

Electronic Companion

A Standard SDDP

We provide a brief presentation of SDDP in its standard form, and refer to [3] for more details. For standard SDDP, the assumption of stagewise independence is crucial.

Assumption 3. *The stochastic process $(\xi_t)_{t \in [T]}$ which appears in the RHS of problem (MSLP) is stagewise independent, that is, for all $t \in [T]$:*

$$\xi_t = b_t(\xi_{[t-1]}, \eta_t) \equiv \eta_t.$$

Under this assumption, all expectations in (5) become unconditional, and $Q_t(\cdot, \cdot)$ and $\mathcal{Q}_t(\cdot)$ do no longer depend on $\xi_{[t-1]}$. With x_{t-1} being the only argument of $\mathcal{Q}_t(\cdot)$, convexity is assured, and thus linear cuts are sufficient for a tight and valid approximation. We can then reformulate subproblems (6) as linear programs

$$\underline{Q}_t^{R_{t+1}}(x_{t-1}, \xi_t) = \begin{cases} \min_{x_t, \theta_{t+1}} & c_t^\top x_t + \theta_{t+1} \\ \text{s.t.} & W_t x_t = -T_{t-1} x_{t-1} + \xi_t \\ & -\beta_{r,t+1}^\top x_t + \theta_{t+1} \geq \alpha_{r,t+1}, \quad r \in R_{t+1} \\ & x_t \geq 0 \end{cases} \quad (41)$$

using the relation

$$\mathfrak{Q}_{t+1}^{R_{t+1}}(x_t) = \min \left\{ \theta_{t+1} : \theta_{t+1} \geq \beta_{r,t+1}^\top x_t + \alpha_{r,t+1}, \quad r \in R_{t+1} \right\}$$

and $|R_{t+1}| \in \mathbb{N}$. Here, R_{t+1} denotes the index set of cuts that have been constructed to approximate $\mathcal{Q}_t(\cdot)$ so far. Quantities $\beta_{r,t+1}$ and $\alpha_{r,t+1}$ are cut-specific parameters whose computation we explain in detail in the remainder of this section. In this setting, cuts are shared between scenarios naturally because, independent of the history $\xi_{[t-1]}$, all nodes in the scenario tree share the same expected value functions.

SDDP consists of two main steps in each iteration i , a *forward pass* and a *backward pass* through the stages $t \in [T]$. In the forward pass, a sequence of incumbents $(x_t^i)_{t \in [T]}$ is generated, at which then new cuts are constructed in the following backward pass. By this alternating procedure, SDDP iteratively improves the cut approximations $\mathfrak{Q}_t(\cdot)$ of $\mathcal{Q}_t(\cdot)$. Alg. 1 provides a pseudo-code for standard SDDP.

We now provide a more detailed and technical look at the algorithmic steps.

A.1 Forward Pass

At the start of each iteration i , the first-stage subproblem

$$\begin{cases} \min_{x_1} & c_1^\top x_1 + \mathfrak{Q}_2^i(x_1) \\ \text{s.t.} & W_1 x_1 = h_1 \\ & x_1 \geq 0 \end{cases} \quad (42)$$

Algorithm 1 SDDP

Input: Problem (MSLP) satisfying Assumptions 1, 2, 3. Bounds $\underline{\theta}_t, t = 2, \dots, T$. Stopping criterion.

1: Initialize cut approximations with $\theta_t \geq \underline{\theta}_t$ for all $t = 2, \dots, T$.

2: Initialize lower bound with $\underline{v}^0 = -\infty$.

3: Set iteration counter to $i \leftarrow 0$.

4: **while** Stopping criterion not satisfied **do**

5: Set $i \leftarrow i + 1$.

Forward Pass

6: Solve the first-stage problem (42) to obtain incumbent x_1^i .

7: **for** stages $t = 2, \dots, T$ **do**

8: Sample one realization ξ_t^i from $\{\xi_t^{(1)}, \dots, \xi_t^{(q_t)}\}$.

9: Solve the stage- t subproblem (41) associated with $\underline{Q}_t^i(x_{t-1}^i, \xi_t^i)$ to obtain incumbents x_t^i .

10: **end for**

Backward Pass

11: **for** stages $t = T, \dots, 2$ **do**

12: **for** realizations $j = 1, \dots, q_t$ **do**

13: Solve the updated stage- t subproblem (41) associated with $\underline{Q}_t^{i+1}(x_{t-1}^i, \xi_t^{(j)})$. Store the optimal dual variables π_{tj}^i and $\rho_{rtj}^i, r \in R_{t+1}$.

14: **end for**

15: Use relations (43), (44), (45) to create an optimality cut for $\underline{Q}_t(\cdot)$.

16: Update the cut approximation $\underline{\Omega}_t^i(\cdot)$ to $\underline{\Omega}_t^{i+1}(\cdot)$ using relation (46).

17: **end for**

18: Solve the updated first-stage problem (42) to obtain a lower bound $\underline{v}^i$.

19: **end while**

Output: (Approximately) optimal feasible policy for (MSLP) defined by x_1^i and $\underline{\Omega}_t^i(\cdot), t = 2, \dots, T$.

is solved, which yields the incumbent x_1^i . Note that for simplicity, we use upper indices i instead of R_{t+1}^i for the value functions and their approximations.

Afterwards, for each stage $t = 2, \dots, T$ we iteratively draw one sample ξ_t^i from the set $\{\xi_t^{(1)}, \dots, \xi_t^{(q_t)}\}$ of all realizations using random sampling, and then solve the subproblem (41) associated with $\underline{Q}_t^i(x_{t-1}^i, \xi_t^i)$; this subproblem contains the cut approximation $\underline{\Omega}_{t+1}^i(\cdot)$. This way, we obtain incumbent x_t^i that can be used as a parameter in the following stage. Over all stages, this yields a sequence of incumbents $(x_t^i)_{t \in [T]}$.

Remark A.1. In standard SDDP it is also common to sample more than one realization per stage, meaning that we run several forward passes and obtain several sequences of incumbents per iteration ([3]). All our results in this paper naturally extend to this case, but for notational convenience, we restrict to the case of one sample per iteration.

Remark A.2. By evaluating the incumbents $(x_t^i)_{t \in [T]}$ in the objective $\sum_{t \in [T]} c_t(\xi_t^i)^\top x_t^i$ for the chosen sample path, we obtain a simulated objective value. This may not provide reasonable information on v^* , though, as it depends on only one specific sample.

By repeating the same evaluation for a large number of sample paths and taking the mean, at least an unbiased estimator of an upper bound for v^* can be computed, a so-called statistical upper bound. This kind of simulation is often executed after SDDP has terminated to assess the quality of the obtained policy. It can either be done using the same realizations of η_t as in SDDP (in-sample simulation) or with realizations that were not considered within the algorithm (out-of-sample simulation).

A.2 Backward Pass

In the backward pass, new cuts for the expected value functions $\mathcal{Q}_t(\cdot)$ are generated at the incumbents x_{t-1}^i , thus improving their approximation from $\mathfrak{Q}_t^i(\cdot)$ to $\mathfrak{Q}_t^{i+1}(\cdot)$.

The backward pass starts at stage T . Here, we consider the subproblems (41) for the incumbent x_{T-1}^i computed in the forward pass, but all possible noise realizations $\xi_T^{(j)}, j = 1, \dots, q_T$. That is, we consider functions $\underline{Q}_T^{i+1}(x_{T-1}^i, \xi_T^{(j)})$ for all $j = 1, \dots, q_T$.

As $\mathfrak{Q}_{T+1}^{i+1}(\cdot) = \mathcal{Q}_{T+1}(\cdot) \equiv 0$, we have $\underline{Q}_T^{i+1}(\cdot, \xi_T^{(j)}) = Q_T(\cdot, \xi_T^{(j)})$. The LP dual to the subproblem (41) associated with $\underline{Q}_T^{i+1}(x_{T-1}^i, \xi_T^{(j)})$ is

$$\begin{cases} \min_{x_T} & \pi_T^\top (-T_{T-1}x_{T-1}^i + \xi_T^{(j)}) \\ \text{s.t.} & W_T^\top \pi_T = c_T \\ & \pi_T \geq 0. \end{cases}$$

Let π_{Tj}^* be a dual optimal solution. Then, by strong duality

$$Q_T(x_{T-1}^i, \xi_T^{(j)}) = (\pi_{Tj}^*)^\top (-T_{T-1}x_{T-1}^i + \xi_T^{(j)}).$$

As the dual feasible set is independent of x_{T-1} , π_{Tj}^* remains feasible, although not necessarily optimal, for any $x_{T-1} \in \mathbb{R}^{n_{T-1}}$. Therefore, we obtain the cut

$$\begin{aligned} Q_T(x_{T-1}, \xi_T^{(j)}) &\geq (\pi_{Tj}^*)^\top (-T_{T-1}x_{T-1} + \xi_T^{(j)}) \\ &= \underbrace{-(\pi_{Tj}^*)^\top T_{T-1}}_{=: \beta_{Tj}^\top} x_{T-1} + \underbrace{(\pi_{Tj}^*)^\top \xi_T^{(j)}}_{=: \alpha_{Tj}} \\ &= \beta_{Tj}^\top x_{T-1} + \alpha_{Tj}. \end{aligned}$$

β_{Tj} is called *cut gradient* and α_{Tj} is called *cut intercept* or *constant*.

