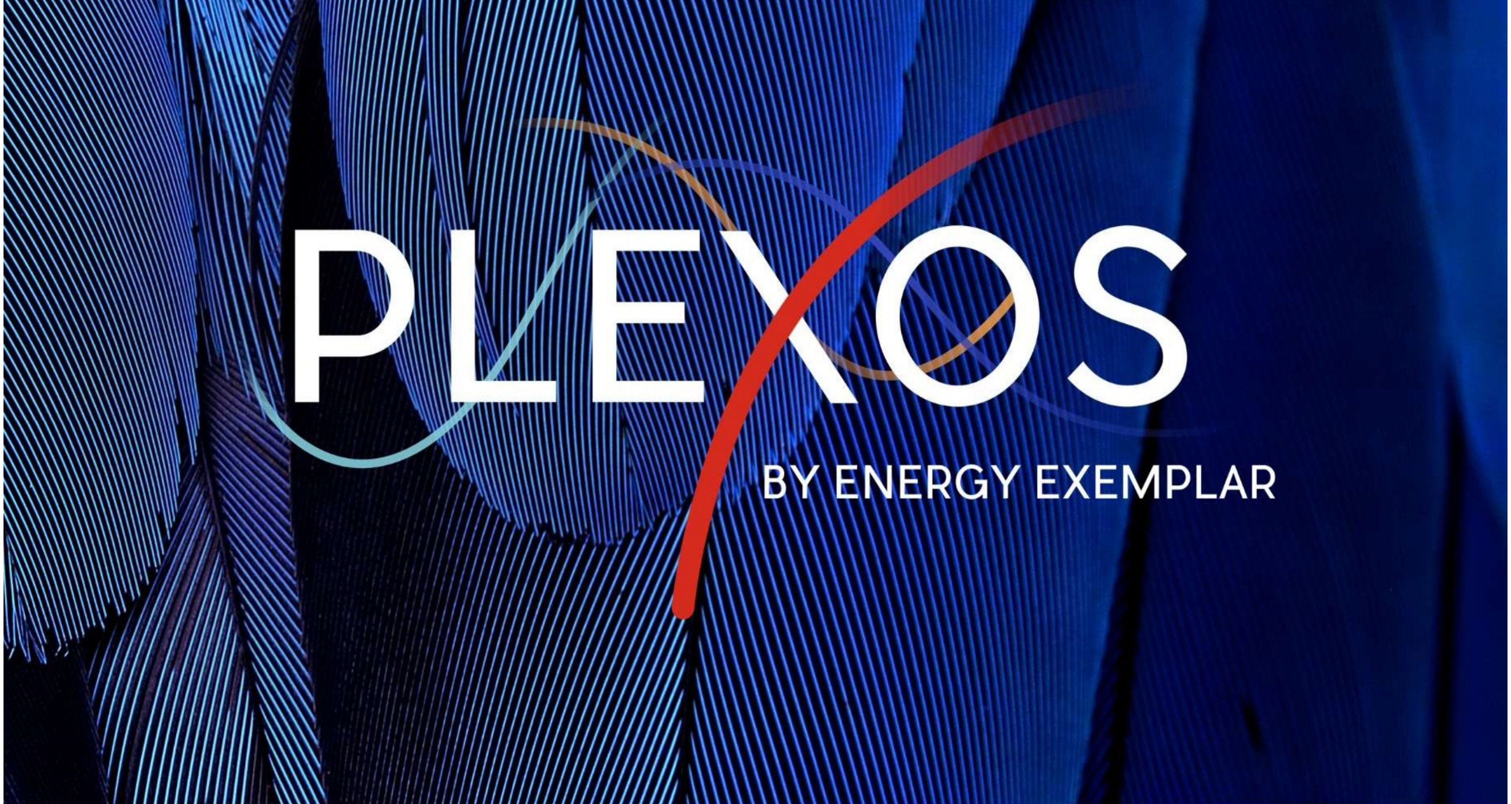


PLEXOS Stochastic Hydro Optimization

Felipe César Valdebenito Tepper, MSc
Belén Elaine Leveroni Villafana, MSc
Glenn Drayton, Ph.D.

Energy Exemplar Pty Ltd
17 Bagot Street, North Adelaide
SA 5006 Australia
+61 8 8361 7508

Principal Contact
Milorad Zecevic
Director Asia & Global Marketing
milorad.zecevic@energyexemplar.com

Executive Summary

In decision making under uncertainty, the decision maker has to make optimal decisions throughout a horizon with incomplete information. Over the considered decision horizon, a number of stages are defined and each stage represents a point in time where decisions are made or where uncertainty partially or totally vanishes. According to the number of stages considered in the optimization problem we can distinguish between two-stage and multi-stage stochastic problems. In a two-stage approach, the plan for the entire multi-period planning horizon is determined before the uncertainty is realized, and only a limited number of recourse actions can be taken afterwards. In contrast, a multi-stage approach allows revision of the planning decisions as more information regarding the uncertainties is revealed across the planning horizon.

Regarding stochastic optimization applied to hydro-thermal coordination, the two-stage approach only finds applications for short term operational planning. For medium and long-term studies, a multi-stage stochastic approach is more suitable because, in practice, the system operator or hydro plant companies are monitoring the state of the dam continuously and can take new decisions (stages) at any time given the actual and historical inflow information.

The disadvantage of multi-stage stochastic optimization is that the simulation time increases exponentially when more stages are added due to an increase in the dimensionality of the mathematical problem, and even when a few number of stages are added it can result in a problem impossible to solve without using reduction or decomposition techniques.

To help solve this dimensionality issue some algorithms have been proposed and this paper explores three of them:

1) Scenario Reduction
2) Stochastic Dual Dynamic Programming (SDDP)
3) Hanging Branches

Scenario reduction techniques only finds applications for a few number of stages and a reduced uncertainty, so it is not an option when the user wants to evaluate many stages and increased uncertainty.

SDDP (or DDP) is the method currently used in many countries where hydro-thermal coordination is paramount. It allows the user to evaluate many stages and uncertainty.

The Hanging Branches method was researched and developed at Energy Exemplar Adelaide's office during 2013 – 2015. The idea behind this method is to formulate the equivalent reduced SDDP multi-stage tree using scenario-wise decomposition techniques plus non-anticipativity constraints.

The comparison table between these three methods is showed below.

	Scenario reduction	SDDP	Hanging Branches
Documented in literature?	Yes	Yes	No. Method was developed at Energy Exemplar
Recursive vs Non-recursive*	Non-recursive	Recursive	Recursive
Multi-stage tree exploration	No resampling	Some SDDP versions resample the tree allowing multiple exploration paths	No resampling
Speed solution	An equivalent SDDP tree is impossible to solve using today's available computers	It uses decomposition techniques and parallel resolution of independent sub problems to improve simulation performance.	It should have equivalent speed to SDDP and much faster than scenario reduction.
Linear or Integer?	Linear or MIP	Linear Only	Linear or MIP
Implemented in PLEXOS?	Yes	No	Yes.

*Recursive: at the beginning of each stage, the decision maker has a perfect insight on the inflow scenario that will be observed at that stage
*Non - Recursive: inflow scenarios are revealed after the release policy is taken

Hanging Branches method is the method that will ultimately replace SDDP. Initial benchmarks between both methods, documented in this paper, using small multi-stage stochastic trees show that the objective function values are identical. Following the tests shown here more development work was undertaken throughout 2016-17 and the method's results benchmarked against full sized datasets confirming the accuracy of the results.

Additional developments supporting the Hanging Branches method have been undertaken related to automating the creation of the required hanging branches in the stochastic tree reading historical inflow information. This was necessary because, for large stochastic trees, it is unpractical to create a csv file manually with the required uncertainty. The developments undertaken were:

a) Historical apertures. Where the Hanging Branches are created randomly sampling the historical inflow information at each stage.

b) PARMA time series. A more powerful option than a) since the hanging branches are be created from past information so, for example, 'wet' observed scenarios are more probably to remain wet in the future. This is achieved using a periodic ARMA time series (PARMA).

Table of Contents

1 Introduction

Power systems having both hydro-electric and thermal generation require a systematic and coordinated approach to determine an optimal policy for dam operations. The goal of a hydro-thermal planning tool is to minimize the expected thermal costs across the simulation period. These types of problems generally require stochastic analysis to deal with inflow uncertainty. This can increase the mathematical size of the problem and can easily become cumbersome to solve.

This whitepaper reviews three different algorithms of stochastic programming to solve this problem and how these techniques can be applied to hydro-thermal scheduling problem and introduces the method developed at Energy Exemplar. In addition, a case study solved using SDDP and the proposed method is done and a comparison is provided.

2 The Challenge

The systematic coordination of a system composed of both hydro-electric and thermal plants requires determining an operational strategy that for each period of the planning horizon produces a scheduling plan of generation. This strategy minimizes the expected operational cost along the period, which is mainly composed of fuel costs plus penalties for failure in load supply. The problem becomes complex to solve because generally in hydro systems:

- Natural inflows are stochastic processes.
- Availability of water stored in dams is limited.
- There are complex cascading hydro systems.
- Existing water usage policies and environmental releases such as irrigation settlements.

Water as a fuel supply is cost-free, but its opportunity cost is fundamental to finding the optimal strategy for operations. This issue creates the need for a decision in each time period. Storage cannot be drained too low, which might incur generation shortfalls or excessive thermal output. On the other hand, we also want to avoid spillage of water and lost generation opportunities. Figure 1 summarizes the dilemma a hydro power planner faces to operate a dam.

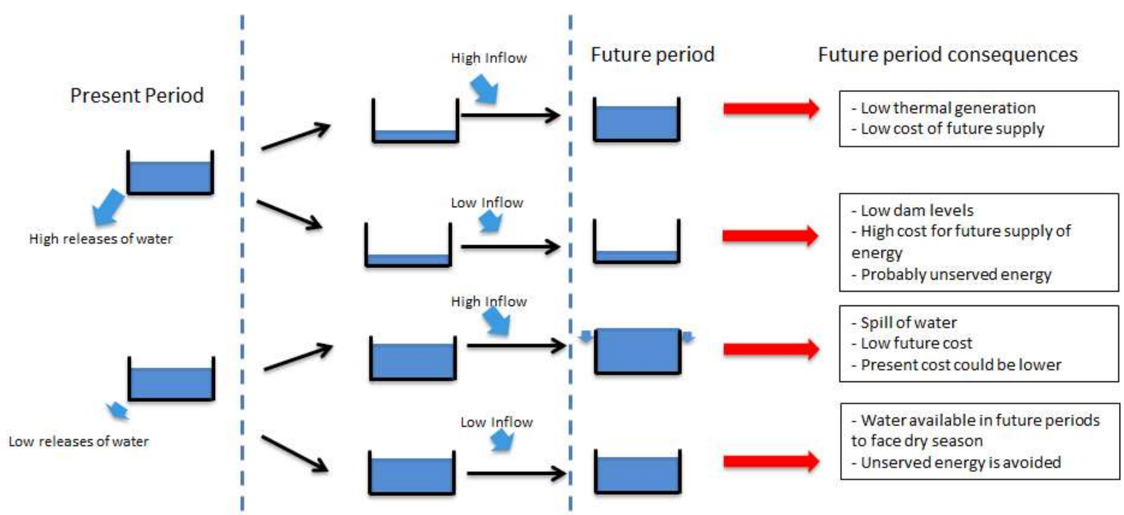


Figure 1: Diagram showing the dilemma hydro power planner faces under uncertainty.

3 The Formulation

A mathematical optimization tool can find a hydro releasing policy that minimizes the expected operational thermal cost by formulating an optimization problem such as the following:

$$Min\,\{Variable\ Costs\}$$

Subject to:

$$Energy\ Balance\ Equation$$
$$FlowLimits$$
$$GenerationLimits$$
$$Hydro\ Balance\colon\ v_{t+1} = v_t + Q_{\text{inflow}} - Q_{release}$$

Where the hydro balance equation shows the link between decisions in both the present and future.

Since water is free (no fuel cost) it is necessary to specify a final condition to minimise thermal costs along the simulation period, and to avoid the storage being completely drained. These final conditions can be represented as a target or a proxy for opportunity costs such as a deviation from targets, usually known as the future cost function or scrap value function.

4 Stochastic Optimization

In decision making under uncertainty, the decision maker must make optimal decisions throughout a decision horizon with incomplete information. Over the considered decision horizon, many stages are defined and each stage represents a point in time where decisions are made or where uncertainty partially or totally vanishes. The amount of information available to the decision maker is usually different from stage to stage. According to the number of stages considered in the optimization problem we can distinguish between:

- Two-stage stochastic problems.
- Multi-stage stochastic problems.

The stochastic problem is graphically represented as a scenario tree.

4.1 Scenario Tree Representation

A scenario tree consists of nodes and lines grouped in stages as showed in Figure 2. Each node in the scenario tree represents a possible state and it is a point in time where decisions are made.

The lines in the scenario tree are called "leaves" and represents the possible outcomes for the random variables so each of them has a probability associated. The sum of probabilities associated to each node is unity.

A path from the root node to any other node describes one realization of the stochastic process from the present time to the period that node appears. A path over the entire planning horizon is called a scenario.

Figure 2 shows a scenario tree consisting of 15 nodes, eight scenarios and three stages. This scenario tree represents a decision-making process where the decisions are re-evaluated four times (three stages plus the root node) along the planning horizon according to the revealed information at each stage.

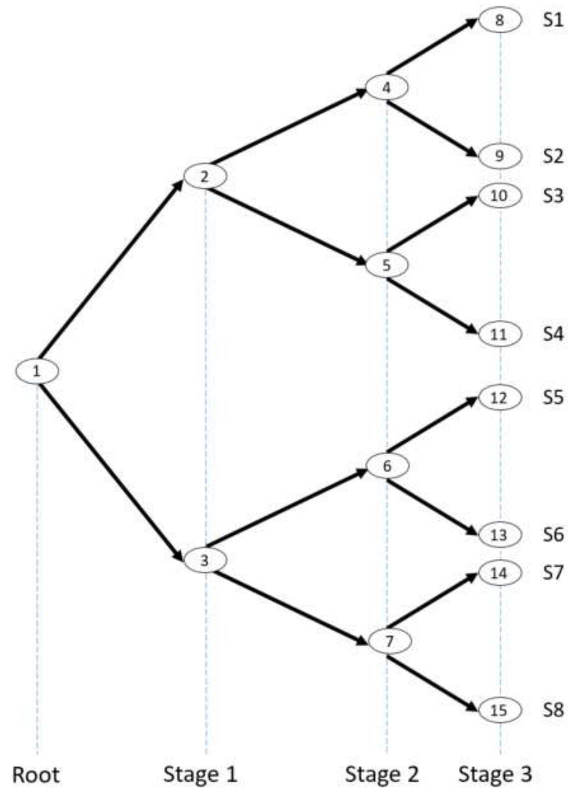


Figure 2: Scenario Tree representation

Two-stage Stochastic Problem

Two-stage applies to a decision-making problem where decisions are made in two stages and there exists uncertainty represented by a set of scenarios or samples. It can be assumed that two different decision variable vectors: $\underline{x}$ and $\underline{y}$ are involved in this problem. Decision $\underline{x}$ is made before knowing the actual value of the scenarios, while $\underline{y}$ is determined after knowing the actual value of the scenario.

