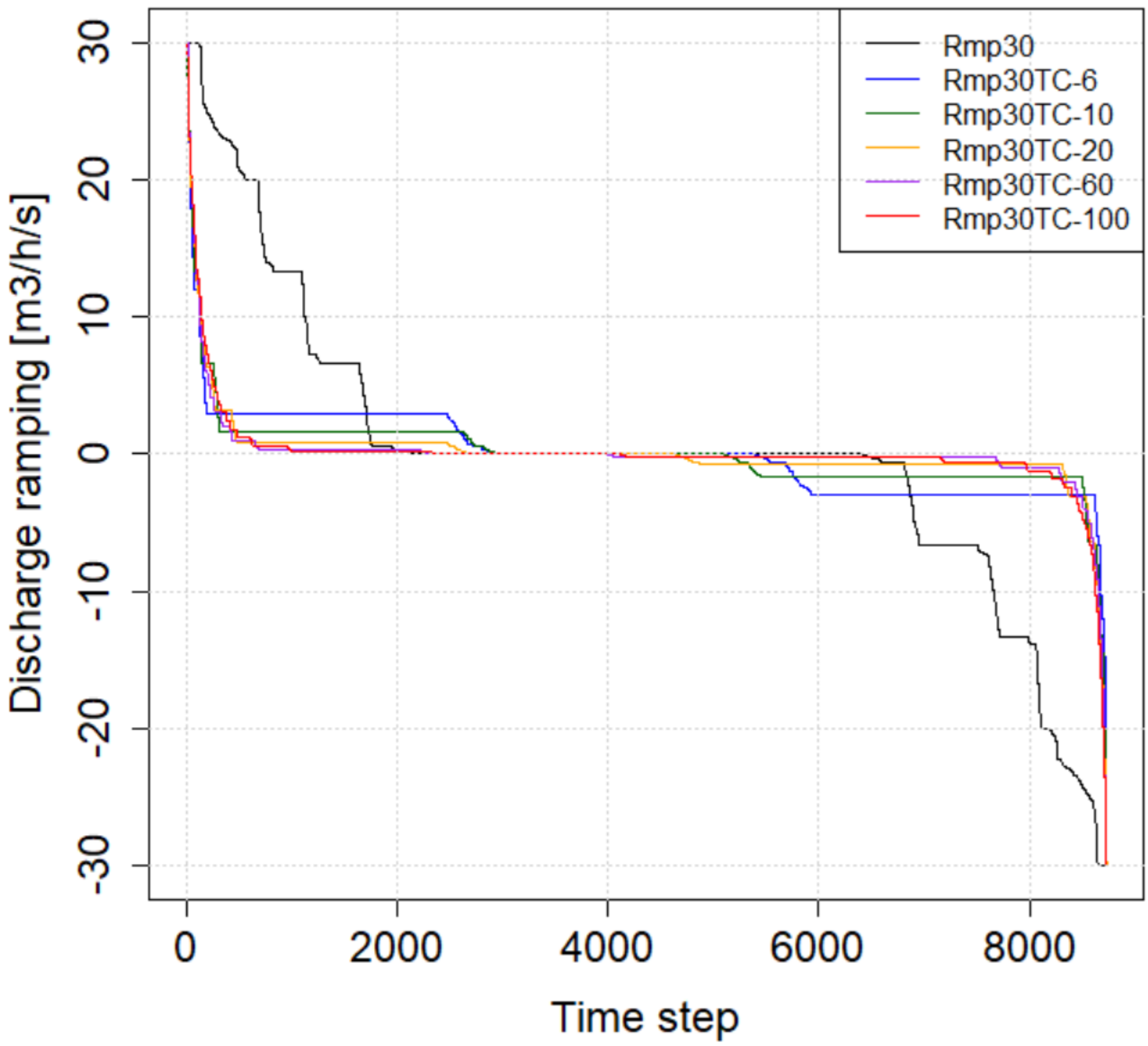

Figure 12 Duration curves for discharge ramping.