Executing this for all $j = 1, \dots, q_T$, and then taking expectations over cuts (A.2), we obtain a cut

$$\mathcal{Q}_T(x_{T-1}) \geq \beta_T^\top x_{T-1} + \alpha_T$$

for the expected value function $\mathcal{Q}_T(\cdot)$, with

$$\beta_T := \sum_{j=1}^{q_T} p_{Tj} \beta_{Tj}, \quad \alpha_T := \sum_{j=1}^{q_T} p_{Tj} \alpha_{Tj}.$$

With this new cut, the cut approximation $\mathfrak{Q}_T^i(\cdot)$ is updated to

$$\mathfrak{Q}_T^{i+1}(x_{T-1}) = \max \left\{ \mathfrak{Q}_T^i(x_{T-1}), \beta_T^\top x_{T-1} + \alpha_T \right\},$$

and thus the updated cut index set satisfies $R_T^{i+1} = \{1, \dots, i+1\} = R_T^i \cup \{i+1\}$.

In the same way, for stages $t = T - 1, \dots, 2$, cuts for $\mathcal{Q}_t(\cdot)$ can be constructed by solving subproblems (41) for the incumbents x_{t-1}^i and all realizations $\xi_t^{(j)}, j = 1, \dots, q_t$. Importantly, by going backwards through the stages, at stage t we can already factor in the cuts that have been constructed at the following stage $t + 1$, thus using a better approximation as the basis to construct a new cut. This means that we consider $\mathfrak{Q}_{t+1}^{i+1}(\cdot)$ and by that $\underline{Q}_t^{i+1}(\cdot, \cdot)$ with index $i + 1$ in the backward pass of iteration i .

The objective function to the dual of the stage- t subproblem (41) associated with $\underline{Q}_t^{i+1}(x_{t-1}^i, \xi_t^{(j)})$ is

$$\pi_t^\top (-T_{t-1}x_{t-1}^i + \xi_t^{(j)}) + \sum_{r \in R_{t+1}^{i+1}} \rho_{rt} \alpha_{r,t+1}.$$

Let $(\pi_{tj}^*, \rho_{tj}^*)$ be a dual optimal solution. By strong duality, we obtain

$$\underline{Q}_t^{i+1}(x_{t-1}^i, \xi_t^{(j)}) = (\pi_{tj}^*)^\top (-T_{t-1}x_{t-1}^i + \xi_t^{(j)}) + \sum_{r \in R_{t+1}^{i+1}} \rho_{rtj}^* \alpha_{r,t+1}.$$

As for stage T , the dual feasible set is independent of x_{t-1} . Therefore, $(\pi_{tj}^*, \rho_{tj}^*)$ remains feasible, although not necessarily optimal, for any $x_{t-1} \in \mathbb{R}^{n_{t-1}}$. We obtain

$$\underline{Q}_t^{i+1}(x_{t-1}, \xi_t^{(j)}) \geq (\pi_{tj}^*)^\top (-T_{t-1}x_{t-1} + \xi_t^{(j)}) + \sum_{r \in R_{t+1}^{i+1}} \rho_{rtj}^* \alpha_{r,t+1}.$$

Finally, as $\underline{Q}_t^{i+1}(\cdot, \xi_t^{(j)})$ underestimates $\mathcal{Q}_t(\cdot, \xi_t^{(j)})$, we obtain the cut

$$\begin{aligned} \mathcal{Q}_t(x_{t-1}, \xi_t^{(j)}) &\geq (\pi_{tj}^*)^\top (-T_{t-1}x_{t-1} + \xi_t^{(j)}) + \sum_{r \in R_{t+1}^{i+1}} \rho_{rtj}^* \alpha_{r,t+1} \\ &= \underbrace{-(\pi_{tj}^*)^\top T_{t-1} x_{t-1}}_{=: \beta_{tj}^\top} + \underbrace{(\pi_{tj}^*)^\top \xi_t^{(j)} + \sum_{r \in R_{t+1}^{i+1}} \rho_{rtj}^* \alpha_{r,t+1}}_{=: \alpha_{tj}} \\ &= \beta_{tj}^\top x_{t-1} + \alpha_{tj}. \end{aligned} \tag{43}$$

Executing this for all $j = 1, \dots, q_t$, and then taking expectations over cuts (43), we obtain a cut

$$\mathcal{Q}_t(x_{t-1}) \geq \beta_t^\top x_{t-1} + \alpha_t \tag{44}$$

for the expected value function $\mathcal{Q}_t(\cdot)$, with

$$\beta_t := \sum_{j=1}^{q_t} p_{tj} \beta_{tj}, \quad \alpha_t := \sum_{j=1}^{q_t} p_{tj} \alpha_{tj}. \tag{45}$$

With this new cut, the cut approximation $\mathfrak{Q}_t^i(\cdot)$ is updated to

$$\mathfrak{Q}_t^{i+1}(x_{t-1}) = \max \left\{ \mathfrak{Q}_t^i(x_{t-1}), \beta_t^\top x_{t-1} + \alpha_t \right\}. \tag{46}$$

At the first stage, the updated first-stage subproblem (42) is solved with optimal value $\underline{v}^i$. As $\mathfrak{Q}_2^{i+1}(\cdot)$ is a lower approximation of $\mathcal{Q}_2(\cdot)$, $\underline{v}^i$ is a valid lower bound to the optimal value v^* of (MSLP). In contrast, we are not guaranteed to obtain a valid upper

bound for v^* in SDDP, as we only consider one sampled scenario per iteration instead of evaluation over the entire scenario tree.

After each iteration of SDDP, one or several stopping criteria are checked, such as statistical stopping criteria, reaching a predefined number of iterations or stalling of the lower bounds $\underline{v}^i$ ([3]). If SDDP does not stop, a new iteration $i + 1$ is started.

As mentioned before, the polyhedral dual feasible sets at each stage t are independent of x_{t-1} , thus they possess finitely many extreme points. Using this reasoning inductively starting at stage T and assuming that we restrict to optimal dual basic solutions in the cut generation process, it can be shown that only finitely many different cuts can be constructed. Additionally, the cuts are *tight*, *i.e.*, evaluating the right side of (44) at x_{t-1}^i , yields $\mathbb{E}_{\xi_t}[Q_t^{i+1}(x_{t-1}^i, \xi_t)]$, which implies that the approximations eventually become exact. Combining this with the properties of the sampling procedure, almost sure finite convergence of SDDP to an optimal policy can be proven [8].