Decision y depends on the decision $\underline{x}$ previously made. The decision-making process is as follows:

1. Decision $\underline{x}$ is made.
2. The uncertainty is revealed.
3. Decision $\underline{y}$ is made.

In this decision-making process, two kinds of decision are made:

1. First-stage or ‘here-and-now’ decisions (Decision $\underline{x}$). These decisions are made before the realization of the stochastic process. Hence, variables representing here-and-now decisions do not depend on each realization of the stochastic process.

2. Second-stage or ‘wait-and-see’ decisions (Decision $\underline{y}$). These decisions are made after knowing the actual realization of the stochastic process. Consequently, these decisions depend on each realization vector of the stochastic process. If the stochastic process is represented by a set of scenarios, a second stage decision variable is defined for each single scenario considered.

For all these decisions to be optimal, they need to be derived simultaneously by solving a single optimization problem, so that the relationships among the decision variables are properly accounted for.

Multi-stage Stochastic Problem

In some cases, decision-making problems have more than two stages, and the two-stage stochastic programming problem showed above is not appropriate to represent them. This fact motivates the use of multi-stage stochastic programming problems. The decision-making process for a multi-stage stochastic problem with r stages is the following:

1. Decisions $\underline{x}_1$ are made.
2. Partial uncertainty is revealed.
3. Decisions $\underline{x}_2$ are made.
4. Partial uncertainty is revealed
5. Decisions $\underline{x}_3$ are made.

. . .

2r-2. Uncertainty is revealed.

2r-1. Decision $\underline{x}_r$ are made.

This decision framework for a multi-stage problem is conveniently visualized using a scenario tree diagram like the tree in Figure 2.

4.2 Stochastic Problem Formulation: Two-stage vs Multi-stage

In a two-stage approach, the plan for the entire multi-period planning horizon is determined before the uncertainty is realized, and only a limited number of recourse actions can be taken afterwards. In contrast, a multi-stage approach allows revision of the planning decisions as more information regarding the uncertainties is revealed along the planning horizon. Consequently, the multi-stage model is a better characterization of the dynamic planning process, and provides more flexibility than does the two-stage model.

Figure 3 shows an expected comparison when the same optimization problem is solved using simple average deterministic, two-stage stochastic optimization, and multi-stage stochastic optimization. As can be observed, the expected cost is lower in a multi-stage problem when the number of stages increases because the decision maker has opportunity to re-evaluate his initial decision as additional information arrives.

The disadvantage of multi-stage stochastic optimization problems is that the dimensionality increases exponentially when more stages are added and even when a few number of stages are added it can be impossible to solve without using reduction or decomposition techniques because of large dimensionality resulting problems.

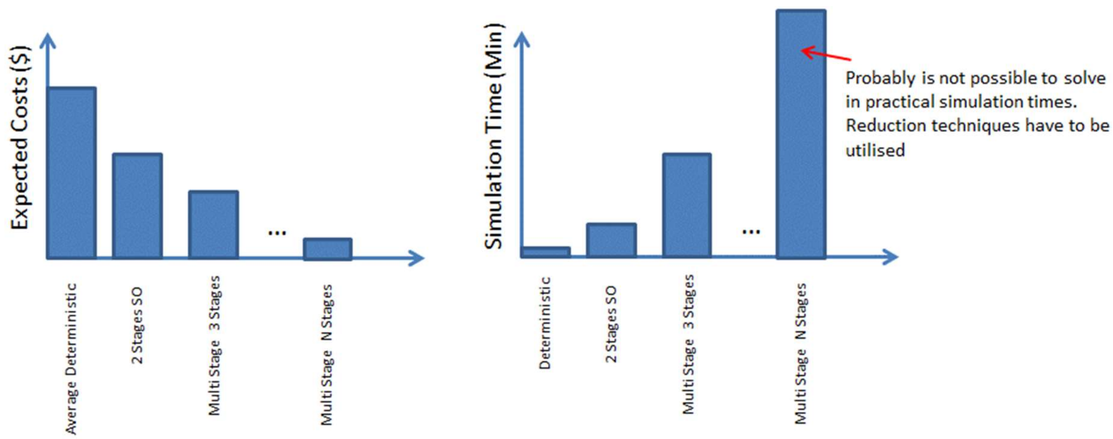


Figure 3: Comparison between expected costs and simulation times solving the problem in deterministic, 2 stages and multi-stage

Regarding stochastic optimization applied to hydro-thermal coordination, the two-stage approach only finds applications for short term operational planning. For medium and long-term studies, a multi-stage stochastic approach is more suitable because, in practice, the system operator or hydro plant companies are monitoring the state of the dam continuously and can take new decisions (stages) at any time given the actuals and historical inflow information.

4.3 Stochastic Problem Formulation: Node vs Scenario-wise Decomposition

A stochastic programming problem can be mathematically formulated using either a node-variable formulation or a scenario-variable formulation. The first formulation relies on variables associated with decision points while the second one relies on variables associated with scenarios.

Consider the example summarized in Figure 4 which shows a multi-stage stochastic problem where at each stage there are two possible outcomes for the inflow: Wet and Dry.

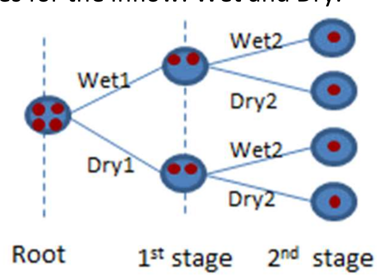


Figure 4: Scenario tree

Node formulation

The node-variable formulation of this problem is as follow:

$$P)\ Min\ \{C_1(R_{11}) + p_{21}[C_2(R_{21}) + p_{31}C_3(R_{31}) + p_{32}C_3(R_{32})] + p_{22}[C_2(R_{22}) + p_{33}C_3(R_{33}) + p_{31}C_3(R_{31})]\}$$

Subject to:

$V_{11} = V_{01} - R_{11} + I_{11}$
$V_{21} = V_{11} - R_{21} + I_{21}$
$V_{31} = V_{21} - R_{31} + I_{31}$
$V_{32} = V_{21} - R_{32} + I_{32}$
$V_{22} = V_{11} - R_{22} + I_{22}$

$V_{33} = V_{22} - R_{33} + I_{33}$
$V_{34} = V_{22} - R_{34} + I_{34}$
Energy Balance
Storage Capacity
Max Capacity

Where:
$V_{i,k}$: End volume at stage i, scenario k
$R_{i,k}$: Storage release at stage i, scenario k
$I_{i,k}$: Inflow at stage i, scenario k
$p_{i,k}$: Probability of scenario k at stage i.
$C(R_{i,k})$: Thermal cost that results from release policy $R_{i,k}$

Scenario-wise Decomposition (SWD) or scenario-variable formulation

The Scenario-wise Decomposition formulation of this problem is as follow:

$$P)\ Min\ \{p_{11}\times p_{21}[C_1(R_{11})+C_2(R_{21})+C_3(R_{31})]+p_{11}\times p_{22}[C_1(R_{12})+C_2(R_{22})+C_3(R_{32})]$$
$$+p_{12}\times p_{21}[C_1(R_{13})+C_2(R_{23})+C_3(R_{33})]+p_{12}\times p_{22}[C_1(R_{14})+C_2(R_{24})+C_3(R_{34})]\}$$

Subject to:
$V_{11} = V_{01} - R_{11} + I_{11}$
$V_{21} = V_{11} - R_{21} + I_{21}$
$V_{31} = V_{21} - R_{31} + I_{31}$
$V_{12} = V_{02} - R_{12} + I_{12}$
$V_{22} = V_{12} - R_{22} + I_{22}$
$V_{32} = V_{22} - R_{32} + I_{33}$
$V_{13} = V_{03} - R_{13} + I_{14}$
$V_{23} = V_{13} - R_{23} + I_{24}$
$V_{33} = V_{23} - R_{33} + I_{34}$
$V_{14} = V_{04} - R_{14} + I_{14}$
$V_{24} = V_{14} - R_{24} + I_{24}$
$V_{34} = V_{24} - R_{34} + I_{34}$
Energy Balance
Storage Capacity
Max Capacity
$V_{01}=V_{02}=V_{03}=V_{04}$
$V_{11}=V_{12}=V_{13}=V_{14}$
$V_{21}=V_{22}$
$V_{23}=V_{24}$
$V_{31}, V_{32}, V_{33}, V_{34}$ free

Where:
$V_{i,k}$: End volume at stage i, scenario k
$R_{i,k}$: Storage release at stage i, scenario k
$I_{i,k}$: Inflow at stage i, scenario k
$p_{i,k}$: Probability of scenario k at stage i.
$C(R_{i,k})$: Thermal cost that results from release policy $R_{i,k}$

Scenario-wise decomposition requires a larger number of variables and constraints including those marked in blue above called "non-anticipativity". These conditions guarantee that decisions cannot be dependent on the scenario realization. They are logical constraints related to availability of information at any decision point in time. Red circles in Figure 4 shows non-anticipativity variables that need to be enforced at each node.

4.4 Recourse vs Non-recursive Multi-stage Stochastic Problem

The decision model should be designed to allow the user to adopt a decision policy that can respond to events as they unfold. To formulate a multi-stage problem with dynamic stochastic data during time, emphasis has to be placed on the decision to be made today, given present resources, future uncertainties and possible recourse actions in the future.

Depending on the availability of information on the uncertain parameters at the beginning of each stage in the scenario tree, different recourse actions are defined for them.

It is possible to identify two types of decision depending on the availability of the information at the beginning of each stage:

1. Recursive: At the beginning of each stage, the decision maker has a perfect insight on the inflow scenario that will be observed at that stage. Thus, the decisions can be adjusted for different inflow scenarios.

2. Non-recourse: If inflow scenario values are revealed after the release policy is taken.

Recursive multi-stage

The recursive multi-stage can be represented in the following scenario tree Figure:

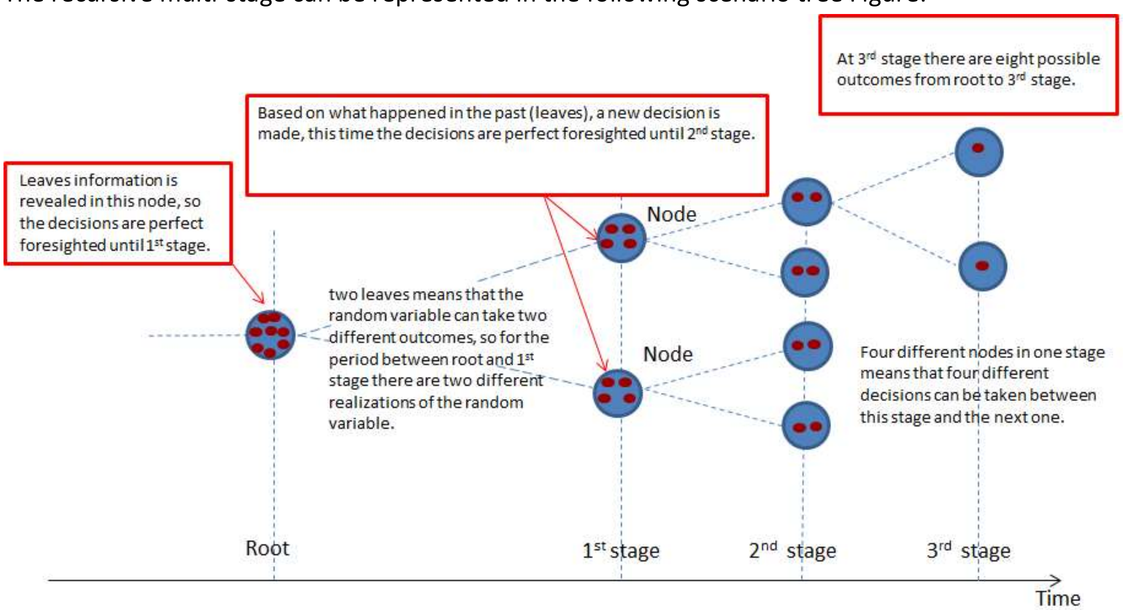


Figure 5: Scenario tree for recursive multi-stage

Non - Recursive multi-stage

The non - recursive multi-stage can be represented in the following scenario tree Figure:

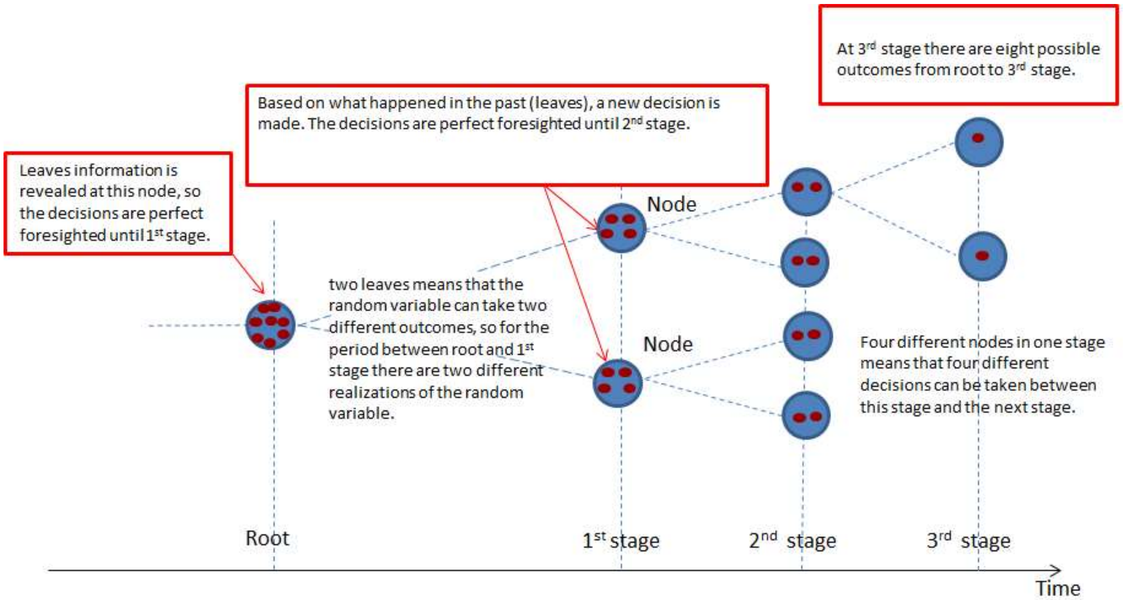
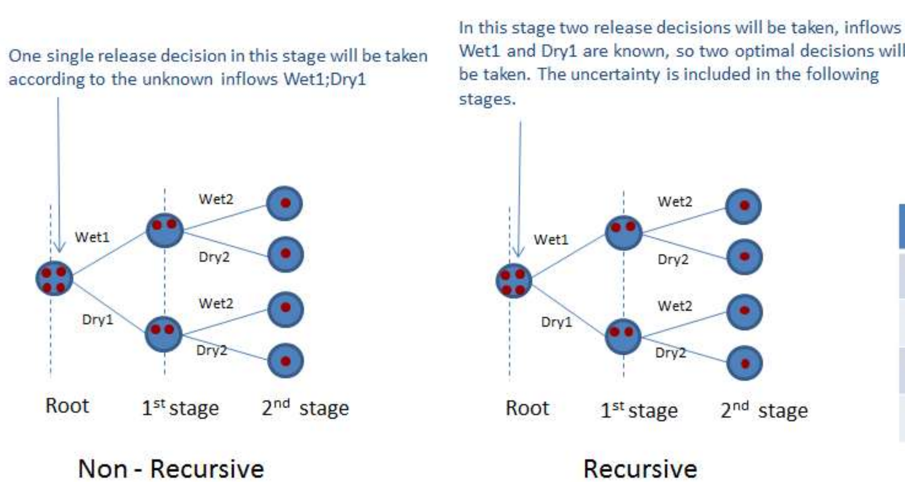


Figure 6: Scenario tree for non-recursive multi-stage

The scenario tree representation doesn't say if the multi-stage problem is recursive or non-recursive so in addition to the diagram it is necessary to specify what type of recourse actions are available.