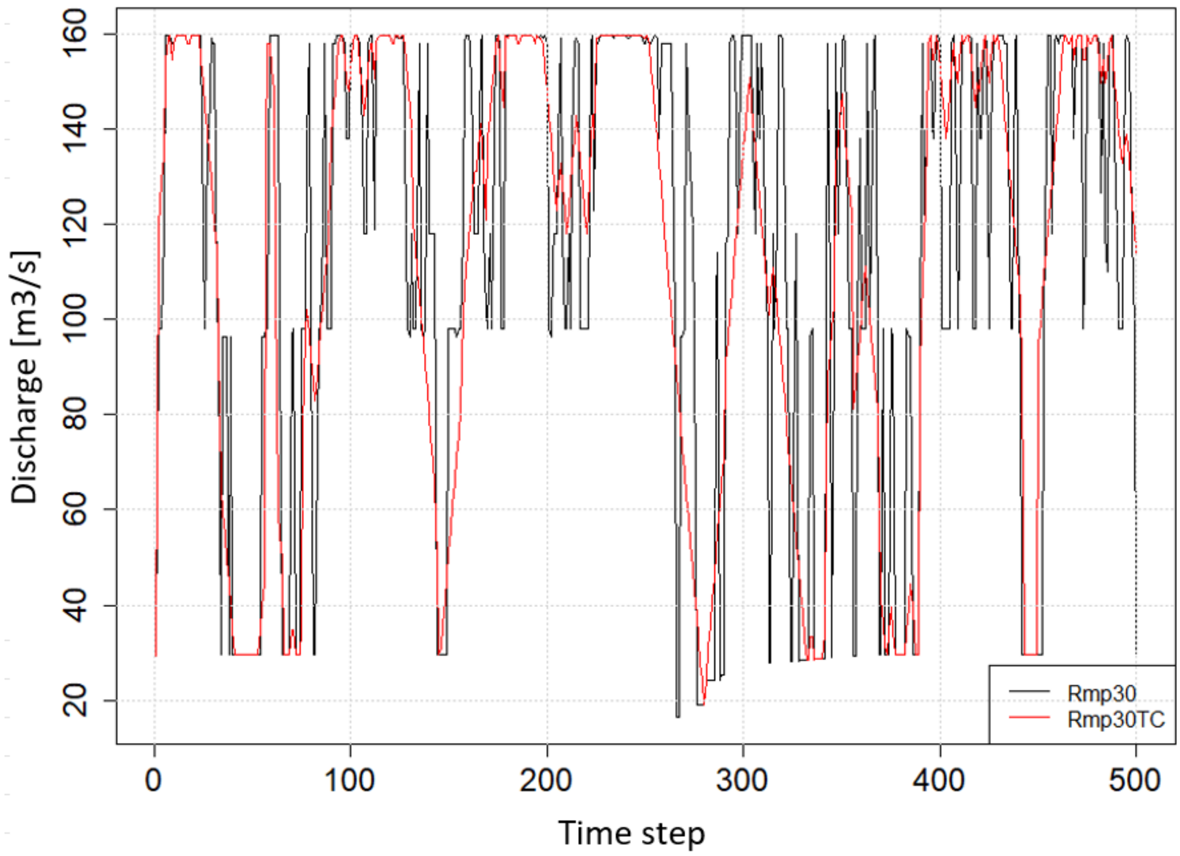

Figure 13 Discharge for a selected time sequence for cases Rmp30 (black) and Rmp30TC-100 (red).

Reservoir operation for the large reservoir Røssvatnet for cases Rmp30 and Rmp30TC-100 are presented in Figure 14. The NoRmp case is similar to case Rmp30. Rmp30-TC100 provides a slightly higher reservoir level on average than Rmp30.