B Long-term Hydrothermal Scheduling Problem

The LTHS problem with 4 EERs and 95 generators that we consider in our computational experiments in Sect. 5 can be formulated as follows [9, 10]:

$$\begin{aligned}
\min \quad & \sum_{t=1}^T \beta^{t-1} \left(\sum_{k \in \mathcal{K}} \left(\sum_{j \in \mathcal{G}_k} c_{tj} g_{tj} \right) + \delta_{tk} r_{tk} \right) \\
\text{s.t.} \quad & v_{tk} = v_{t-1,k} + a_{tk} + q_{tk} + s_{tk}, & t = 1, \dots, T, k \in \mathcal{K} \\
& q_{tk} + \sum_{j \in \mathcal{G}_k} g_j + r_{tk} + \sum_{\ell \in \mathcal{K}} (f_{t\ell k} - f_{tk\ell}) = d_{tk}, & t = 1, \dots, T, k \in \mathcal{K}, \\
& 0 \leq v_{tk} \leq \bar{v}_k, & t = 1, \dots, T, k \in \mathcal{K} \\
& 0 \leq q_{tk} \leq \bar{q}_k, & t = 1, \dots, T, k \in \mathcal{K} \\
& s_{t5} = 0, 0 \leq s_{tk}, & t = 1, \dots, T, k \in \mathcal{K} \\
& \underline{g}_{tj} \leq g_{tj} \leq \bar{g}_{tj}, & t = 1, \dots, T, j \in \mathcal{G} \\
& 0 \leq r_{tk} \leq \bar{r}_{tk}, \underline{f}_{tk\ell} \leq f_{tk\ell} \leq \bar{f}_{tk\ell}, & t = 1, \dots, T, k \in \mathcal{K}, \ell \in \mathcal{K}.
\end{aligned}$$

Here $\mathcal{K}$ denotes the set of five sub-systems in the overall power system. The variables v_{tk} , a_{tk} , q_{tk} and s_{tk} denote the water level, inflow, hydro power generation and spillage for the reservoirs associated with systems $k \in \mathcal{K}$ at stage t . Note that only four of these systems contain a hydro reservoir, so for the fifth one the bounds $\bar{v}_k$ and $\bar{q}_k$ are zero, and the spillage variable is always set to zero as well. The first set of constraints describes the water balance of the hydro reservoirs. The initial levels v_{0k} are given, while the inflow vector a_t is considered uncertain and described by a PAR process.

The set $\mathcal{G}_k$ contains all the thermal generators in system $k \in \mathcal{K}$. The variables g_{tj} denote the thermal power generation of generator $j \in \mathcal{G}$ at stage t , with lower limits $\underline{g}_{tj}$ and upper limits $\bar{g}_{tj}$. The second set of constraints ensures the load balance for all systems $k \in \mathcal{K}$ at all stages t by satisfying the load d_{tk} using thermal generation, hydro power generation and energy exchange between systems which is represented by variables $f_{t\ell k}$ for $k, \ell \in \mathcal{K}$. Additionally, we may allow for a load deficit, represented by variable r_{tk} . Both types of variables are bounded as well.

In the objective function we minimize costs, which are discounted using the annual discount rate β . The costs include the generation costs by thermal generators, taken

into account using specific costs c_{tj} , as well as the penalization of load curtailment, *i.e.*, a positive deficit r_{tk} . We assume that the latter costs are piecewise linear so that they increase with the amount of curtailment. This is modeled using the piecewise linear function δ_{tk} . It is assumed that hydro generation does not cause any costs.

C Inflow Modeling

The following procedure is applied to all four EERs (SE, S, N, NE) independently, as we do not take spatial inflow dependencies into account. The performed model fitting and validation steps are standard in time series analysis as part of the Box-Jenkins method [1]. However, we follow an adapted variant for PAR models, which is mostly based on [5]. In that regard, our inflow modeling deviates from [10], where the same historical data is used, but the model selection (*i.e.*, identifying the lag order of the process) and model validation are executed for a single non-seasonal AR model and not specifically for a PAR model. For details, we refer to our GitHub project at <https://github.com/ChrisFue10R/LogLinearSDDP.jl>, which also contains the preparation of the inflow data.

C.1 Data Preparation

Let X_t denote the true historical inflow time series and let $Y_t = \log(X_t)$. The monthly box-plot diagram in Figure 1a shows that Y_t follows a seasonal, *i.e.*, monthly pattern. To account for this pattern, we perform a deseasonalization using monthly means μ_t and standard deviations σ_t , which yields a time series of the form

$$Z_t := \frac{Y_t - \mu_t}{\sigma_t}, \quad (47)$$

see also Remark 4.1 (e). The monthly means and standard deviations satisfy $\mu_t = \mu_{t+12}, \sigma_t = \sigma_{t+12}$. The box-plot diagram in Figure 1b for Z_t shows that the monthly pattern is removed.

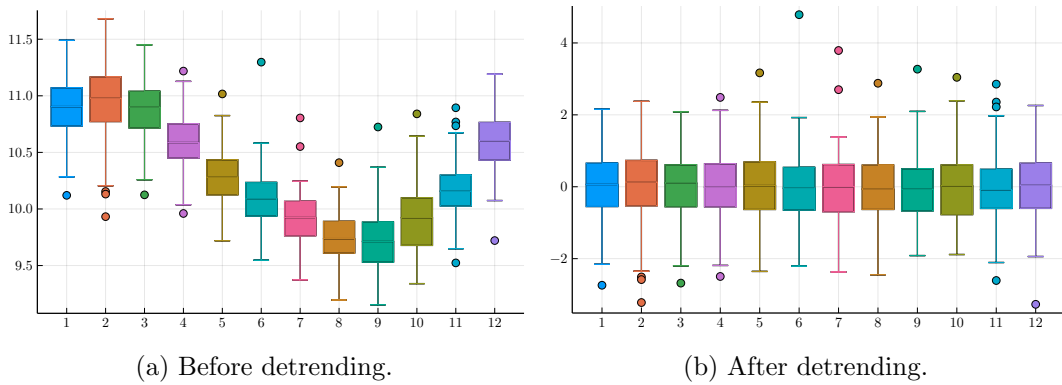


Figure 1: Boxplot of Y_t before and after detrending for system SE.

C.2 Stationarity Analysis

To evaluate the time series Z_t for stationarity, we analyze its partial autocorrelation function (PACF) and perform an augmented Dickey-Fuller (ADF) test. The PACF

shows a fast decline, which is an indicator for stationarity. The ADF test rejects the null hypothesis of a unit root at a 5% significance level, which again implies stationarity. Note that stationarity is a prerequisite for most of the following fitting and validation techniques.

C.3 Model Selection and Fitting

Next, we split the data into a training data set (from 1931 to 1994) and a test data set (from 1995 to 2009). The model selection, fitting and validation via hypothesis tests are only using the training data. Later on, we use the test data for out-of-sample validation.

As we consider a log-linear PAR model of type (22) to model X_t , the time series Y_t follows a standard PAR model. We assume a monthly resolution, so the coefficients of the AR model may vary with each month. This can be interpreted as fitting separate models for each month, only considering the corresponding lagged data, see [5, 6] for details. To identify a reasonable lag order for each of these models, for each month, we first estimate the *periodic* autocorrelation function (ACF) [5]. Additionally, for each month, we fit AR models with lag orders between 1 and 12 and use the Bayesian Information Criterion (BIC) to compare these models. The latter approach yields the following lag orders for the components of the model, summarized in Table 1.

Table 1: Lag orders identified using the BIC.

System	Month											
	1	2	3	4	5	6	7	8	9	10	11	12
SE	1	2	1	1	1	1	3	1	1	1	1	1
S	1	2	1	2	1	1	1	1	1	1	1	1
NE	1	2	1	1	1	3	10	1	1	1	2	1
N	1	4	1	1	1	10	3	4	5	1	1	1

In conformity with Sect. 5, we refer to the model using these lag orders as **LOG-BIC**. For comparison, we also consider a PAR model of lag order 1 for Y_t , which we refer to as **LOG-1**. Moreover, we consider a linear model that is obtained by linearizing the log-linear PAR model for X_t using a first-order Taylor approximation (cf. Sect. 4.2). Varying coefficients have been reported for this model and the given inflow data in the literature [2, 7, 9]. We therefore apply two different variants: For **LIN-SHA** we use the model coefficients as reported in [9], for **LIN-FIT** we use coefficients obtained using the same fitting and validation steps that we describe for the log-linear models below. We should mention that in contrast to our approach, **LIN-SHA** uses a multivariate normal distribution for the error terms.

Recall that we assume a constant lag order in our cut formulas, see Sect. 4, so the values in Table 1 imply the choice $p = 10$, whereas all other models satisfy $p = 1$.

C.4 Model Validation – Diagnostic

For validation of the monthly models constituting **LOG-BIC** (or **LOG-1**, **LIN-FIT**, respectively), we perform several diagnostic tests and analyses.

We observe that all monthly models exhibit a reasonable goodness of fit (adjusted $\mathbb{R}^2 > 0.35$) and autoregression coefficients that are significantly different from zero (as validated using a t-test for lag-order $p = 1$ or an F-test for lag orders $p > 1$, respectively).

We then analyze the residuals of the PAR model. First, we check for heteroscedasticity using a scatter plot of the residuals (this can be done on a monthly basis or for all residuals of the PAR model combined). The plots give no indication of variance changes in the residuals.