The following figure compares both approaches applied to stochastic multi-stage hydro problems where end volumes have to be decided given uncertainty in future inflows.



4 Scenarios:

1: Wet, Wet
2: Wet, Dry
3: Dry, Wet
4: Dry, Dry

Condition	Probability
Wet1	P11
Dry1	P12
Wet2	P21
Dry2	P22

Figure 7: Recursive vs non-recursive multi-stage stochastic problems

4.5 Stochastic multi-stage hydro dimensionality issue

When using multi-stage stochastic optimization, many possible scenarios can be generated as shown in Figure 8. For such a high number of scenarios, it is impossible to numerically obtain a solution for the multi-stage optimization problem. Different techniques have been introduced in the literature to help solve this problem and commonly involve scenario tree reduction or a more simplified tree solved with decomposition techniques.

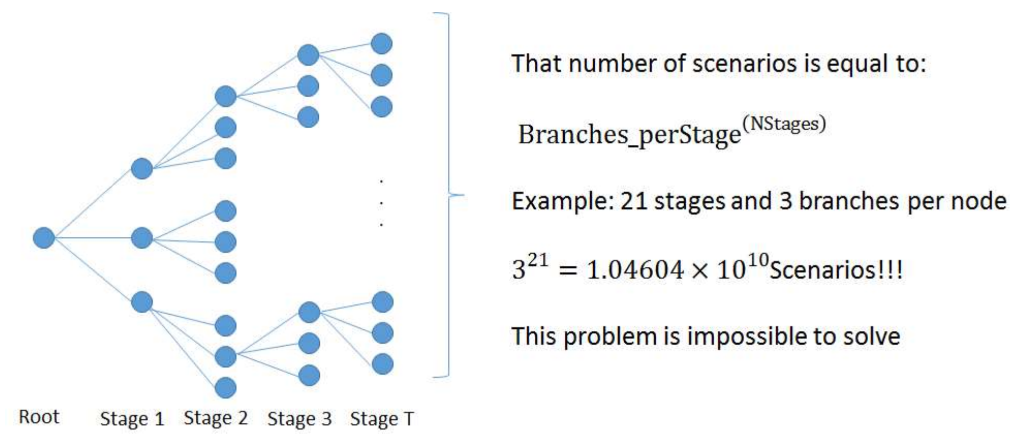


Figure 8: Multi Stage dimensionality issue

5 Algorithms to solve stochastic hydro-thermal coordination

5.1 Multi-stage tree reduction

PLEXOS implements scenario reduction techniques as an option for solving these problems. These techniques use strategies to reduce the number of scenarios in the optimization problem using algorithms for constructing a multi-stage scenario tree out of a given set of scenarios.

Since generating a very small number of scenarios by Monte Carlo simulation is not desired because less scenarios give less information, the objective is to lose minimum information by the reduction process applied to the complete set of scenarios.

The disadvantage of this technique is that it is necessary to reduce the tree to a very small tree to make it mathematically solvable by current solvers. In most real cases, the resultant tree doesn't represent the uncertainty well.

5.2 Stochastic Dual Dynamic Programming

Stochastic Dual Dynamic Programming (also called simple "Dual Dynamic Programming") is a method developed in the 1970s （Read，1979）where the hydro temporal coupling decisions are broken and replaced by the concept called the Future Cost Function.

The hydro problem at each stage becomes:

Objective function	$Z(v_{t-1}^{*}) = \min(CG + \alpha_t)$	(1)
Hydro Constraints	$v_t = v_{t-1} + I - R \rightarrow \pi$	(2)

$$v_{min} \leq v \leq v_{max} \quad (3)$$

$$Gh = \rho \cdot q \quad (4)$$

Generation Constraints $$Gmin \leq G \leq Gmax \quad (5)$$

Load supply $$\sum G = D \quad (6)$$

Where:

α: Future Cost Function (FCF)

v_t: stored volume at the end of stage t

v_{t-1}: initial volume at stage t, or stored volume at the end of stage t-1

R: Release or outflow volumes (turbined and spilled)

I: Inflow volumes (lateral inflow plus releases from upstream plants)

v_{min}: Minimum storage (if required)

v_{max}: Maximum storage

G: Energy production

Gmax: Maximum generation capacity

An iterative approach is needed to create the Future Cost Function with cuts that approximate the real future cost function with a piecewise linear function (see Figure 10) that samples the storage at "interesting" states. One cut is created at each iteration and the method stops when a convergence criterion is met.

Schematic representation of SDDP

The multi-period stochastic hydro problem can be decomposed in multiple steps where each step can be represented by the sum of:

a) Actual cost: Corresponds to the thermal variable generation costs in that step.

b) Future cost: Corresponds to the future thermal variable generation costs associated to the future steps.

At each step, the corresponding actual costs decrease if more water is used but future costs increase so there is an optimal point where the release decision minimizes the sum of actual and future costs.

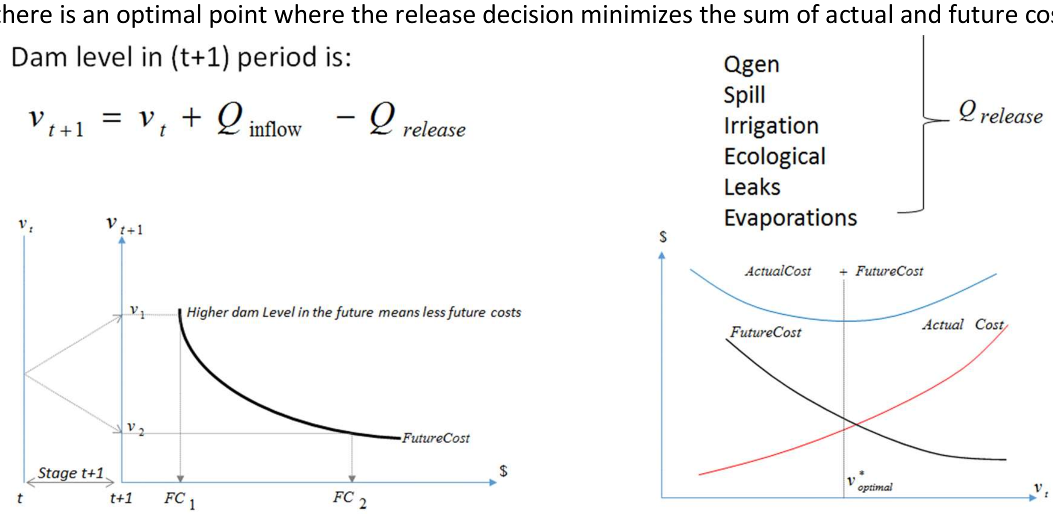


Figure 9: Actual and future costs schematic representation

Each step is modelled as a linear programming (LP) problem and the iterative procedure has two passes: forward and backward. One cut for the future cost function at that particular step is created in each backward pass.

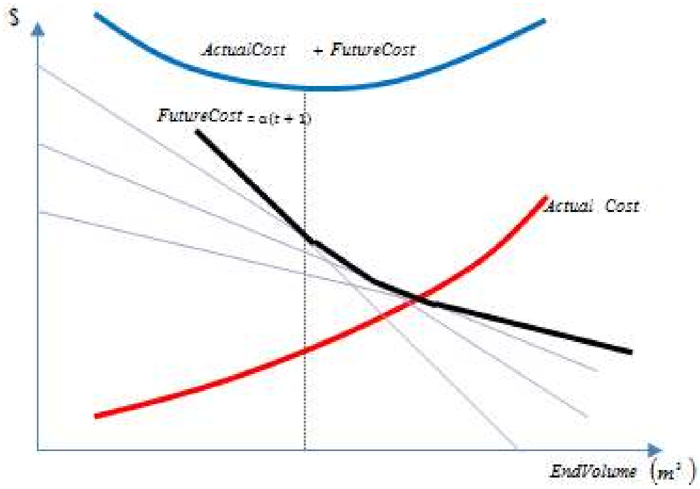


Figure 10: Future cost function approximation

The algorithm to build the approximations to the Future Cost Function is summarized in the following section.

SDDP Algorithm

Forward pass

Define a set of inflow scenarios $I_t = \{I_t^1, \dots, I_t^m, \dots, I_t^M\}$ for all stages t = 1, …, T

For each inflow scenario $I_t = I_t^1, \dots, I_t^m, \dots, I_t^M$

Initialize storage value for stage 1 as $v_t^m = v_1$

For t=1, …, T

Solve the one-stage scheduling problem for initial storage v_t^m and inflow I_t^m:

$$\min \mathrm{c_t}(\mathrm{u_t^m}) + \alpha_{t+1} \tag{7}$$

Subject to

$$v_{t+1}^m = v_t^m - u_t^m - s_t + I_t^m$$
$$v_{t+1}^m \le v_{max}$$
$$u_t^m \le u_{max}$$
$$\alpha_{t+1} \ge \varphi_{t+1}^n v_{t+1} + \delta_{t+1}^n \quad n = 1, \dots, N$$

Next

Next

Backward pass

Set number of linear segments N=number of initial storage values M.

Initialize future cost function for the last stage as zero: $\{\varphi_{T+1}^n and\ \delta_{T+1}^n\}$ for n=1, …, N

For t = T, T-1, …, 1

For each storage value $v_t = \{v_t^m, m = 1, \dots, M\}$

For each inflow scenario $I_t = I_t^1, \dots, I_t^k, \dots, I_t^K$

Solve the one-stage scheduling problem for initial storage $v_t^m$and inflow I_t^k.

$$\mathrm{\alpha_t^k(v_t^m) = Min\ C_t(G_t)} + \alpha_{t+1} \tag{8}$$

Subject to

$$v_{t+1} = v_t^m - u_t - s_t + I_t^k \;\rightarrow \pi_t^k$$
$$v_{t+1} \le v_{max}$$
$$u_t \le u_{max}$$
$$\alpha_{t+1} \ge \varphi_{t+1}^n v_{t+1} + \delta_{t+1}^n \quad n = 1, \ldots, N$$

Next

Calculate the coefficient and constant term for the mth linear segment of the future cost function in the previous stage:

$\varphi_t^m = \sum_{k=1}^{K} p_k \times \pi_{ht}^k$ and $\delta_t^m = \sum_{k=1}^{K} p_k \times \alpha_t^k (v_t^m) - \varphi_t^m \times v_t^m$

Next

Next

Where:

m: set of plants immediately upstream

u_t: turbined outflow volume during stage t

s_t: spilled outflow during stage t

Lower bound calculation

The lower bound is calculated as:

$$\mathrm{Z_{lower}} = \alpha_1 \mathrm{v}_1 \tag{9}$$

Upper bound calculation

The upper bound is calculated as the sum of all immediate costs along the study period.

$$z^m = \sum_{t=1}^{T} c_t(u_t^m)$$

Optimality Check

Optimality check is achieved when the lower bound (See equation (9)) is inside the following confidence interval.

Equation 10 shows the expected operation cost which is estimated as the mean total cost over all simulation scenarios.

$$\hat{\mathrm{z}} = \frac{1}{\mathrm{M}} \sum_{\mathrm{t}=1}^{\mathrm{T}} \mathrm{z}^{\mathrm{m}} \tag{10}$$

Equation 11 is generally used for a 95% confidence interval.

$$\bar{\mathrm{z}} \in [\hat{\mathrm{z}} - 1.96\hat{\sigma}; \hat{\mathrm{z}} + 1.96\hat{\sigma}] \tag{11}$$

Where $\hat{\sigma}$ is obtained by Equation 12 which is the standard deviation of the estimator.

$$\hat{\sigma} = \left[\frac{1}{\mathrm{M}-1} \sum_{\mathrm{t}=1}^{\mathrm{T}} (\mathrm{z}^{\mathrm{m}} - \bar{\mathrm{z}})^2\right]^{1/2} \tag{12}$$

New iteration

If the lower bound is outside the confidence interval, the backward recursion is executed again with an additional set of storage values. The natural candidates for the new values are the volumes $\{v_t^m = 1, \dots, M\}$ produced in 0.