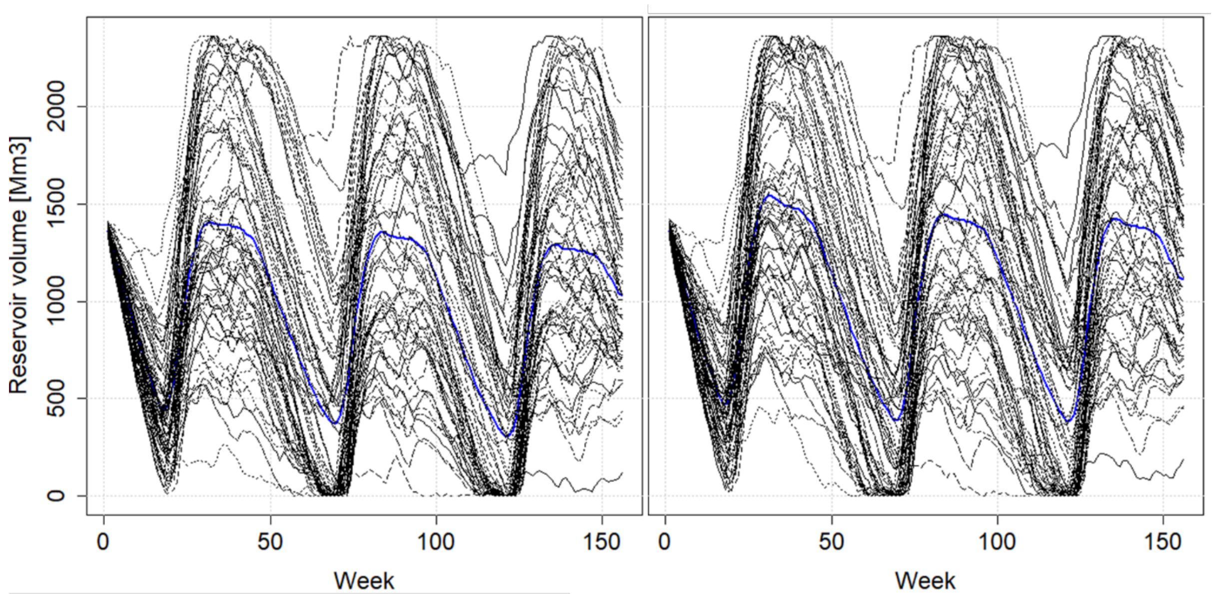

Figure 14 Reservoir operation Røssvatnet for Rmp30 (left) and Rmp30TC-100 (right) for all simulated scenarios, where the blue line shows the mean value.

5 Conclusions

This report describes how ramping constraints on discharge were modeled in a prototype of the medium-term hydropower scheduling model ProdRisk, in the context of a price-taking producer operating in a liberalized market. The presence of ramping constraints may be motivated by ecological, physical, or even security reasons. Current versions of model input, with a time resolution of one hour or coarser, overestimate the flexibility to ramp discharge between consecutive time steps. In principle, this challenge can be met by facilitating a finer time resolution, but this will come at the cost of dramatically increased computation times. Since the medium-term scheduling is already a computationally time-consuming task that is often carried out on a daily basis, such dramatic changes are not welcome among model users. Thus, we suggested a new technique for approximating the cost associated with strictly following the ramping requirements.

In the proposed technique, the cost associated with following the maximum allowed ramping gradient between consecutive time steps is expressed as a function of decision variables. The reasoning is that when ramping up (resp. down) due to a higher (resp. lower) power price in the subsequent time step, an additional water volume needs to be discharged (resp. held back) in the prevailing time-step. This "misplaced" volume involves a cost corresponding to the lost revenue caused by increased generation at a lower price. We assume (i) that the exogeneous power prices motivate changes in discharge, (ii) ramping is carried out symmetrically around the shift between time steps, (iii) the "misplaced" volume is valuated according to the best efficiency, and (iv) the misplaced volume is not accounted for in reservoir balances. Through basic computational experiments we demonstrate that the technique provides solutions that are similar to what is achieved by significantly refining time-resolution, with a tendency to overestimate the cost of ramping.

Traditional ramping constraints on discharge and the new transition cost functionality were implemented in a research prototype of the ProdRisk model. The transition costs were added as additional cost components in the objective, using linear inequalities to approximate the quadratic expressions. The prototype was tested in two case studies, and basic validation of its performance and constraint handling is documented in the report. The case studies apply a rather coarse time resolution, favoring the transition cost functionality over the traditional constraints modelling in representing the ramping constraints. The results demonstrate that the transition cost functionality, forcing the model to reflect the maximum ramping capability in the transition between consecutive time-steps, significantly impacts the model's willingness to ramp. The computation times increased substantially with the addition of transition costs but are within what can be considered acceptable times.