Second, we check the residuals for autocorrelation. To this end, we estimate the periodic residual autocorrelation function (RACF) [5]. For all months and all systems, we see no indication of significant autocorrelation. Still, we perform a Portmanteau test, as proposed in [5], to statistically assess the hypothesis of no autocorrelation. In most cases, this hypothesis cannot be rejected.

Finally, we check the residuals for normal distribution. Not only does the validity of the previous test results rely on normally distributed residuals, we also may assume the stagewise independent error term $\boldsymbol{\eta}_t$ appearing in formula (22) to be normally distributed in this case. For our analysis, we use histograms, qq-plots and different hypothesis tests (Jarque-Bera test, Anderson-Darling test, Kolmogorov-Smirnov test). It turns out that in many cases, the hypothesis of normality is debatable and can be rejected statistically. Importantly, we make the same observation when we fit the linearized model by Shapiro et al. [10] to the data (note that in [10] and the related technical report [9], model validation is only performed for the full time series, but not for the actual periodical model).

Since using normally distributed error terms $\boldsymbol{\eta}_t$ still yields reasonable forecasting and simulation results, see the following subsections, in our computational tests in Sect. 5, we nevertheless follow the normality hypothesis. In the future, a more elaborate PAR model may be applied to the LTHS problem.

C.5 Model Validation – Forecasting

To further validate our fitted models, we use them for forecasts and compare them to the historical data. More precisely, we use the fitted models (without error term) together with the historical data to compute one-step ahead forecasts for the training horizon (in-sample forecast) and the test horizon (out-of-sample forecast). For the obtained point forecasts we remove the deseasonalization and take the exponential function to obtain forecasts for the original time series X_t . Note that in general, simply taking the exponential function of forecasts for a logarithmized time series yields biased forecasts for the original time series, which can be avoided using a correction factor [4]. However, without this correction we observe better forecasting results. The mean absolute forecasting errors for the test set data are presented in Table 2. For comparison, also the errors for LIN-FIT and LIN-SHA are presented. As we can see, the linearized models show smaller one-step ahead forecasting errors than the log-linear ones, but relative to the mean value of the original time series, the differences are subtle.

Table 2: One-step-ahead forecasting MAE (mean absolute error) on the test data.

	SE	S	NE	N
LOG-BIC	4888.3	4085.0	1309.7	978.1
LOG-1	4879.7	4106.3	1309.7	1009.5
LIN-FIT	4903.5	4047.8	1336.5	968.7
LIN-SHA	4853.3	3913.4	1392.2	958.2
Mean historical value	34805.9	10118.5	6496.1	6041.4

C.6 Model Validation – Simulation

As a last step, that is most relevant for our application of SDDP, we use the fitted models and $N(0, (\hat{\sigma}_t^e)^2)$ -distributed errors η_t together with some historical starting values to generate 1000 alternative scenarios (sample paths) of 79 years length. Here, $(\hat{\sigma}_t^e)^2$ denotes the sample variance of the residuals. As before, we remove the deseasonalization and take the exponential function for each value in the obtained time series. Then, we compare the statistical properties of the obtained series with the historical data. More precisely, we compare the monthly means and standard deviations. This is exemplified in Figures 2 to 5, which show box plots of the means and standard deviations for the systems SE and S. The historical monthly values are indicated by black crosses in each plot. Again, we consider all four different types of models.

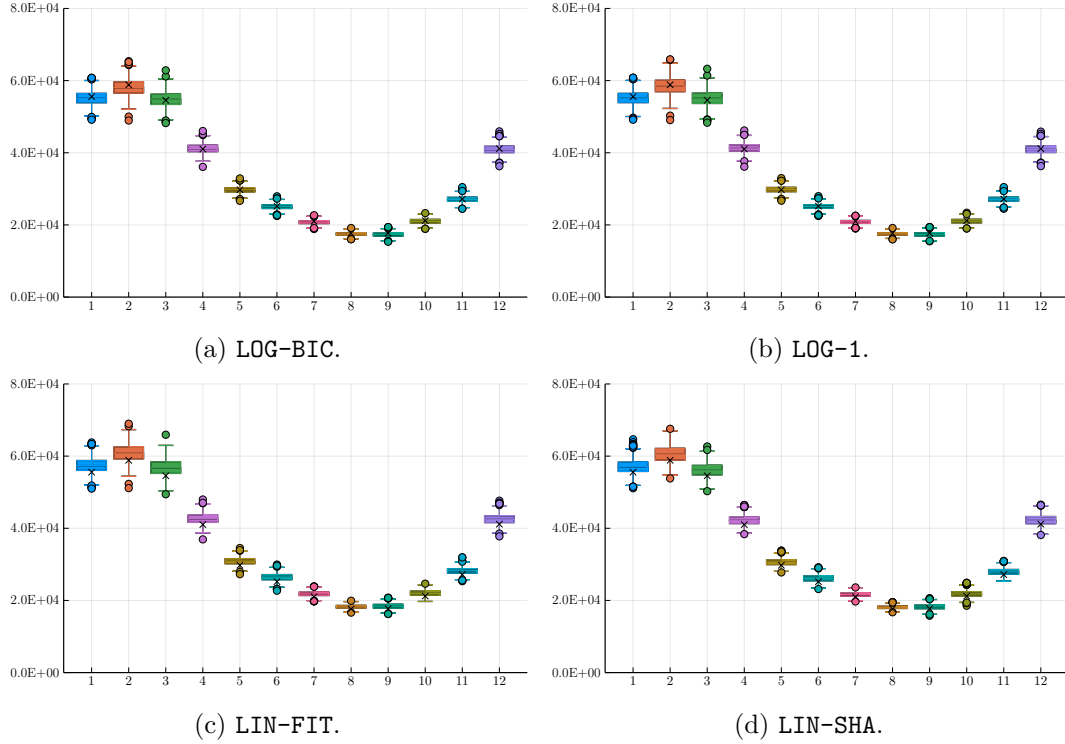


Figure 2: Boxplot of monthly means for system SE over 1000 sample paths.

For system SE, the simulated paths provide a good fit for the statistical properties observed in the historical data. The log-linear models seem to match the true monthly means slightly better. Especially in the first three months, the historical inflow means are consistently exceeded by the inflows from the linearized models, which is not the case for the log-linear models.

For system S, fitting seems most difficult among all four systems. Even though the linearized models provide a reasonable fit to the mean and standard deviation observed in the historical data, the historical values are again matched better using the log-linear models. In particular, we do not observe the systematic underestimation of the historical mean that is apparent for the linearized case. Additionally, the log-linear models lead to less variability in the observed means and standard deviations over all sample paths.

In Tables 3 and 4 we provide some additional information on our simulation comparison for all four systems. Table 3 addresses the monthly averages and their deviation

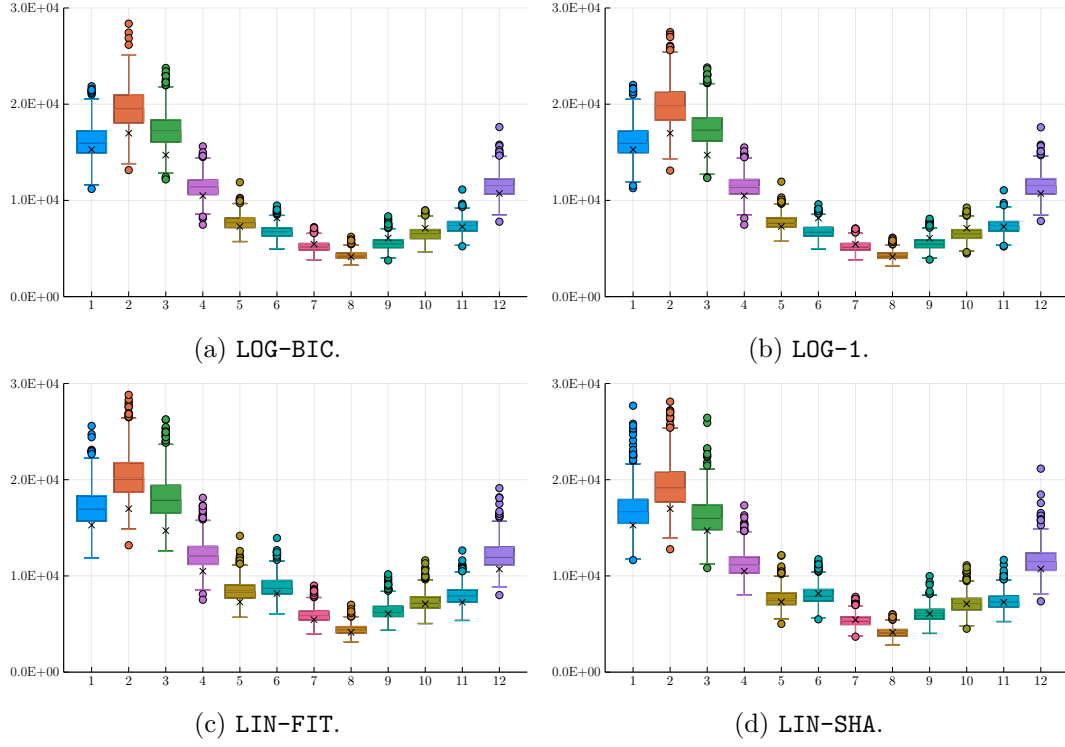


Figure 3: Boxplot of monthly standard deviations for system SE over 1000 sample paths.