Solving the full multi-stage stochastic graphically tree using SDDP

The SDDP algorithm can be explained graphically using the following multi-stage tree:

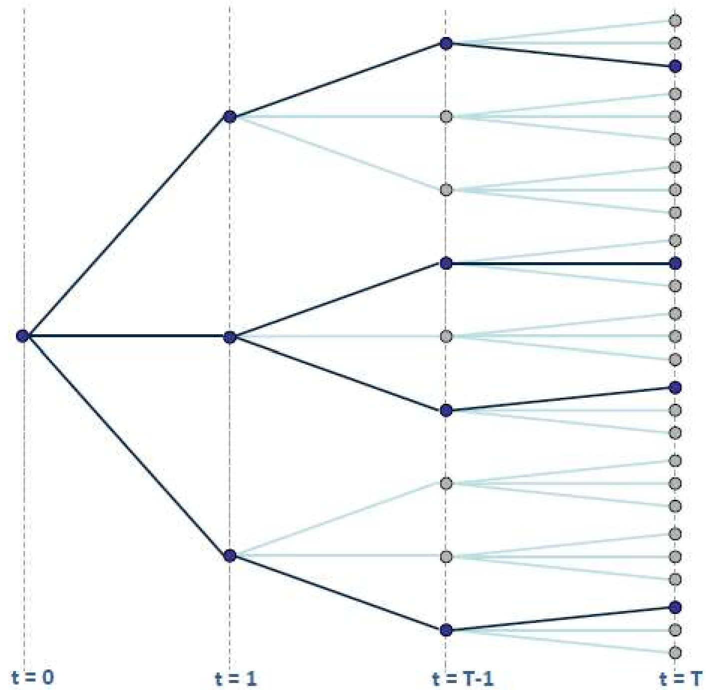


Figure 11: Full multi-stage tree to be solved using SDDP algorithm

Blue paths are the forward simulation paths and light blue are the paths representing uncertainty, these light blue paths are used in the backward pass.

Forward Pass

The forward pass can be summarized in the following figures, where the problem is decomposed in steps with a duration coincident with stage duration and the link between stages is represented using a Future Cost Function (FCF).

The first step mathematical problem can be represented using the following diagram:

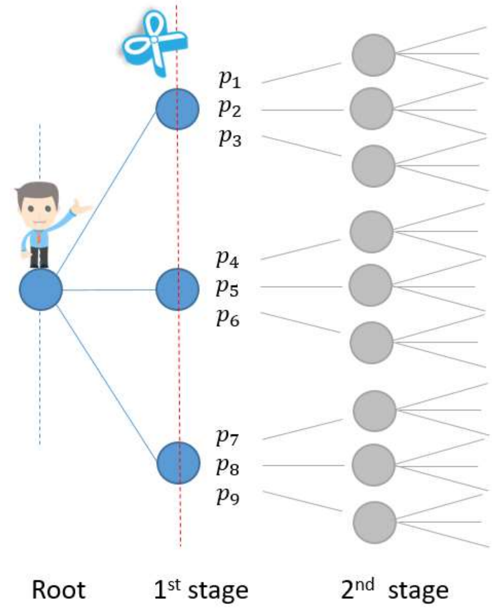


Figure 12: First step SDDP forward simulation, sub problem 1

The second step mathematical problems can be represented using the following diagrams:

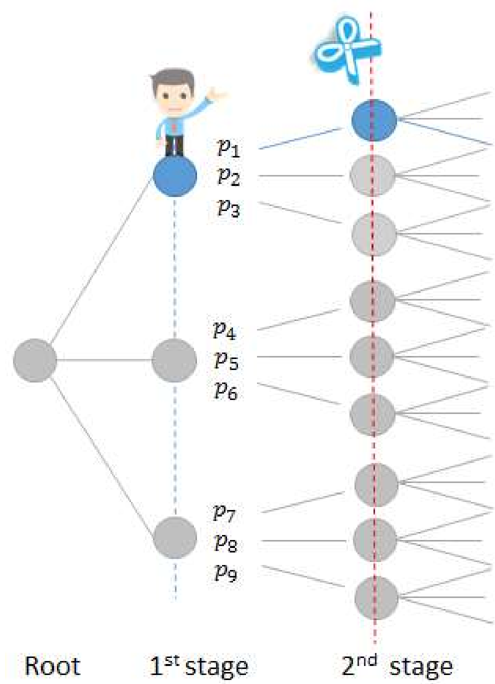


Figure 13: Second step SDDP forward simulation, sub problem 2

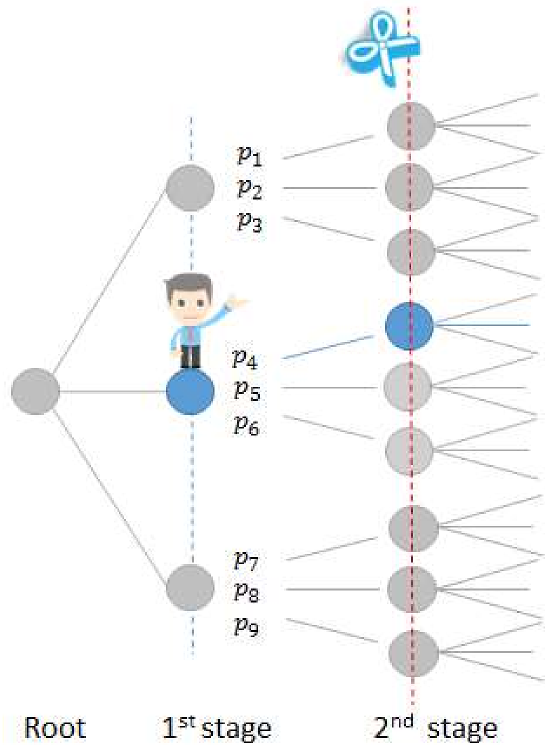


Figure 14: Second step SDDP forward simulation, sub problem 3

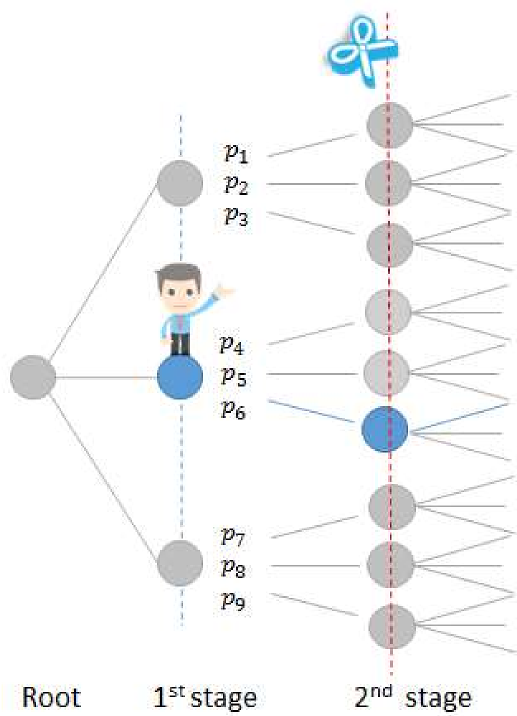


Figure 15: Second step SDDP forward simulation, sub problem 4

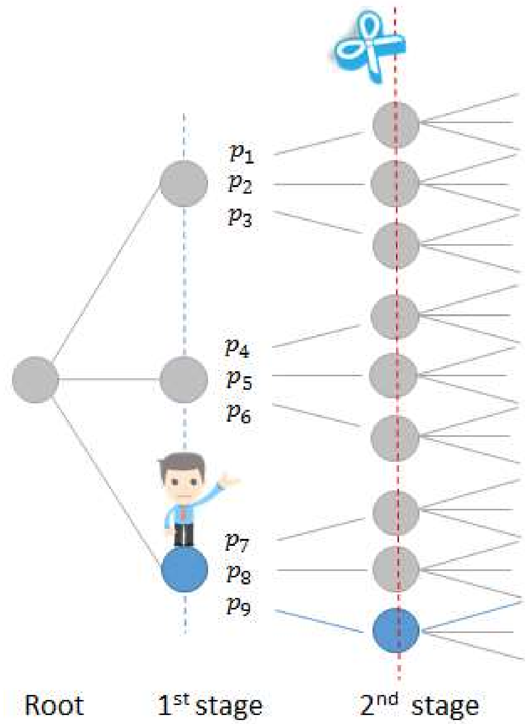


Figure 16: Second step SDDP forward simulation, sub problem 5

The sub-problems are independent and sub problems in same step can be run in parallel.

The third step is similar.

Backward Pass

The backward pass at each stage can be summarized in the following figures, where the yellow branches are the independent problems solved at each stage. Each backward stage produces a new cut to calculate the FCF.

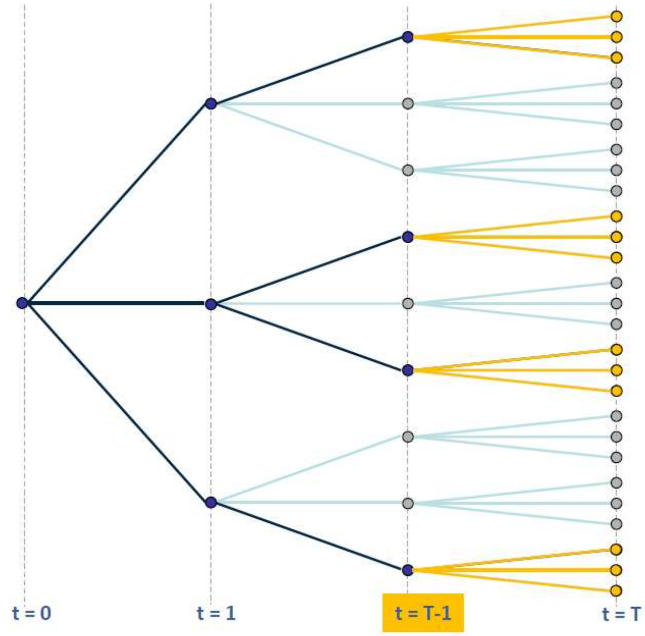


Figure 17: SDDP backward pass and FCF approximation at t=T-1

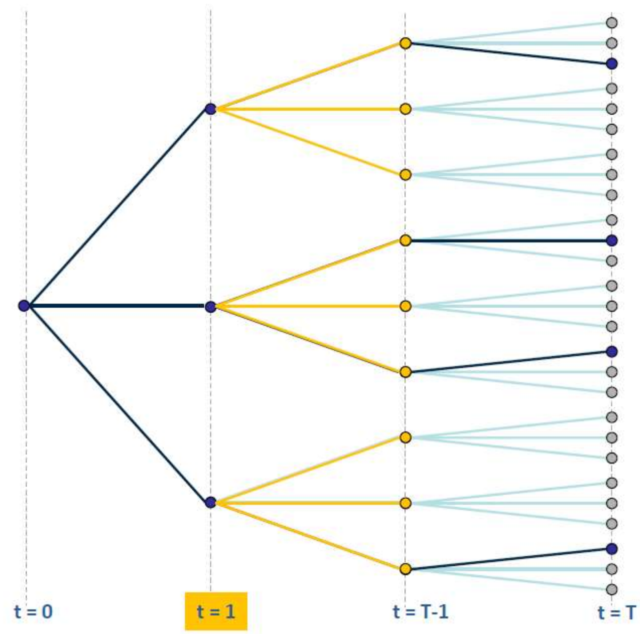


Figure 18: SDDP backward pass and FCF approximation at t=1

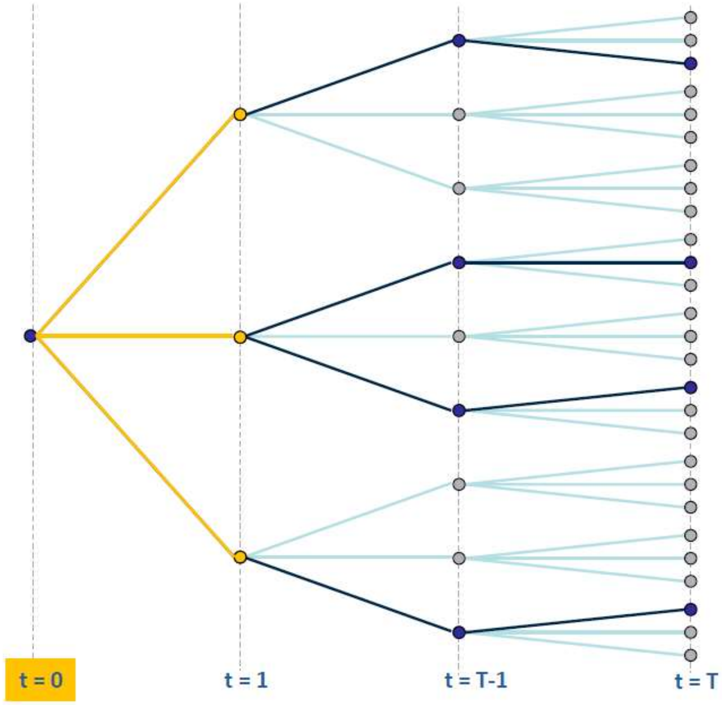


Figure 19: SDDP backward pass at t=0

It can be observed that all nodes of the forward and backward pass are solved independently.

Simplified tree solved using SDDP

SDDP algorithm can be formulated to solve the full multi-stage stochastic tree but it has the same dimensionality issue described in Figure 8, where the number of sub-problems are equal to: $Leaves^{NumberStages}$.

The simplified SDDP tree reduces the size of the problem and then become mathematically solvable. This reduced tree has the following number of subproblems: $Leaves \times NumberStages$. The following figure summarizes this simplified tree where red paths are the paths explored during forward simulation and grey branches represent uncertainty at each stage.