Further work could improve the process of creating (a priori) and managing (during solution) linear inequalities approximating the transition cost to reduce computation time. Additional testing would also provide more insight in the interplay of discharge constraints and transition cost with other constraints, e.g., time-dependent minimum discharge, and the impact of the penalty cost hierarchy.

6 References

[1] A. Helseth, "Environmental Constraints in Seasonal Hydropower Scheduling - Survey and Feasibility," HydroCen Report nr.12, Trondheim, Norway, 2019.

[2] SINTEF Energi, "ProdRisk," https://www.sintef.no/programvare/prodrisk/.

[3] A. Gjelsvik, M. M. Belsnes and A. Haugstad, "An algorithm for stochastic medium-term hydrothermal scheduling under spot price uncertainty," in *Proc. 13th Power System Computation Conference*, Trondheim, 1999.

[4] A. Gjelsvik, B. Mo and A. Haugstad, "Handbook of Power Systems I," Springer, 2010, pp. 33-55.

[5] A. Helseth, M. Fodstad and B. Mo, "Optimal Medium-Term Hydropower Scheduling Considering Energy and Reserve Capacity Markets," *IEEE Transactions on Sustainable Energy,* vol. 7, pp. 934-942, 2016.

[6] A. Helseth, M. Fodstad and B. Mo, "Optimal Hydropower Maintenance Scheduling in Liberalized Markets," *IEEE Transactions on Power Systems,* vol. 33, pp. 6989-6998, 2018.

[7] A. Helseth, B. Mo, H. O. Hågenvik and L. E. Schäffer, "Hydropower Scheduling with State-dependent Discharge Constraints – an SDDP approach," *Journal of Water Resources Planning and Management,* vol. 148, 2022.

[8] A. Shapiro, "Analysis of stochastic dual dynamic programming method," *European Journal of Operational Research,* vol. 209, pp. 63-72, 2011.

[9] C. Ø. Naversen, M. Parvania, A. Helseth and H. Farahmand, "Continuous Hydrothermal Flexibility Coordination Under Wind Power Uncertainty," *IEEE Transactions on Sustainable Energy,* vol. 13, pp. 1900-1912, 2022.

[10] A. Helseth, S. Rex and B. Mo, "Approximating Ramping Constraints in Hydropower Scheduling," in *European Energy Markets*, Lappeenranta, Finland, 2023.

[11] A. Helseth, A. Gjelsvik, B. Mo and Ú. Linnet, "A model for optimal scheduling of hydro thermal systems including pumped-storage and wind power," *IET Generation, Transmission & Distribution,* vol. 7, pp. 1426-1434, 2013.

[12] M. N. Hjelmeland, J. Zou, A. Helseth and S. Ahmed, "Nonconvex Medium-Term Hydropower Scheduling by Stochastic Dual Dynamic Integer Programming," *IEEE Transactions on Sustainable Energy,* vol. 10, pp. 481-490, 2018.

A Experimental code

Below is the computer code in Julia/JuMP used to validate the transition cost functionality, as discussed in Section 3.4.