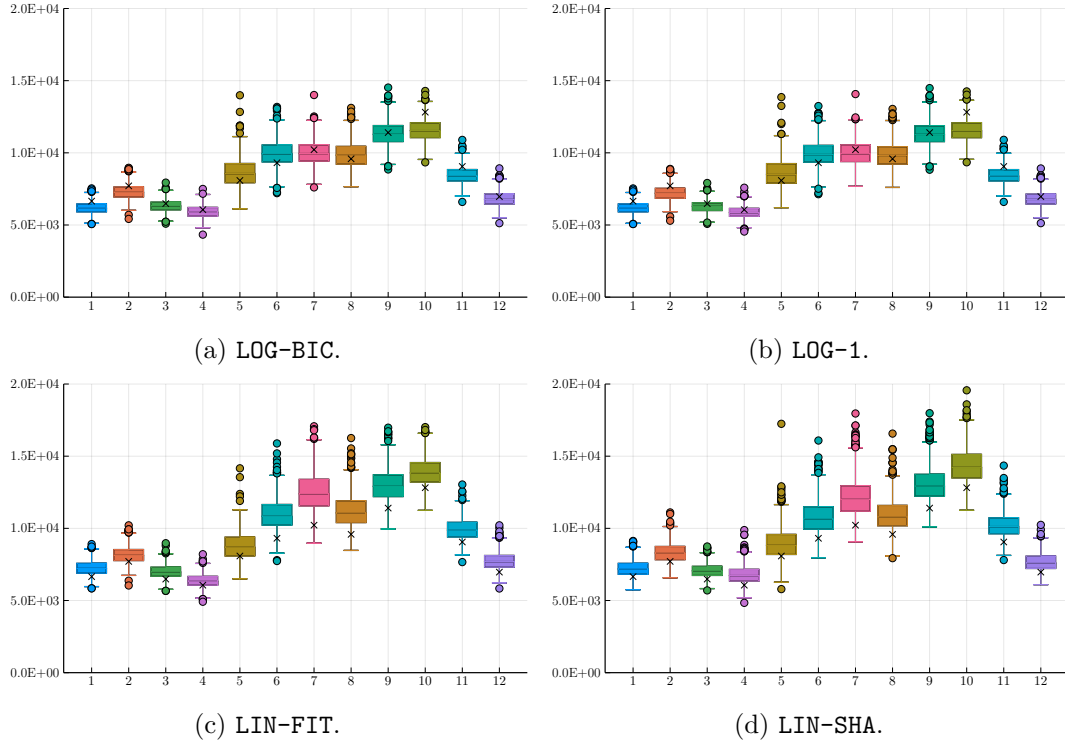


Figure 4: Boxplot of monthly means for system S over 1000 sample paths.

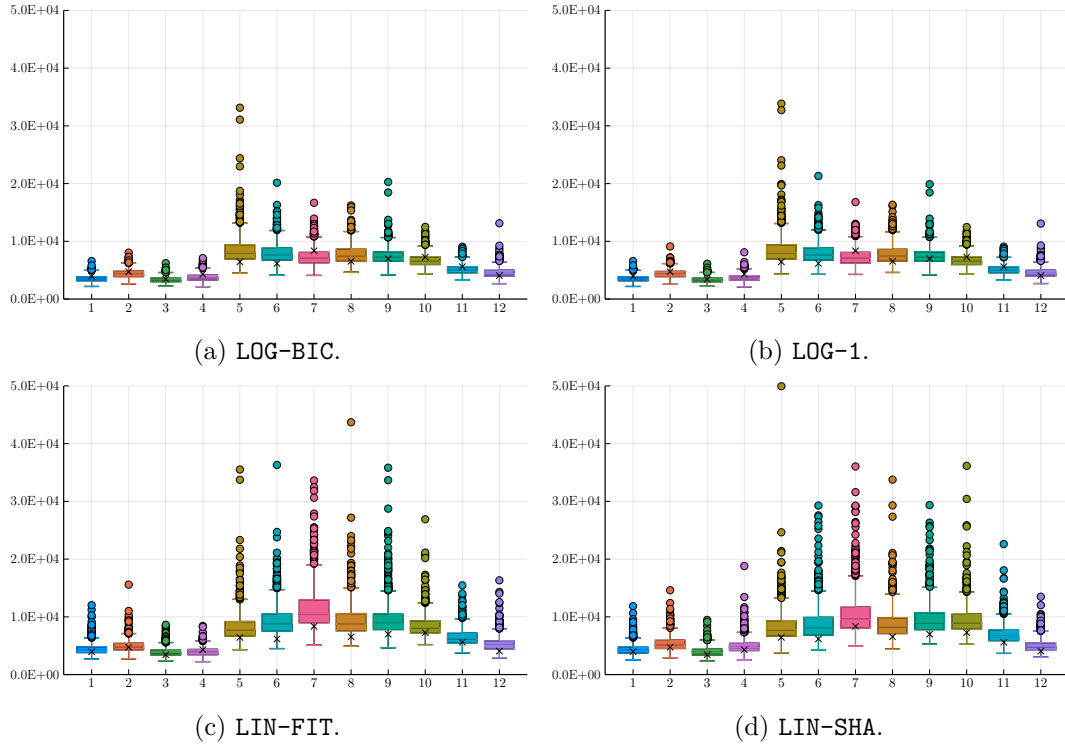


Figure 5: Boxplot of monthly standard deviations for system S over 1000 sample paths.

from the historical monthly averages. It presents the average of these deviations. As from the boxplots, we can see that the log-linear models on average provide a more accurate fit for the historical time series, except for system NE where the deviations are similar.

Table 4 presents similar results, but for deviations from the monthly averages obtained using LOG-BIC. Hence, it gives a hint on the relative differences in the inflow levels between the log-linear and the linearized models. We observe a considerable difference in the average inflow level, especially for systems SE and S. As SE is by far the reservoir with the largest capacity, we conclude that the linearized models yield significantly larger *total* inflows as well.

Table 3: Averages of deviation (in %) of monthly averages from historical monthly averages over 1000 sample paths.

	SE	S	NE	N
LOG-BIC	-0.6	-1.8	5.6	1.6
LOG-1	-0.4	-2.1	5.6	1.8
LIN-FIT	3.8	11.4	5.2	4.7
LIN-SHA	2.9	3.2	5.9	4.1

Importantly, these difference in the inflow levels, as well as the accuracy of simulating inflows with similar statistical properties to the historical data, do still occur if we use the exact same parameters in the log-linear and linearized models (instead of those obtained in the fitting process). These differences seem to be directly linked to the difference between the log-linear model formulas (24) and their Taylor approximations.

Table 4: Averages of deviation (in %) of monthly averages from LOG-BIC monthly averages over 1000 sample paths.

	SE	S	NE	N
LOG-1	0.2	-0.3	0.0	0.2
LIN-FIT	4.5	13.6	-0.4	3.1
LIN-SHA	3.5	5.3	0.3	2.4

We should note that similar observations have been already made by Löhndorf and Shapiro in [7].

D Computational Experiments

D.1 Inflow Data

Fig. 6 and 7 present the inflows for in-sample and out-of-sample simulations for LOG-1 and LIN-SHA data, respectively. It can be observed that the out-of-sample inflows tend to slightly exceed the in-sample ones, on average by about 0.3% in both cases.

Fig. 8 compares the in-sample inflows for LOG-1 and LIN-SHA in one plot. It can be observed that the inflows from the linearized model constantly exceed those from the log-linear model. An overview on some inflow statistics is provided in Table 5.