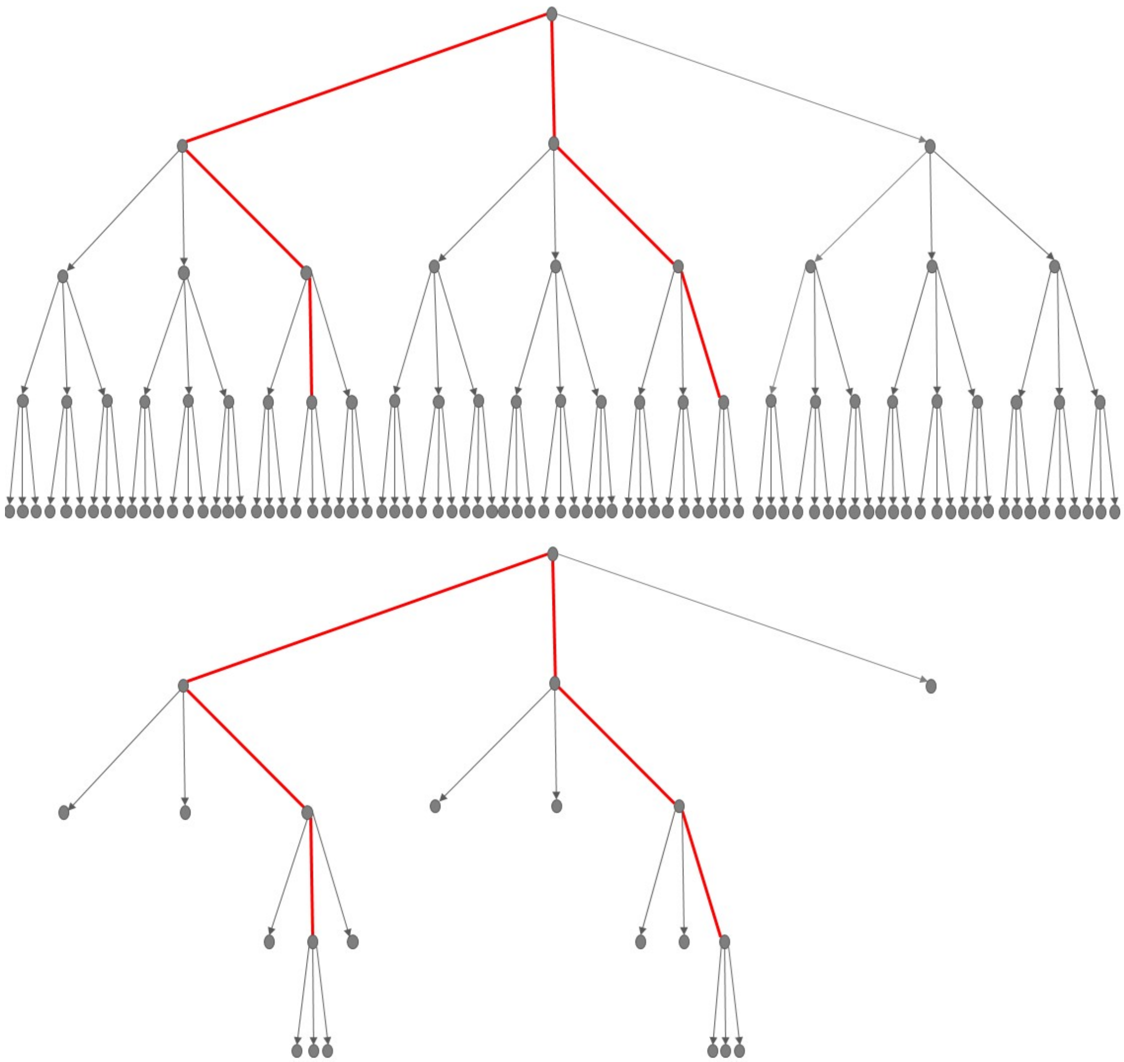


Figure 20: Full tree vs SDDP tree

5.3 Hanging branches

This method was researched and developed at Energy Exemplar's Adelaide office during 2013 -2015. The problem is formulated using recursive scenario-wise decomposition formulation and the SDDP stochastic tree.

The full tree can be classified in full branches, hanging branches and death branches as showed below:

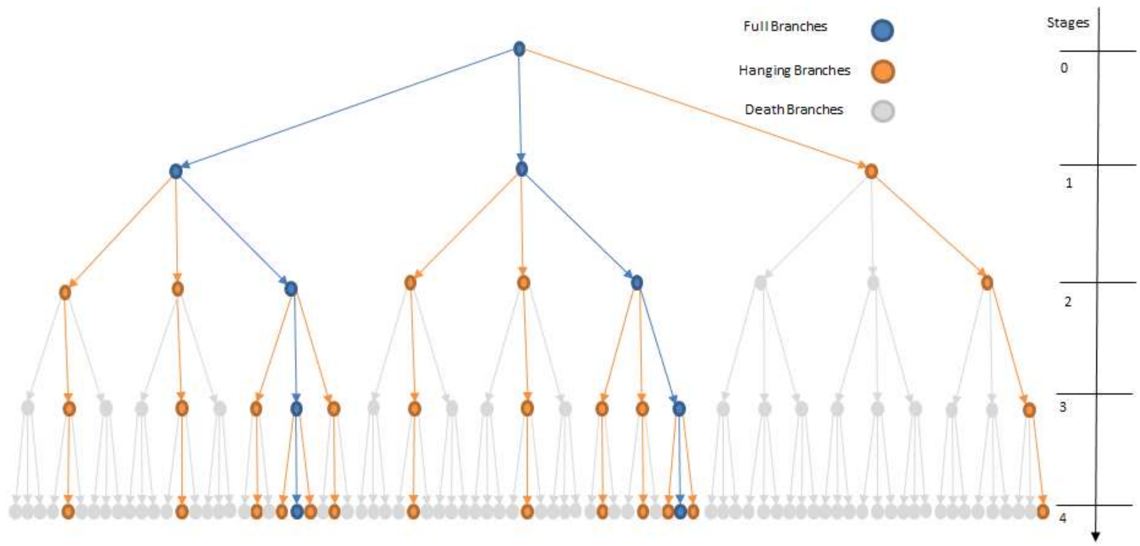


Figure 21: Full multi-stage tree illustrating full, hanging and death branches

The resulting tree to be formulated in an optimization problem is:

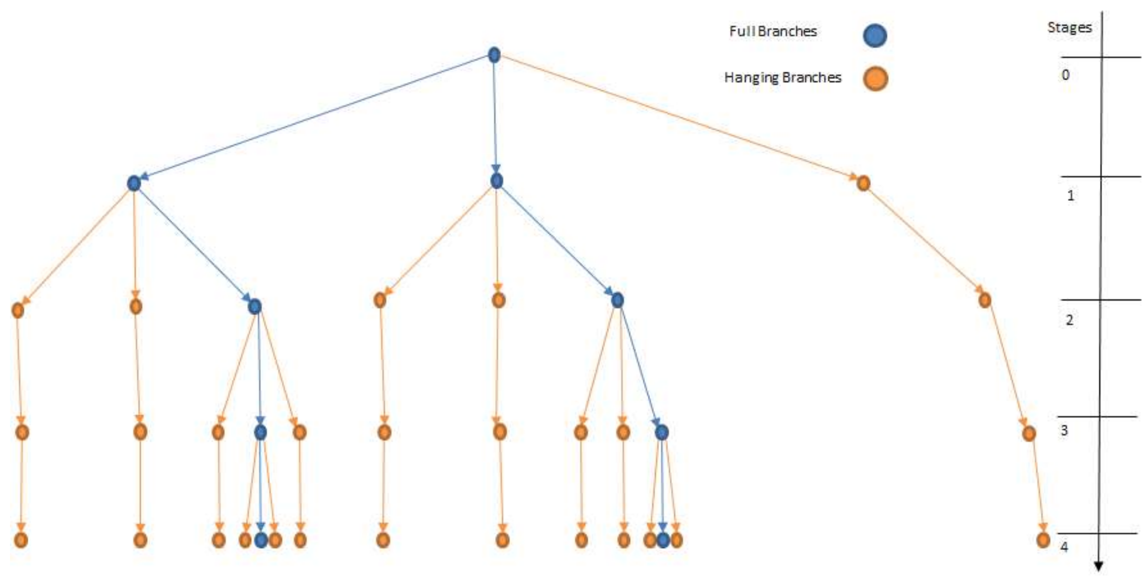


Figure 22: Reduced multi-stage tree illustrating the equivalent tree.

As in SDDP, this method formulates a full recursive multi-stage stochastic problem.

This method in PLEXOS is defined using the Global class of objects as showed in figure below:

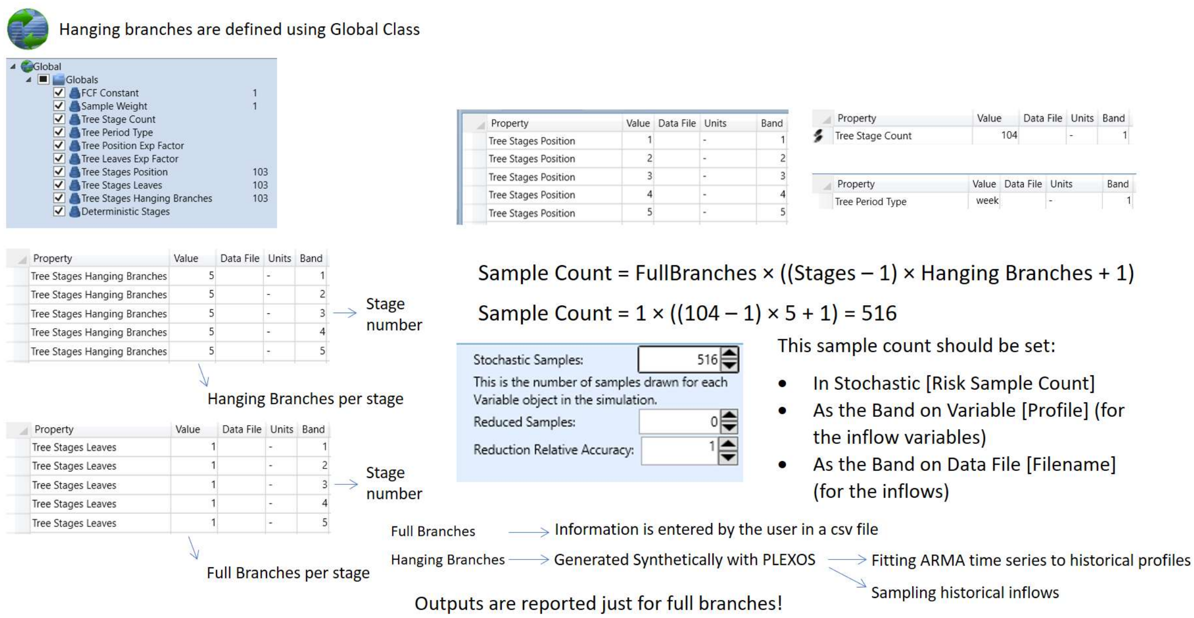


Figure 23: Hanging branches implementation in PLEXOS for 2 years horizon, 5 hanging branches and 1 full branch per stage and stages placed at the end of each week.

To help to improve the speed, the hanging branches were designed to have one block per stage from the stage that follows the stage they were created. This is like the equivalent SDDP reduced tree showed in Figure 20.

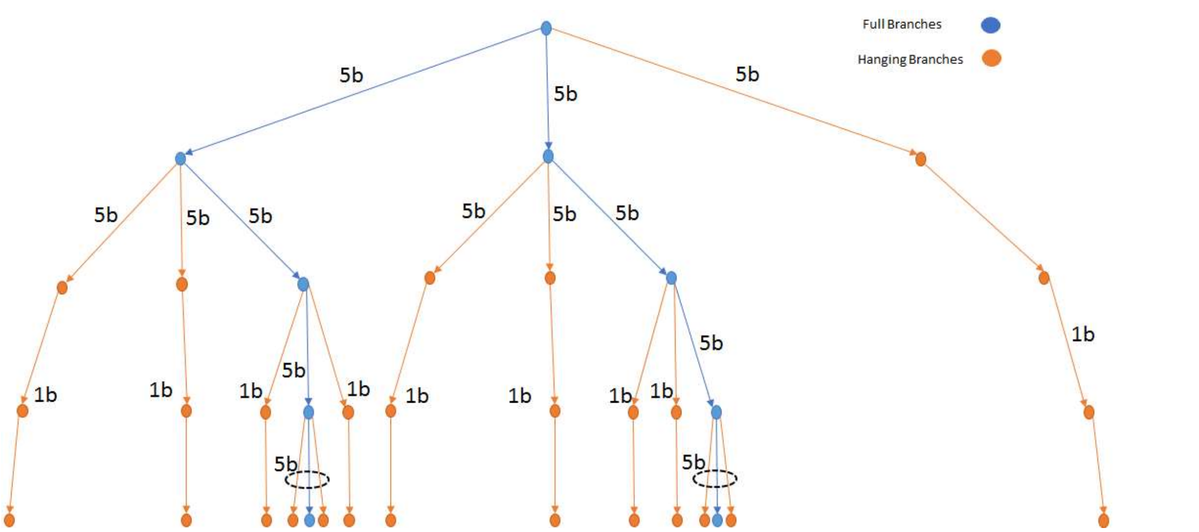


Figure 24: First stage of hanging branches have a block duration equal the blocks specified and the further stages are reduced to 1 block. This example is showing the reduction when the user specifies 5 blocks LDC.

Hanging branch weights

When the hanging branches equivalent multi-stage tree is formulated, it exists a solver limitation to solve the problem in one single step.

For example, if the user would like to solve this problem: 120 stages, 40 hanging branches per stage and 40 full branches for 10 years using 5 blocks LDC per month. This means the following:

- The first 41 branches (40 hanging + 1 full branch) has 1/41 probability occurrence.
- The second stage has 41 branches (40 hanging + 1 full branch) so each one has 1/41 probability of occurrence. So, from the first stage the probability for each second stage branch is 1/41*1/41.
- For next stages is the same

Then solving the problem from the first stage until the last one produces tiny weights for future branches, this is illustrated in the table below for the example described above:

Table 1: Brach Weight per stage solving the problem from root node

Stage Number	Probability	Probability
1	1/41	0.024390244
2	(1/41)^2	0.000594884
3	(1/41)^3	1.45094E-05
4	(1/41)^4	3.53887E-07
5	(1/41)^5	8.63139E-09
6	(1/41)^6	2.10522E-10
7	(1/41)^7	5.13468E-12
8	(1/41)^8	1.25236E-13
9	(1/41)^9	3.05454E-15
10	(1/41)^10	7.45009E-17
11	(1/41)^11	1.81709E-18
12	(1/41)^12	4.43194E-20
13	(1/41)^13	1.08096E-21
14	(1/41)^14	2.63649E-23
15	(1/41)^15	6.43046E-25
16	(1/41)^16	1.56841E-26
17	(1/41)^17	3.82538E-28
18	(1/41)^18	9.33019E-30
19	(1/41)^19	2.27566E-31
…	…	…

A safe range of objective coefficients guaranteed by a commercial solver is between 10^-6 and 10^6, that means it can only be ensured a multi-stage solution until stage three.

The multi-stage stochastic optimization says that at each stage the decision maker can change his mind and take a new decision because additional information is revealed, so this limitation can be solved in the same way a multi-stage stochastic problem is solved in the real life: using a *rolling horizon* approach splitting the horizon in steps. The only information passed between steps is related to storage end/initial volumes.

Hanging branches with rolling horizon

The Rolling Horizon approach is designed to overcome the limitation of vanishingly small probabilities deep into the future. The method looks ahead until certain point in the future and the end volumes in that point are passed as initial volumes at the start of the next step.

For a horizon divided in four different steps, the algorithm works as follows:

Step 1:

a) Starting date is beginning of root node.
b) Multi-stage tree formulated and up to some stage in the future (user decides when) no more branches to avoid weights issue.

Step 2:

a) Starting date is beginning of stage 1.
b) End volumes in step 1 are passed as initial volumes in step 2.
c) The past branches are not formulated because that part of the problem is already solved.

Step 3:

a) Starting date is beginning of stage 2.
b) End volumes in step 2 are passed as initial volumes in step 3.
c) The past branches are not formulated because that part of the problem is already solved.
d) More hanging branches are formulated when the weights provide information to the optimization solver.