In file example.jl:

```
using JuMP
using CPLEX
using MathOptInterface
using Printf

const MOI = JuMP.MathOptInterface

#Constants
const Big = 1.0E10
const InfUB = 1.0E6
const TS = 168.0
const M3S2MM3WEEK = 0.6048

#Case data
NHOUR = 3.0
TAU = NHOUR/TS
NSeg = 2
MaxRes = 10.0
Inflow = 1.0
ResInit = 5.0
MaxDis = [20 10]
Eff = [2.0 1.8]
Price = [107.50 40.56 31.43 55.51 46.50 43.60 103.57 53.00 54.40 13.13 14.53 68.47
78.93 79.64 12.96 43.23 68.79 87.55 45.96 109.92 89.16 15.56 81.83 10.50 51.25
26.17 71.93 80.05 53.34 87.92 75.60 51.71 79.21 57.54 43.73 23.24 105.47 98.19
83.11 30.37 32.07 95.06 67.23 29.52 85.42 28.75 32.57 69.74 96.57 40.40 71.89 19.77
52.52 100.90 21.70 65.36]
NK = length(Price)

#Create TC-cuts
MAXRAMP = 10.0 #m3/s/hour
DISC = 0.05
u = collect(-MAXRAMP*NHOUR:DISC:MAXRAMP*NHOUR)
NCut = length(u)
println("NCut = ",NCut)

CR = zeros(Float64,NK)
a = zeros(Float64,NK,NCut)
b = zeros(Float64,NK,NCut)
for k = 2:NK
   DP = Eff[1,1]*abs(Price[k]-Price[k-1])
   CR[k] = DP/(8*MAXRAMP)
   for c = 1:NCut
      a[k,c] = 2*CR[k]*u[c]
      b[k,c] = -CR[k]*u[c]^2
   end
end

#Build and solve problems
include("hprob.jl")
SPQ = BuildProb(true,false)
SPA = BuildProb(false,true)
SP0 = BuildProb(false,false)

optimize!(SPQ)
optimize!(SPA)
```

```
optimize!(SP0)

@printf("%s     %12.2f     %12.2f     %12.2f     \n","Objective:     ",JuMP.objec-
tive_value(SPQ),JuMP.objective_value(SPA),JuMP.objective_value(SP0))
```

In file hprob.jl:

```
function BuildProb(LQuad,LApprox)
   M = Model(CPLEX.Optimizer)
   set_optimizer_attribute(M,"CPX_PARAM_SCRIND",0)
   set_optimizer_attribute(M,"CPX_PARAM_PREIND",0)

   @variable(M,0.0 <= res[k=1:NK] <= MaxRes)                        #Mm3
   @variable(M,0.0 <= disSeg[iSeg=1:NSeg,k=1:NK] <= MaxDis[iSeg])  #m3/s
   @variable(M,0.0 <= dis[k=1:NK])                                  #m3/s
   @variable(M,0.0 <= spi[k=1:NK] <= InfUB)                         #m3/s
   @variable(M,0.0 <= ghy[k=1:NK] <= InfUB)                         #MW
   @variable(M,0.0 <= y[k=2:NK] <= InfUB)

   #OBJECTIVE [MU]
   if LQuad
      @objective(M,MathOptInterface.MAX_SENSE,
        sum(Price[k]*ghy[k] for k=1:NK)-sum(CR[k]*(dis[k]-dis[k-1])^2 for k=2:NK))
   elseif LApprox
      @objective(M,MathOptInterface.MAX_SENSE,
        sum(Price[k]*ghy[k] for k=1:NK)-sum(y[k] for k=2:NK))
   else
      @objective(M,MathOptInterface.MAX_SENSE, sum(Price[k]*ghy[k] for k=1:NK))
   end


   #INITIAL RESERVOIR BALANCE [MM3]
   @constraint(M,resbal0,
      res[1]+TAU*M3S2MM3WEEK*dis[1]+spi[1] == TAU*Inflow+ResInit)

   #RESERVOIR BALANCE [MM3]
   @constraint(M,resbal[k=2:NK],
      res[k]-res[k-1]+TAU*M3S2MM3WEEK*dis[k]+spi[k] == TAU*Inflow)

   @constraint(M,sumdis[k=1:NK],dis[k]-sum(disSeg[iSeg,k] for iSeg=1:NSeg) == 0)

   #HYDROPOWER GENERATION [MWh]
   @constraint(M,hygen[k=1:NK],
      ghy[k]-NHOUR*sum(Eff[iSeg]*disSeg[iSeg,k] for iSeg=1:NSeg) == 0.0)

   if LApprox
      @constraint(M,ycut[k=2:NK,c=1:NCut],
         y[k] >= a[k,c]*(dis[k]-dis[k-1])+b[k,c])
   end

   return M
end
```

www.hydrocen.no

ISSN: 2535-5392
ISBN: 978-82-93602-40-8

HydroCen
v/ Vannkraftlaboratoriet, NTNU
Alfred Getz vei 4,
Gløshaugen, Trondheim

www.hydrocen.no
HydroCen
@FMEHydroCen