Table 5: Inflow means over 2000 scenario paths and 5 runs for in-sample (IS) and out-of-sample (OOS) simulations. Parentheses show deviations from LOG-BIC values.

		SE	S	NE	N	Total sum
LOG-BIC	IS	33801.1	8569.7	8867.0	6258.1	57495.9
LOG-1	IS	34007.9	8560.9	8741.2	6299.5	57609.5 (+0.2%)
LIN-FIT	IS	35440.1	9785.1	8771.3	6575.2	60571.6 (+5.3%)
LIN-SHA	IS	35199.8	9850.5	8841.4	6507.1	60398.9 (+5.0%)
LOG-BIC	OOS	33715.4	8488.4	8890.8	6215.6	57310.2
LOG-1	OOS	33894.4	8479.0	8809.8	6258.6	57441.8 (+0.5%)
LIN-FIT	OOS	35323.4	9681.1	8844.3	6534.3	60383.0 (+4.8%)
LIN-SHA	OOS	35091.0	9751.9	8885.2	6503.1	60231.3 (+4.1%)

D.2 Out-of-sample Simulation Results

Fig. 9 shows the distribution of the total cost of out-of-sample simulations for runs with cut selection. Fig. 10-11 present additional results for the out-of-sample simulations.

D.3 Historical Simulation Results

Fig. 12 presents additional results for the simulations on historical data. It shows that on average, LOG-1 policies keep higher reservoir levels, and therefore less deficit costs are caused in times of shortage. The total deficit costs are also displayed in Table 6.

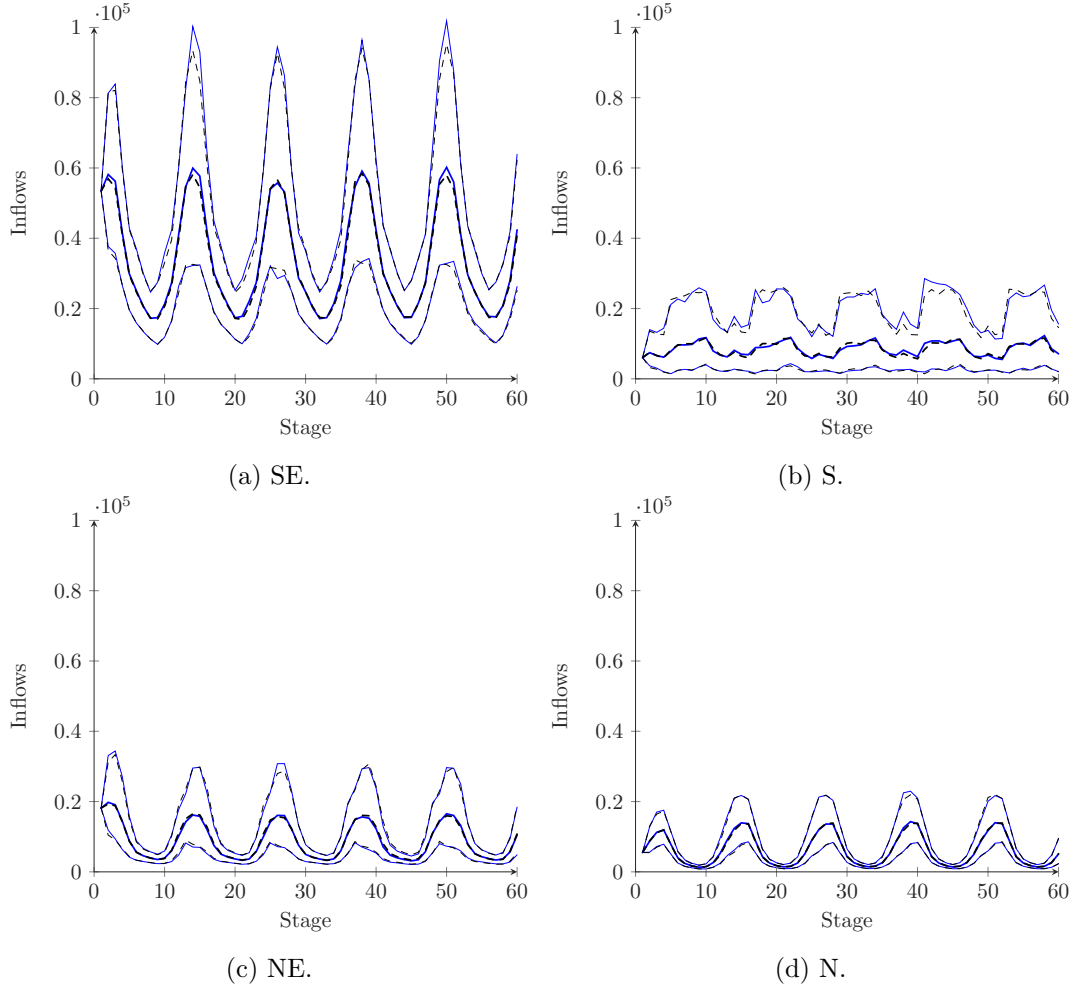


Figure 6: Inflows of the EERs for in-sample simulations (blue) and out-of-sample simulations (black, dashed) when using LOG-1 data. Each plot contains the mean, the 5%-quantile and the 95%-quantile of all sample paths. All values averaged over 5 runs.

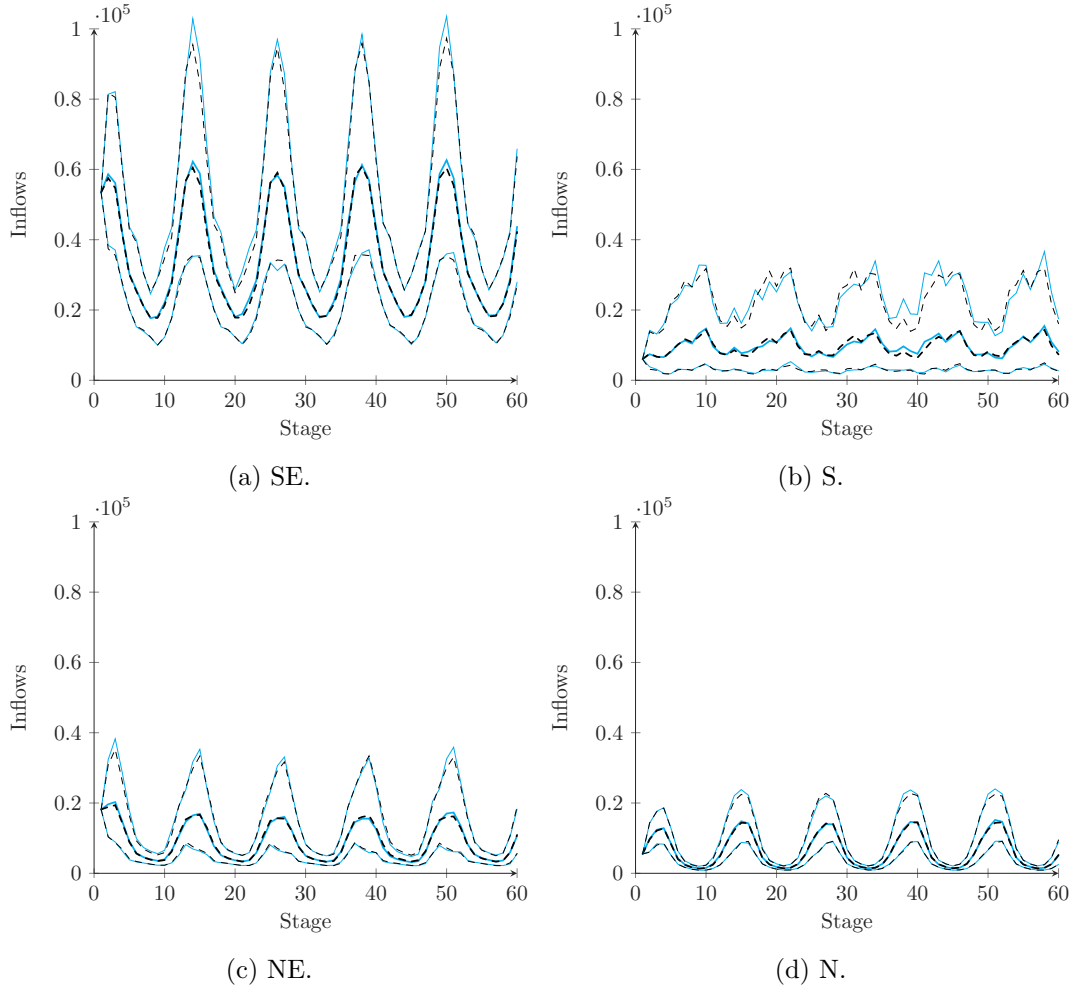


Figure 7: Inflows of the EERs for in-sample simulations (cyan) and out-of-sample simulations (black, dashed) when using LIN-SHA data. Each plot contains the mean, the 5%-quantile and the 95%-quantile of all sample paths. All values averaged over 5 runs.