Step 4:

a) Starting date is beginning of stage 3.
b) End volumes in step 3 are passed as initial volumes in step 4.
c) The past branches are not formulated because that part of the problem is already solved.
d) The problem becomes a simple deterministic problem since no more uncertainty is added because it is a **recursive** multi-stage problem.

Figure 25 shows the hanging branches method for a multi-stage problem consisting of 4 stages, 2 hanging branches per stage and 1 full branch where the horizon is divided in 4 different steps.

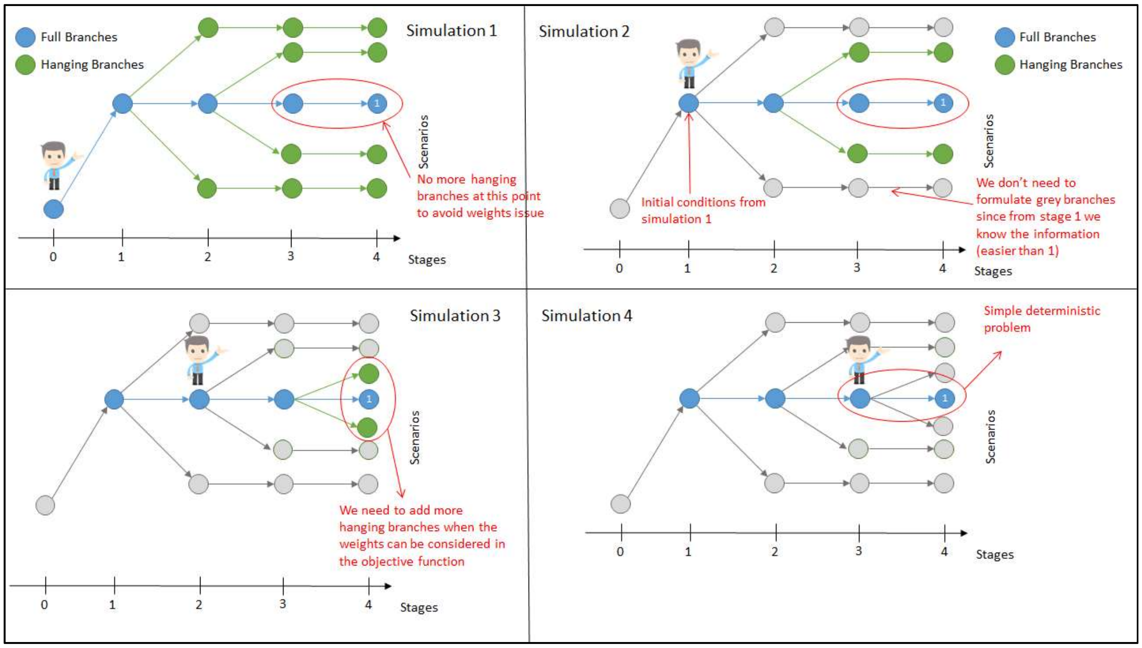


Figure 25: Rolling horizon for hanging branches

5.4 Comparison of different stochastic programming algorithms

Table 2 summarizes the difference between the three methods studied: scenario tree reduction, SDDP and Hanging Branches approach to solve stochastic multi-stage problems.

Table 2. Differences in multi-stage formulations.

	Scenario reduction	SDDP	Hanging Branches
Documented in literature?	Yes	Yes	No. Method was developed at Energy Exemplar
Recursive vs Non Recursive*	Non Recursive	Recursive	Recursive
Multi-stage tree exploration	No resampling	Some SDDP versions resample the tree allowing multiple exploration paths	No resampling
Speed solution	An equivalent SDDP tree is impossible to solve using today's available computers	It uses decomposition techniques and parallel resolution of independent sub problems to improve simulation performance.	It should equivalent to SDDP but faster than scenario reduction.
Linear or Integer?	Linear or MIP	Linear Only	Linear MIP
Implemented in PLEXOS?	Yes	No	Yes

6 Case Study

The following is a case study showing the resolution of the problem using SDDP and hanging branches method.

6.1 Data of the system

The following model has 1 hydro generator with storage, 1 thermal generator and 1 node. Each unit has a max capacity equal to 100 MW.

The operational costs (C) are 1 $/MWh for the thermal generator and 0 $/MWh for the hydro generator. The unserved energy cost (USE) is 10 $/MWh.

Table 3 summarizes generators data.

Table 3: Generator technical data

Generator	Name	Node	MSL (MW)	Max Capacity (MW)
Thermal	G_{th}	Gens	0	100
Hydro	G_h	Gens	0	100

Table 4 summarizes generator cost parameters.

Table 4: Generator costs parameters

Name	C ($/MWh)
C_{th}	1
C_h	0
C_{USE}	10

The horizon is segmented into 3 blocks. The first two blocks have 1 week duration and the third block has two weeks duration. The loads (D) are 90, 160 and 110 MW for blocks 1, 2 and 3 respectively.

The initial volume (v_0) of the storage is 60.48 Mm^3 and its max capacity is 100 Mm^3. The storage has recycle end effects with a penalty cost equal to 1.5 times unserved energy cost (1.5 USE). The hydro generator has 1 $MW/m^3/s$ efficiency. The inflow is 50 m^3/s for the first block; then there are 3 inflow possibilities for the second stage: 10, 50 or 90 m^3/s and the same 3 inflow possibilities for the last stage.

Table 5 and Table 6 summarize the additional input information.

Table 5: Storage properties

Min Vol (Mm3)	Max Vol (Mm3)	ρ ($MW/m^3/s$)
60.48	100	1

Table 6: General Information per block

Stages (t)	1	2	3
Duration (h)	168	168	336
Load (MW)	90	160	110
Inflow (m3/s)	50	10	40
		50	50
		90	60

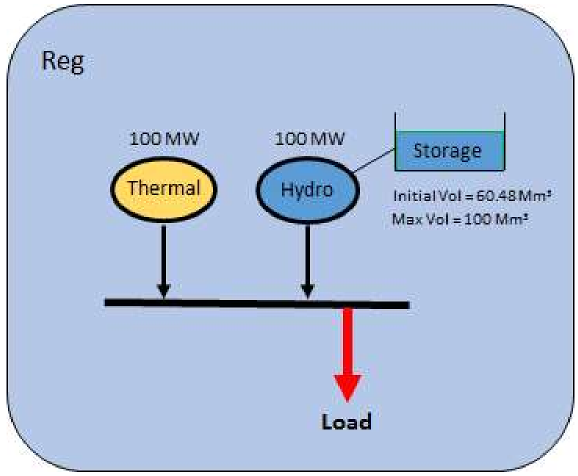


Figure 26. Representation of the Power System

The optimization problem determines the optimal dispatch of the system that minimizes production costs given inflows uncertainty. Each decision is re-evaluated at the beginning of each block when new inflow forecast arrives to the decision maker.

The problem is a stochastic multi-stage optimization problem that can be represented in mathematical terms as shown in equation (13) where each block represents a stage:

$$Z = min\left[\sum_{t=1}^{H}\sum_{k=1}^{K(t)} p_{t,k} \cdot \Delta t \cdot \left(C\_th \cdot G\_th_{t,k} + C\_USE \cdot USE_{t,k}\right)\right] \tag{13}$$

Subject to:

$$G\_h_{t,k} + G\_th_{t,k} + USE_{t,k} = D_t$$
$$G\_h_{t,k} = \rho \cdot q_{t,k}$$
$$v_{t,k} = v_{t-1,k} + I - R$$
$$\begin{bmatrix} G\_th_{min} \\ G\_h_{min} \\ 0 \\ v_{min} \end{bmatrix} \leq x = \begin{bmatrix} G\_th_{t,k} \\ G\_h_{t,k} \\ USE_{t,k} \\ v_{t,k} \end{bmatrix} \leq \begin{bmatrix} G\_th_{max} \\ G\_h_{max} \\ D_t \\ v_{max} \end{bmatrix}$$

Where:

$p_{t,k}$: Probability of sub-problem k occurring in stage t. Note that $\sum_{k=1}^{K} p_k = 1$
Δt: Duration of the stage
C_th: Cost of thermal generator per MWh
C_USE: cost of unserved energy per MWh
$G\_th_{t,k}$: Thermal generation of sub-problem k at stage t
$G\_h_{t,k}$: Hydro generation of sub-problem k at stage t
$USE_{t,k}$: Unserved energy of sub-problem k at stage t
$v_{t,k}$: Stored end volume of sub-problem k at stage t
$v_{t-1,k}$: Stored end volume of sub-problem k at stage t-1
D_t: Load in stage t
I: Inflow ($I = u \cdot A_{t,k}$)
R: Release ($R = u \cdot q_{t,k}$)
u: Hydro factor, refer to equation (14)
$A_{t,k}$: Inflow (m3/s) of sub-problem k at stage t

Equation (14) shows the value for factor "u" to convert hydro inflows into cubic meters.

$$u = u(\Delta t) = 3600\left[\frac{s}{h}\right]\cdot \Delta t[h] = 0.0036[Ms] \tag{14}$$

Because the stages have different durations then equation (14) takes the following values:

$$u_{1,2} = u(168) = 0.6048$$
$$u_3 = u(336) = 1.2096$$

By using the equations above, an inflow equal to 50 m^3/s stores 30.24 Mm^3 of water in the first and second stages, and 60.48 Mm^3 in the last stage.

1 MWh deviation in the end effect recycle condition is equal to the following cost per Mm^3:

$$\left[\frac{\$}{Mm^3}\right] = \left[\frac{\$}{MWh}\right]\cdot \rho\left[\frac{MW}{m^3/s}\right]\cdot \frac{10^6}{3600[s/h]} \tag{15}$$

$$= 1.5 \cdot 10 \cdot 1 \cdot 277.78\ [\$/Mm^3]$$
$$= 1.5 \cdot 2777.78\ [\$/Mm^3]$$

This value can be used to create a Future Cost Function (FCF) at the end of the planning horizon. The constraint is built using equation (16).

$$\propto_{t,k} \geq \alpha^*_{t,k} + \pi^*_{t,k}(v_{t,k} - v^*_{t-1,k}) \tag{16}$$

Equation (16) takes the following values

$$\alpha_{3,k} \geq 0 - 1.5 \cdot 2777.78 \cdot (v_{3,k} - 60.48) = 4166.67 \cdot (60.48 - v_{3,k})$$

Equation (13) can be rewritten in the following extensive form

$$Z = \min\left[G\_th_{1,1} + 10 \cdot USE_{1,1} + \frac{1}{3}\sum_{k=1}^{3}\left[G\_th_{2,k} + 10 \cdot USE_{2,k}\right] + \frac{1}{9}\sum_{k=1}^{9}\left[G\_th_{3,k} + 10 \cdot USE_{3,k}\right]\right]$$
$$G\_h_{1,1} + G\_th_{1,1} + USE_{1,1} = 90$$
$$G\_h_{2,1} + G\_th_{2,1} + USE_{2,1} = 160$$
$$G\_h_{2,2} + G\_th_{2,2} + USE_{2,2} = 160$$
$$G\_h_{2,3} + G\_th_{2,3} + USE_{2,3} = 160$$
$$G\_h_{3,1} + G\_th_{3,1} + USE_{3,1} = 110$$
$$G\_h_{3,2} + G\_th_{3,2} + USE_{3,2} = 110$$
$$G\_h_{3,3} + G\_th_{3,3} + USE_{3,3} = 110$$
$$G\_h_{3,4} + G\_th_{3,4} + USE_{3,4} = 110$$
$$G\_h_{3,5} + G\_th_{3,5} + USE_{3,5} = 110$$
$$G\_h_{3,6} + G\_th_{3,6} + USE_{3,6} = 110$$
$$G\_h_{3,7} + G\_th_{3,7} + USE_{3,7} = 110$$
$$G\_h_{3,8} + G\_th_{3,8} + USE_{3,8} = 110$$
$$G\_h_{3,9} + G\_th_{3,9} + USE_{3,9} = 110$$
$$v_{1,1} = v_0 + 30.24 - 0.6048 \cdot G\_h_{1,1}$$
$$v_{2,1} = v_{1,1} + 6.048 - 0.6048 \cdot G\_h_{2,1}$$
$$v_{2,2} = v_{1,1} + 30.24 - 0.6048 \cdot G\_h_{2,2}$$
$$v_{2,3} = v_{1,1} + 54.432 - 0.6048 \cdot G\_h_{2,3}$$
$$v_{3,1} = v_{2,1} + 48.384 - 1.2096 \cdot G\_h_{3,1}$$
$$v_{3,2} = v_{2,1} + 60.48 - 1.2096 \cdot G\_h_{3,2}$$
$$v_{3,3} = v_{2,1} + 72.576 - 1.2096 \cdot G\_h_{3,3}$$
$$v_{3,4} = v_{2,2} + 48.384 - 1.2096 \cdot G\_h_{3,4}$$

$$v_{3,5} = v_{2,2} + 60.48 - 1.2096 \cdot G\_h_{3,5}$$
$$v_{3,6} = v_{2,2} + 72.576 - 1.2096 \cdot G\_h_{3,6}$$
$$v_{3,7} = v_{2,3} + 48.384 - 1.2096 \cdot G\_h_{3,7}$$
$$v_{3,8} = v_{2,3} + 60.48 - 1.2096 \cdot G\_h_{3,8}$$
$$v_{3,9} = v_{2,3} + 72.576 - 1.2096 \cdot G\_h_{3,9}$$
$$\begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \le x = \begin{bmatrix} G\_th_{t,k} \\ G\_h_{t,k} \\ USE_{1,k} \\ USE_{2,k} \\ USE_{3,k} \\ v_{t,k} \end{bmatrix} \le \begin{bmatrix} 100 \\ 100 \\ 90 \\ 160 \\ 110 \\ 100 \end{bmatrix}$$

(17)

6.2 Model solution using SDDP algorithm.

Figure 27 illustrates the decision tree of the stochastic problem. It shows the sub-problems (t,k) together with the inflow data ($A_{t,k}$).