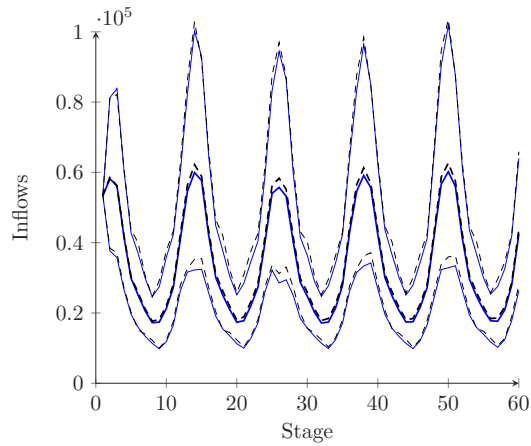


Figure 8: Inflows of the EERs for in-sample simulations using LOG-1 (blue) and LIN-SHA (black, dashed) data. Each plot contains the mean, the 5%-quantile and the 95%-quantile of all sample paths. All values averaged over 5 runs.

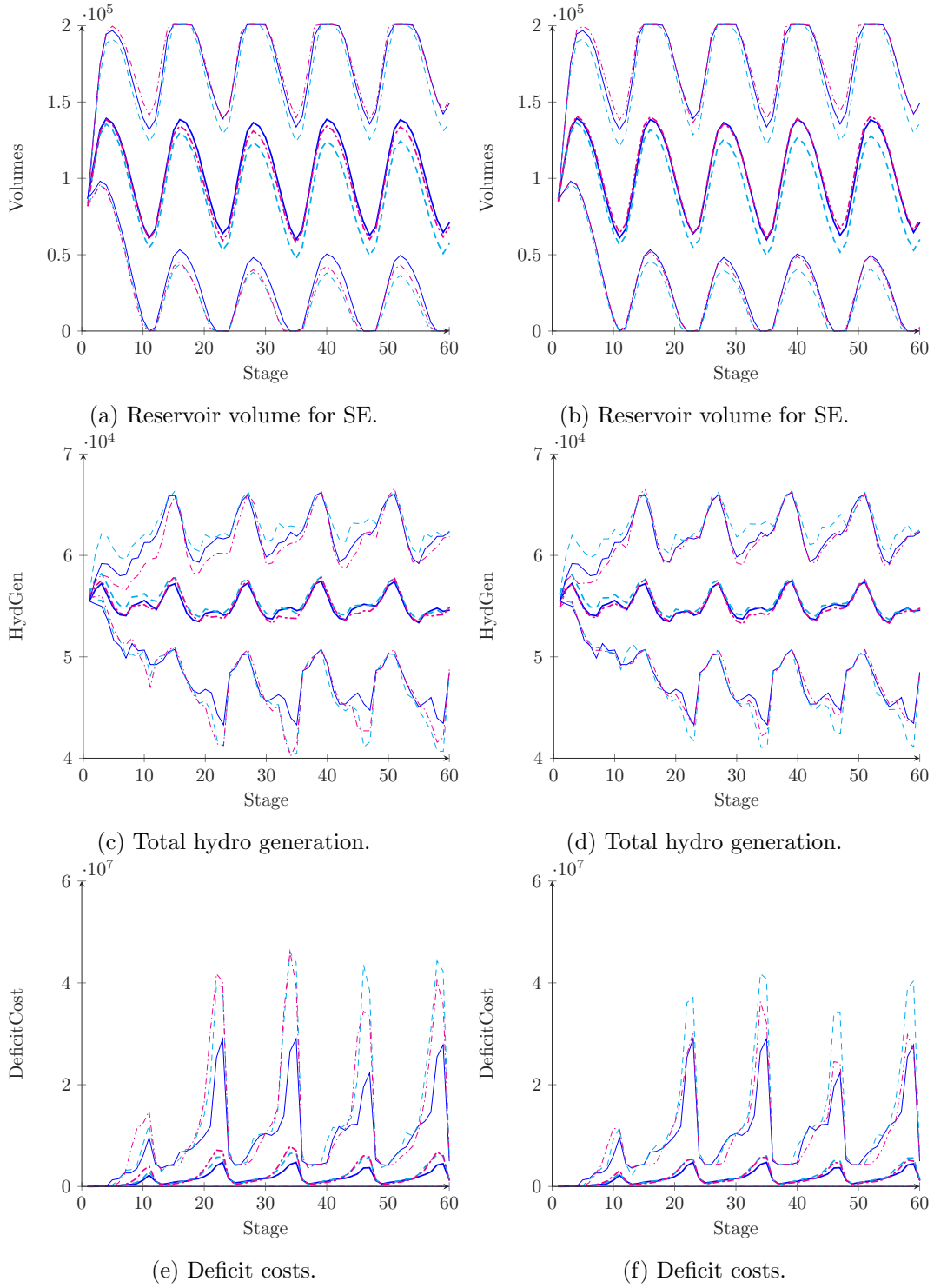


Figure 10: Plots of volumes of reservoir SE (up), total hydro generation for all 4 EERs (middle) and deficit costs (down) in out-of-sample simulation with LOG-1 data. LEFT: Runs without cut selection. RIGHT: Runs with cut selection for SDDP and MC-SDDP. Each plot contains the mean, the 5%-quantile and the 95%-quantile of all 2000 sample paths. Each plot contains results for policies `logSDDP LOG-1` (blue), `SDDP LIN-SHA` (cyan, dashed) and `MC-SDDP lattice` (magenta, dash dot). All values averaged over 5 runs.