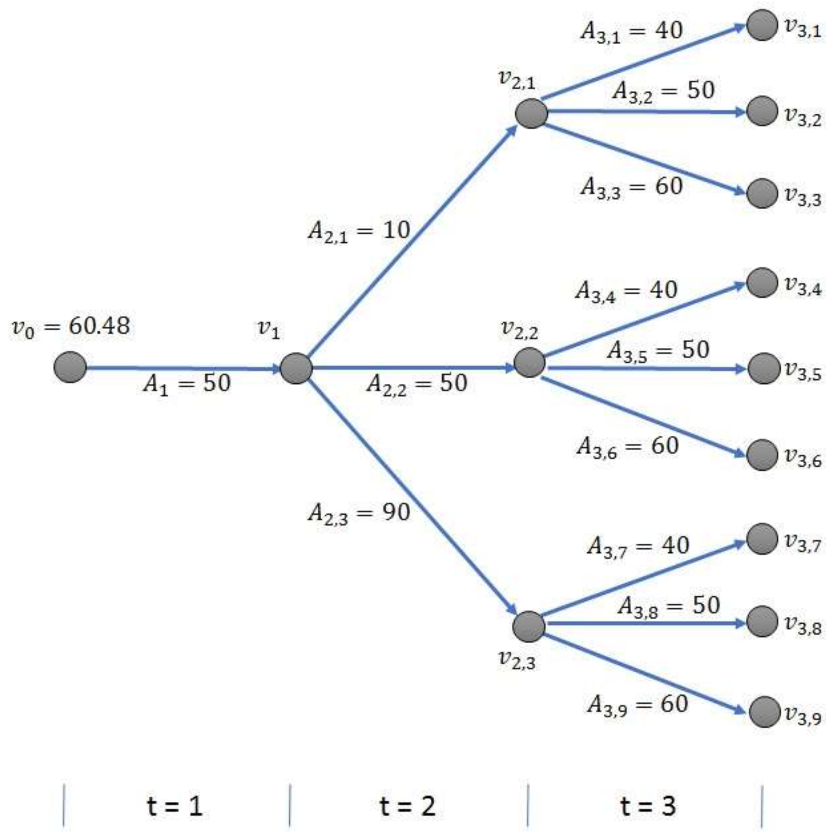


Figure 27. Decision Tree

When this system is solved using SDDP algorithm, 4 sub-problems are solved in forward pass (1 on the first stage and 3 on the second one) and 12 sub-problems are solved in backward pass (9 on the last stage and 3 on the second one). Equation (18) shows the sub-problem to be solved in each iteration.

$$Z_{t,k}(v_{t-1,k}) = min[\Delta t \cdot (C\_th \cdot G\_th_{t,k} + C\_USE \cdot USE_{t,k}) + \alpha_{t,k}] \tag{18}$$

$$G\_h_{t,k} + G\_th_{t,k} + USE_{t,k} \geq D^t$$
$$G\_h_{t,k} = \rho \cdot q_{t,k}$$
$$v_{t,k} = v_{t-1,k} + I - R \qquad \rightarrow \pi_t^*$$
$$\alpha_{t,k} \geq 0$$
$$\begin{bmatrix} G\_h_{min} \\ G\_th_{min} \\ 0 \\ v_{min} \end{bmatrix} \leq \begin{bmatrix} G\_h_{t,k} \\ G\_th_{t,k} \\ USE_{t,k} \\ v_{t,k} \end{bmatrix} \leq \begin{bmatrix} G\_h_{max} \\ G\_th_{max} \\ USE_t \\ v_{max} \end{bmatrix}$$

Where:

α: variable representing the expected future cost value of the following stage sub-problem

ITERATION 1

Direction: Forward Pass

Equation (18) takes the following form for the stage 1 on forward pass in iteration 1.

$$Z_1(0) = min[G\_th_1 + 10USE_1 + \alpha_1]$$
$$G\_th_1 + G\_h_1 + USE_1 = 90$$
$$v_1 = 90.72 - 0.6048 \cdot G\_h_1$$
$$\alpha_1 \geq 0$$
$$\begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \leq x = \begin{bmatrix} G\_h_1 \\ G\_th_1 \\ USE_1 \\ v_1 \end{bmatrix} \leq \begin{bmatrix} 100 \\ 100 \\ 90 \\ 100 \end{bmatrix}$$

Equation (18) takes the following forms for the stage 2 on forward pass in iteration 1.

$$Z_{2,1}(v_{1,1}) = min[G\_th_{2,1} + 10USE_{2,1} + \alpha_{2,1}]$$
$$G\_th_{2,1} + G\_h_{2,1} + USE_{2,1} = 160$$
$$v_{2,1} = v_{1,1} + u \cdot A_{2,1} - 0.6048 \cdot G\_h_{2,1}$$
$$\alpha_{2,1} \geq 0$$
$$\begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \leq x = \begin{bmatrix} G\_h_{2,1} \\ G\_th_{2,1} \\ USE_2 \\ v_2 \end{bmatrix} \leq \begin{bmatrix} 100 \\ 100 \\ 160 \\ 100 \end{bmatrix}$$

$$Z_{2,2}(v_{1,1}) = min[G\_th_{2,2} + 10USE_{2,2} + \alpha_{2,2}]$$
$$G\_th_{2,2} + G\_h_{2,2} + USE_{2,2} = 160$$
$$v_{2,2} = v_{1,1} + u \cdot A_{2,2} - 0.6048 \cdot G\_h_{2,2}$$
$$\alpha_{2,2} \geq 0$$
$$\begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \leq x = \begin{bmatrix} G\_h_{2,2} \\ G\_th_{2,2} \\ USE_2 \\ v_2 \end{bmatrix} \leq \begin{bmatrix} 100 \\ 100 \\ 160 \\ 100 \end{bmatrix}$$

$$Z_{2,3}(v_{1,1}) = min[G\_th_{2,3} + 10USE_{2,3} + \alpha_{2,3}]$$
$$G\_th_{2,3} + G\_h_{2,3} + USE_{2,3} = 160$$
$$v_{2,3} = v_{1,1} + u \cdot A_{2,3} - 0.6048 \cdot G\_h_{2,3}$$
$$\alpha_{2,3} \geq 0$$

$$\begin{bmatrix}0\\0\\0\\0\end{bmatrix} \le x = \begin{bmatrix}G\_h_{2,3}\\G\_th_{2,3}\\USE_3\\v_3\end{bmatrix} \le \begin{bmatrix}100\\100\\160\\100\end{bmatrix}$$

Because in this first pass we do not have approximation for the Future Cost Function (FCF), the stored volume at the end of the stages is zero (See Figure 28). This means that all the inflows were used to generate electricity, and the remaining load was supplied using the thermal gen and perhaps incur in unserved energy.

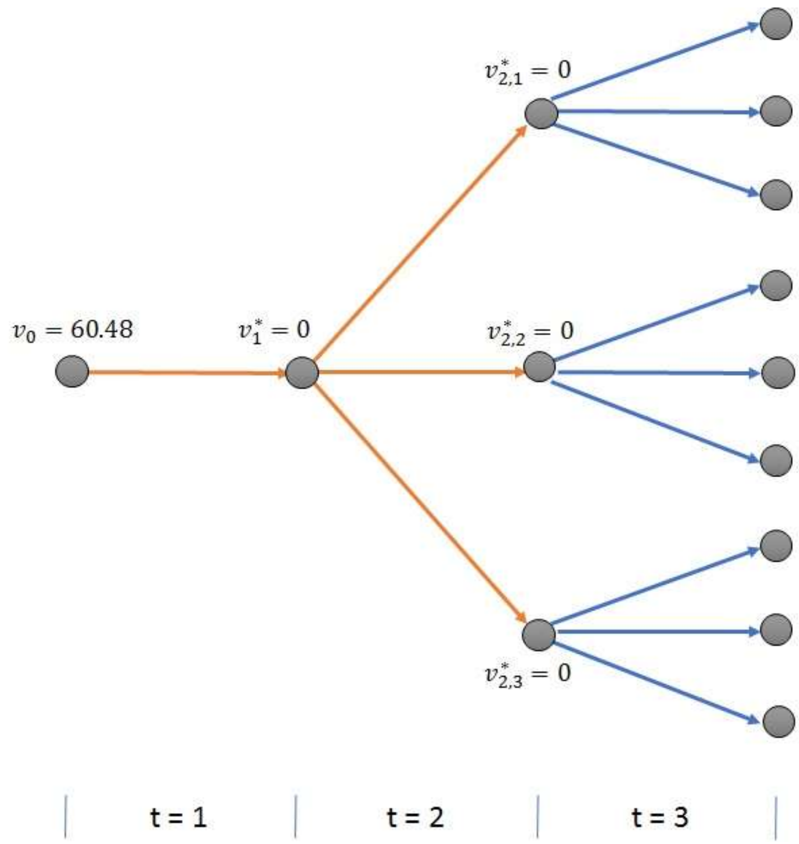


Figure 28. Iteration 1 Forward Pass.

The results for each sub-problems are presented below. Table 7 summarizes the costs of each sub-problem.

$$x_{1,1} = [G\_h_{1,1}, G\_th_{1,1}, USE_{1,1}, v_{1,1}]$$
$$x_{1,1} = [90, 0, 0, 36.29]$$
$$Z_{1,1}(0) = 0 + 10 \cdot 0 = \$0$$
$$x_{2,1} = [G\_h_{2,1}, G\_th_{2,1}, USE_{2,1}, v_{2,1}]$$
$$x_{2,1} = [70, 90, 0, 0]$$
$$Z_{2,1}(0) = 90 + 10 \cdot 0 = \$90/h \times 168h = \$15{,}120$$
$$x_{2,2} = [G\_h_{2,2}, G\_th_{2,2}, USE_{2,2}, v_{2,2}]$$
$$x_{2,2} = [100, 60, 0, 6.05]$$
$$Z_{2,2}(0) = 60 + 10 \cdot 10 = \$60/h \times 168h = \$10{,}080$$
$$x_{2,3} = [G\_h_{2,3}, G\_th_{2,3}, USE_{2,3}, v_{2,3}]$$
$$x_{2,3} = [100, 60, 0, 30.24]$$
$$Z_{2,3}(0) = 60 + 10 \cdot 0 = \$60/h \times 168h = \$10{,}080$$

Table 7: Results of Iteration 1 Forwards Pass.

t	K	$v^*_{t,k}$	AC	FCF
1	1	36.29	0	0
2	1	0	15120	0
	2	6.048	10080	0
	3	30.24	10080	0

As an example, the results of the sub-problem 1 of stage 2 means that the 10 m^3/s are used to generate 70 MW with the hydro gen and 90 MW with the thermal gen, so there is no unserved energy. Thus, the cost of the sub-problem is $100,800.

Direction: Backward Pass

Equation (18) takes the following form for the stage 3 on the backward pass of iteration 1.

$$Z_{3,1}(v_{2,1}) = min[G\_th_{3,1} + 10 \cdot USE_{3,1} + \alpha_{3,1}]$$
$$G\_th_{3,1} + G\_h_{3,1} + USE_{3,1} = 110$$
$$v_{3,1} = v_{2,1} + u \cdot A_{3,1} - 1.2096 \cdot G\_h_{3,1}$$
$$\alpha_{3,1} \geq 0$$
$$\alpha_{3,1} \geq 4166.67 \cdot (60.48 - v_{3,1})$$
$$\begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \leq x = \begin{bmatrix} G\_h_{3,1} \\ G\_th_{3,1} \\ USE_3 \\ v_3 \end{bmatrix} \leq \begin{bmatrix} 100 \\ 100 \\ 110 \\ 100 \end{bmatrix}$$

$$Z_{3,2}(v_{2,1}) = min[G\_th_{3,2} + 10 \cdot USE_{3,2} + \alpha_{3,2}]$$
$$G\_th_{3,2} + G\_h_{3,2} + USE_{3,2} = 110$$
$$v_{3,2} = v_{2,1} + u \cdot A_{3,2} - 1.2096 \cdot G\_h_{3,2}$$
$$\alpha_{3,2} \geq 0$$
$$\alpha_{3,2} \geq 4166.67 \cdot (60.48 - v_{3,2})$$
$$\begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \leq x = \begin{bmatrix} G\_h_{3,2} \\ G\_th_{3,2} \\ USE_3 \\ v_3 \end{bmatrix} \leq \begin{bmatrix} 100 \\ 100 \\ 110 \\ 100 \end{bmatrix}$$

$$Z_{3,3}(v_{2,1}) = min[G\_th_{3,3} + 10 \cdot USE_{3,3} + \alpha_{3,3}]$$
$$G\_th_{3,3} + G\_h_{3,3} + USE_{3,3} = 110$$
$$v_{3,3} = v_{2,1} + u \cdot A_{3,3} - 1.2096 \cdot G\_h_{3,3}$$
$$\alpha_{3,3} \geq 0$$
$$\alpha_{3,3} \geq 4166.67 \cdot (60.48 - v_{3,3})$$
$$\begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \leq x = \begin{bmatrix} G\_h_{3,3} \\ G\_th_{3,3} \\ USE_3 \\ v_3 \end{bmatrix} \leq \begin{bmatrix} 100 \\ 100 \\ 110 \\ 100 \end{bmatrix}$$

$$Z_{3,4}(v_{2,2}) = min[G\_th_{3,4} + 10 \cdot USE_{3,4} + \alpha_{3,4}]$$
$$G\_th_{3,4} + G\_h_{3,4} + USE_{3,4} = 110$$
$$v_{3,4} = v_{2,2} + u \cdot A_{3,4} - 1.2096 \cdot G\_h_{3,4}$$
$$\alpha_{3,4} \geq 0$$
$$\alpha_{3,4} \geq 4166.67 \cdot (60.48 - v_{3,4})$$

$$\begin{bmatrix}0\\0\\0\\0\end{bmatrix} \leq x = \begin{bmatrix}G\_h_{3,4}\\G\_th_{3,4}\\USE_3\\v_3\end{bmatrix} \leq \begin{bmatrix}100\\100\\110\\100\end{bmatrix}$$

$$Z_{3,5}(v_{2,2}) = min[G\_th_{3,5} + 10 \cdot USE_{3,5} + \alpha_{3,5}]$$
$$G\_th_{3,5} + G\_h_{3,5} + USE_{3,5} = 110$$
$$v_{3,5} = v_{2,2} + u \cdot A_{3,5} - 1.2096 \cdot G\_h_{3,5}$$
$$\alpha_{3,6} \geq 0$$
$$\alpha_{3,5} \geq 4166.67 \cdot (60.48 - v_{3,5})$$
$$\begin{bmatrix}0\\0\\0\\0\end{bmatrix} \leq x = \begin{bmatrix}G\_h_{3,5}\\G\_th_{3,5}\\USE_3\\v_3\end{bmatrix} \leq \begin{bmatrix}100\\100\\110\\100\end{bmatrix}$$

$$Z_{3,6}(v_{2,2}) = min[G\_th_{3,6} + 10 \cdot USE_{3,6} + \alpha_{3,6}]$$
$$G\_th_{3,6} + G\_h_{3,6} + USE_{3,6} = 110$$
$$v_{3,6} = v_{2,2} + u \cdot A_{3,6} - 1.2096 \cdot G\_h_{3,6}$$
$$\alpha_{3,6} \geq 0$$
$$\alpha_{3,6} \geq 4166.67 \cdot (60.48 - v_{3,6})$$
$$\begin{bmatrix}0\\0\\0\\0\end{bmatrix} \leq x = \begin{bmatrix}G\_h_{3,6}\\G\_th_{3,6}\\USE_3\\v_3\end{bmatrix} \leq \begin{bmatrix}100\\100\\110\\100\end{bmatrix}$$