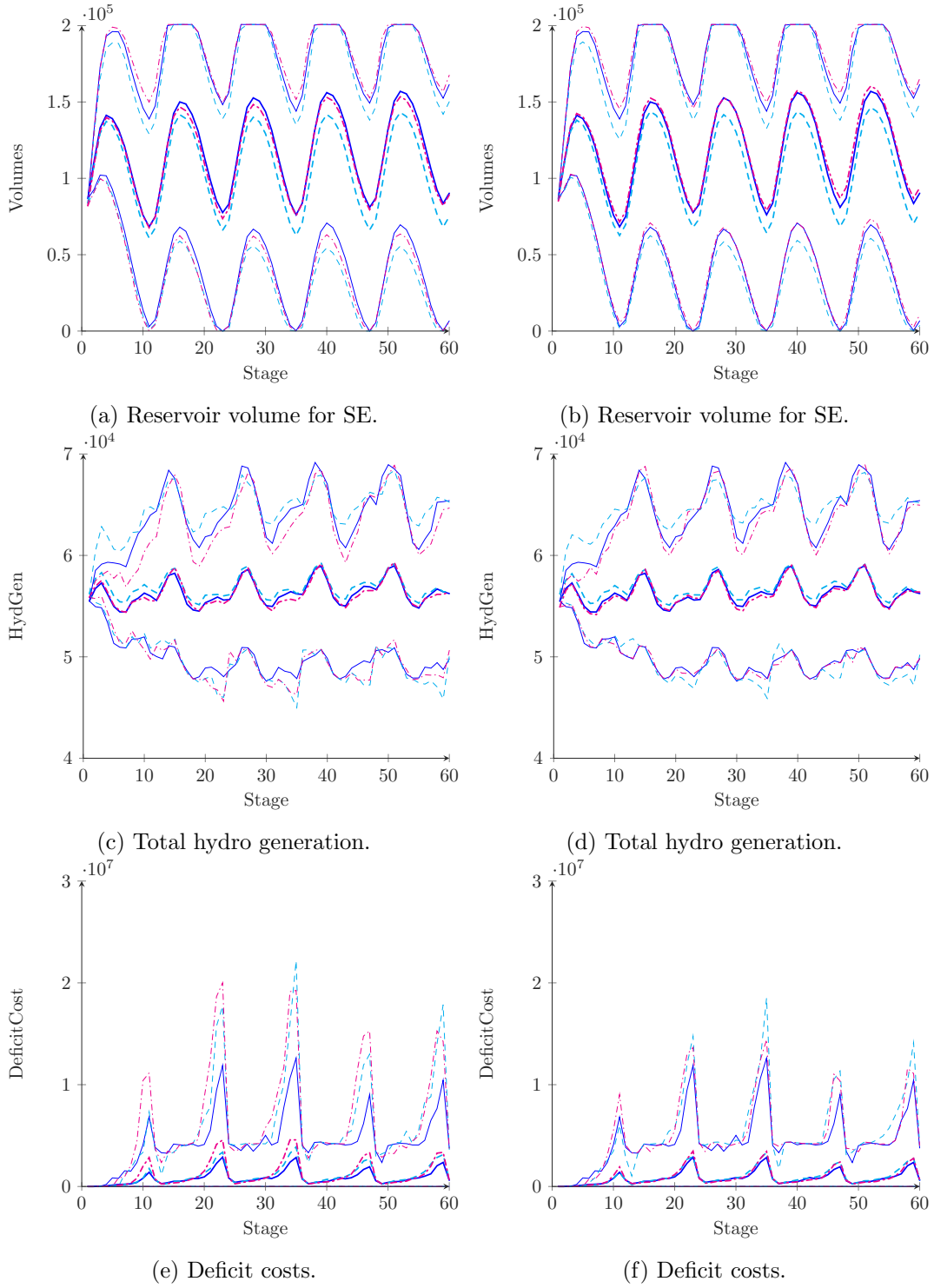


Figure 11: Plots of volumes of reservoir SE (up), total hydro generation for all 4 EERs (middle) and deficit costs (down) in out-of-sample simulation with LIN-SHA data. LEFT: Runs without cut selection. RIGHT: Runs with cut selection for SDDP and MC-SDDP. Each plot contains the mean, the 5%-quantile and the 95%-quantile of all 2000 sample paths. Each plot contains results for policies `logSDDP LOG-1` (blue), `SDDP LIN-SHA` (cyan, dashed) and `MC-SDDP lattice` (magenta, dash dot). All values averaged over 5 runs.

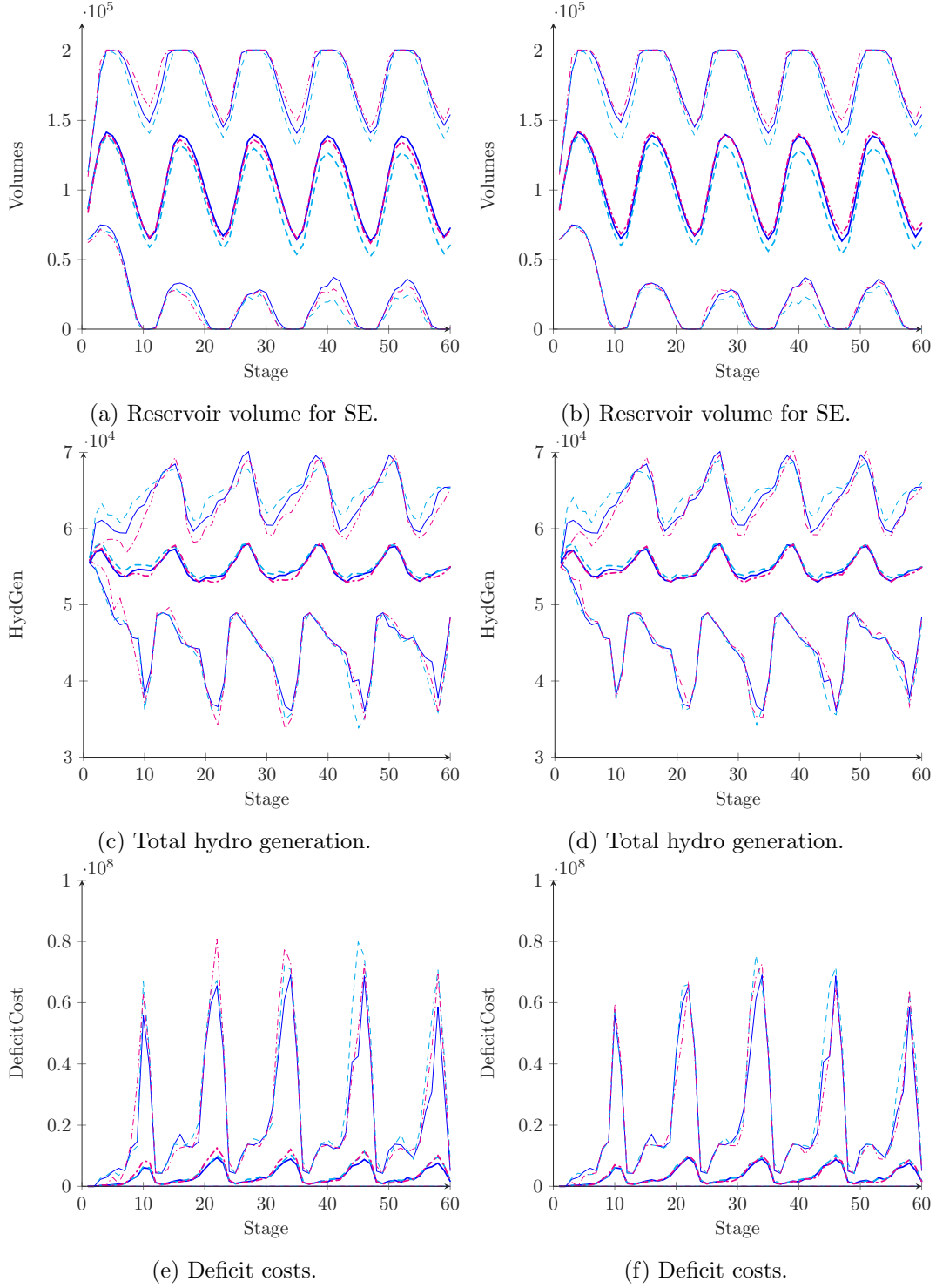


Figure 12: Plots of volumes of reservoir SE (up), total hydro generation for all 4 EERs (middle) and deficit costs (down) in simulation with historical data. LEFT: Runs without cut selection. RIGHT: Runs with cut selection for SDDP and MC-SDDP. Each plot contains the mean, the 5%-quantile and the 95%-quantile of all 70 historical paths. Each plot contains results for policies `logSDDP LOG-1` (blue), `SDDP LIN-SHA` (cyan, dashed) and `MC-SDDP lattice` (magenta, dash dot). All values averaged over 5 runs.

References

- [1] G. E. P. Box, G. M. Jenkins, G. C. Reinsel, and G. M. Ljung. *Time series analysis: forecasting and control*. Wiley, Hoboken, New Jersey, fifth edition, 2016.
- [2] L. Ding, S. Ahmed, and A. Shapiro. A Python package for multi-stage stochastic programming. Preprint, available online at http://www.optimization-online.org/DB_FILE/2019/05/7199.pdf, 2019.
- [3] C. Füllner and S. Rebennack. Stochastic dual dynamic programming and its variants – A review. Preprint, available at http://www.optimization-online.org/DB_FILE/2021/01/8217.pdf, 2023.
- [4] D. N. Gujarati and D. C. Porter. *Basic econometrics*. McGraw-Hill, New York, NY, fifth edition, 2009.
- [5] K. W. Hipel and A. I. McLeod. *Time series modelling of water resources and environmental systems*. 2006. available online at <https://www.eng.uwaterloo.ca/~kwhipel/Time%20Series%20Book.htm>.
- [6] T. Lohmann, A. S. Hering, and S. Rebennack. Spatio-temporal hydro forecasting of multireservoir inflows for hydro-thermal scheduling. *European Journal of Operational Research*, 255(1):243–258, 2016.
- [7] N. Löhdorf and A. Shapiro. Modeling time-dependent randomness in stochastic dual dynamic programming. *European Journal of Operational Research*, 273(2):650–661, 2019.
- [8] A.B. Philpott and Z. Guan. On the convergence of stochastic dual dynamic programming and related methods. *Operations Research Letters*, 36(4):450–455, 2008.
- [9] A. Shapiro and W. Tekaya. Report for technical cooperation between Georgia Institute of Technology and ONS – Operador Nacional do Sistema Elétrico – risk averse approach. Technical report, 2011.
- [10] A. Shapiro, W. Tekaya, J. P. da Costa, and M. P. Soares. Risk neutral and risk averse stochastic dual dynamic programming method. *European Journal of Operational Research*, 224(2):375–391, 2013.