$$Z_{3,7}(v_{2,3}) = min[G\_th_{3,7} + 10 \cdot USE_{3,7} + \alpha_{3,7}]$$
$$G\_th_{3,7} + G\_h_{3,7} + USE_{3,7} = 110$$
$$v_{3,7} = v_{2,3} + u \cdot A_{3,7} - 1.2096 \cdot G\_h_{3,7}$$
$$\alpha_{3,7} \geq 0$$
$$\alpha_{3,7} \geq 4166.67 \cdot (60.48 - v_{3,7})$$
$$\begin{bmatrix}0\\0\\0\\0\end{bmatrix} \leq x = \begin{bmatrix}G\_h_{3,7}\\G\_th_{3,7}\\USE_3\\v_3\end{bmatrix} \leq \begin{bmatrix}100\\100\\110\\100\end{bmatrix}$$

$$Z_{3,8}(v_{2,3}) = min[G\_th_{3,8} + 10 \cdot USE_{3,8} + \alpha_{3,8}]$$
$$G\_th_{3,8} + G\_h_{3,8} + USE_{3,8} = 110$$
$$v_{3,8} = v_{2,3} + u \cdot A_{3,8} - 1.2096 \cdot G\_h_{3,8}$$
$$\alpha_{3,8} \geq 0$$
$$\alpha_{3,8} \geq 4166.67 \cdot (60.48 - v_{3,8})$$
$$\begin{bmatrix}0\\0\\0\\0\end{bmatrix} \leq x = \begin{bmatrix}G\_h_{3,8}\\G\_th_{3,8}\\USE_3\\v_3\end{bmatrix} \leq \begin{bmatrix}100\\100\\110\\100\end{bmatrix}$$

$$Z_{3,9}(v_{2,3}) = min[G\_th_{3,9} + 10 \cdot USE_{3,9} + \alpha_{3,9}]$$
$$G\_th_{3,9} + G\_h_{3,9} + USE_{3,9} = 110$$
$$v_{3,9} = v_{2,3} + u \cdot A_{3,9} - 1.2096 \cdot G\_h_{3,9}$$
$$\alpha_{3,9} \geq 0$$
$$\alpha_{3,9} \geq 4166.67 \cdot (60.48 - v_{3,9})$$

$$\begin{bmatrix}0\\0\\0\\0\end{bmatrix} \le x = \begin{bmatrix}G\_h_{3,9}\\G\_th_{3,9}\\USE_3\\v_3\end{bmatrix} \le \begin{bmatrix}100\\100\\110\\100\end{bmatrix}$$

The FCF's are calculated using the related sub-problems, as shown in Figure 29. Figure 29 also shows the resultant volumes in the last stage.

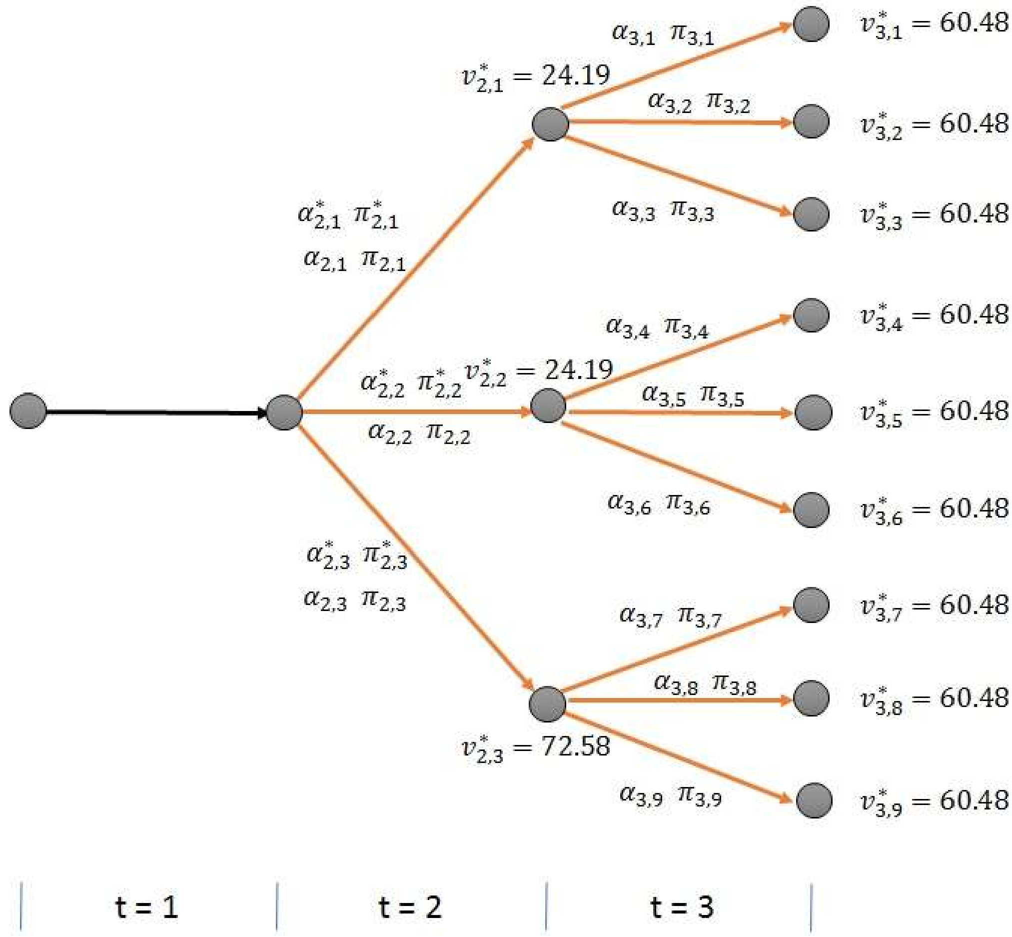


Figure 29. Iteration 1 Backward Pass.

The results for each sub-problem are as follows. Table 8 summarizes the costs of each sub-problem.

$$x_{3,1} = [0, 100, 10, 48.384]$$
$$Z_{3,1}(0) = 100 + 10 \cdot 10 = \$200/h \times 336h = \$67{,}200$$
$$x_{3,2} = [0, 100, 10, 60.48]$$
$$Z_{3,2}(0) = 100 + 10 \cdot 10 = \$200/h \times 336h = \$67{,}200$$
$$x_{3,3} = [10, 100, 0, 60.48]$$
$$Z_{3,3}(0) = 100 + 10 \cdot 0 = \$100/h \times 336h = \$33{,}600$$
$$x_{3,4} = [0, 100, 10, 54.432]$$
$$Z_{3,4}(0) = 100 + 10 \cdot 10 = \$200/h \times 336h = \$67{,}200$$
$$x_{3,5} = [5, 100, 5, 60.48]$$
$$Z_{3,5}(0) = 100 + 5 \cdot 10 = \$150/h \times 336h = \$50{,}400$$
$$x_{3,6} = [15, 95, 0, 60.48]$$
$$Z_{3,6}(0) = 95 + 10 \cdot 0 = \$95/h \times 336h = \$31{,}920$$
$$x_{3,7} = [15, 95, 0, 60.48]$$

$$Z_{3,7}(0) = 95 + 10 \cdot 0 = \$95/h \times 336h = \$31{,}920$$
$$x_{3,8} = [25, 85, 0, 60.48]$$
$$Z_{3,8}(0) = 85 + 10 \cdot 0 = \$85/h \times 336h = \$28{,}560$$
$$x_{3,9} = [35, 75, 0, 60.48]$$
$$Z_{3,9}(0) = 75 + 10 \cdot 0 = \$75/h \times 336h = \$25{,}200$$
$$x_{2,1} = [37.5, 100, 22.5, 19.66]$$
$$Z_{2,1}(0) = 100 + 10 \cdot 22.5 = \$325/h \times 168h = \$54{,}600$$
$$x_{2,2} = [60, 100, 0, 30.24]$$
$$Z_{2,2}(0) = 100 + 10 \cdot 0 = \$100/h \times 168h = \$16{,}800$$
$$x_{2,3} = [60, 100, 0, 54.432]$$
$$Z_{2,3}(0) = 100 + 10 \cdot 0 = \$100/h \times 168h = \$16{,}800$$

Table 8: Results of Iteration 1 Backward Pass

t	k	AC	FCF	$\pi_{t,k}$
3	1	67200	50400	4166.667
	2	67200	0	4166.667
	3	33600	0	2777.778
	4	67200	25200	4166.667
	5	50400	0	2777.778
	6	31920	0	277.7778
	7	31920	0	277.7778
	8	28560	0	277.7778
	9	25200	0	277.7778
2	1	54600	0	2777.778
	2	16800	0	2407.407
	3	16800	21840	277.7778

From the estimations of the third and second stage it is possible to calculate new approximations of FCF or Benders Cuts for the second and first stage, respectively.

First, it is necessary to weight the expected values of the dual variable (π) and the expected values of the optimal solution (α) according to the probability of occurrence of the sub-problem, as shown in equation (19) and equation (20).

$$\alpha^*_{t,k} = \sum_{k=1}^{K(t)} p_k \left(AC_{t+1,k} + FCF_{t+1,k}\right) \tag{19}$$

$$\pi^*_{t,k} = \sum_{k=1}^{K(t)} p_k \pi_{t+1,k} \tag{20}$$

Since the volumes obtained for each second stage sub-problem are different, then three different cuts are calculated for the second stage, using the corresponding sub-problem of stage three. Equation (19) takes the following values

$$\alpha^*_{2,1} = \left(\frac{1}{3}\right) \cdot (2 \cdot 67{,}200 + 33{,}600 + 50{,}400) = 72{,}800$$
$$\alpha^*_{2,2} = \left(\frac{1}{3}\right) \cdot (67{,}200 + 50{,}400 + 31{,}920 + 25{,}200) = 58{,}240$$

$$\alpha_{2,3}^{*} = \left(\frac{1}{3}\right) \cdot (31{,}920 + 28{,}560 + 25{,}200) = 28{,}560$$

$$\alpha_{1}^{*} = \left(\frac{1}{3}\right) \cdot (54{,}600 + 2 \cdot 16{,}800 + 21{,}840) = 36{,}680$$

Equation (20) takes the following values

$$\pi_{2,1}^{*} = \left(\frac{1}{3}\right) \cdot (2 \cdot 4{,}166.67 + 2777.78) = 3{,}703.70$$

$$\pi_{2,2}^{*} = \left(\frac{1}{3}\right) \cdot (4{,}166.67 + 2777.78 + \ 277.78) = 2{,}407.41$$

$$\pi_{2,3}^{*} = \left(\frac{1}{3}\right) \cdot (3 \cdot 277.78) = 277.78$$

$$\pi_{1}^{*} = \left(\frac{1}{3}\right) \cdot (2777.78 + 2407.41 + 277.78) = 1{,}820.99$$

The expected value of the dual variable and the expected value of the optimal solution together with equation (16) are used to build Benders Cut to add to the master problem.

$$\begin{aligned} \alpha_{2,1} &\geq 72800 - 3703.70 \cdot (v_{2,1} - 0) \qquad (21) \\ \alpha_{2,2} &\geq 58240 - 2407.41 \cdot (v_{2,2} - 6.05) \\ \alpha_{2,3} &\geq 28560 - 277.78 \cdot (v_{2,3} - 30.24) \\ \alpha_{1} &\geq 36680 - 1820.99 \cdot (v_{1} - 36.29) \qquad (22) \end{aligned}$$

Equation (21) shows the constraints that must be included in the sub-problems of the second stage. Similarly, equation (22) shows the constraint that must be included in the sub-problems of the first stage in the second iteration.

From Table 8 it can be observed that in stage 3, the natural inflows for sub -problems 1 and 4 (k_3= 1, 4) are not enough to meet recycle end volume conditions, so these sub problems are penalized with future costs. For these subproblems it can be observed that the thermal generators are generating at maximum capacity, the hydro gen is not generating and there are 10 MW of unserved energy. Therefore, the cost of these sub problems has two components, one based in the actual costs AC ($200) and the FCF estimation ($150). Both have to be multiplied for the stage duration, resulting in a total cost of $117,6000 ($67,200 + $50,400).

Convergence

The convergence is calculated in equation (25) using equation (23) for the Upper Bound, and equation (24) for the Lower Bound

$$Z_{upper} = \frac{1}{K}\sum_{k=1}^{K} AC_k \qquad (23)$$

$$Z_{lower} = AC_1 + FCF_1 \qquad (24)$$

$$\varepsilon = \frac{Z_{upper} - Z_{lower}}{Z_{upper}} \cdot 100 < \hat{\varepsilon} = 1\% \qquad (25)$$

Above equations take the following values:

$$Z_{upper} = 0 + \frac{(15120 + 2 \cdot 10080)}{3} + \frac{(3 \cdot 67200 + 33600 + 50400 + 2 \cdot 31920 + 28560 + 25200)}{9}$$
$$= 56,560$$

$$Z_{lower} = 0 + 0 = 0$$

$$\varepsilon = \frac{104,760 - 0}{104,760} \cdot 100 = 100\% > 1\%$$

ITERATION 2

Direction: Forward Pass

Equations (21) and (22) are added to the sub-problems of iteration 2 on the forward pass.

For stage 1:

$$Z_1(0) = min[G\_th_1 + 10USE_1 + \alpha_1]$$
$$G\_th_1 + G\_h_1 + USE_1 = 90$$
$$v_1 = 90.72 - 0.6048 \cdot G\_h_1$$
$$\alpha_1 \geq 0$$
$$\alpha_1 \geq 36680 - 1820.99 \cdot (v_1 - 36.29)$$
$$\begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \leq x = \begin{bmatrix} G\_h_1 \\ G\_th_1 \\ USE_1 \\ v_1 \end{bmatrix} \leq \begin{bmatrix} 100 \\ 100 \\ 90 \\ 100 \end{bmatrix}$$

For stage 2:

$$Z_{2,1}(v_{t-1}^*) = min\left[G\_th_{2,1} + 10USE_{2,1} + \alpha_{2,1}\right]$$
$$G\_th_{2,1} + G\_h_{2,1} + USE_{2,1} = 160$$
$$v_{2,1} = v_{1,1} + u \cdot A_{2,1} - 0.6048 \cdot G\_h_{2,1}$$
$$\alpha_{2,1} \geq 0$$
$$\alpha_{2,1} \geq 72800 - 3703.70 \cdot \left(v_{2,1} - 0\right)$$
$$\begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \leq x = \begin{bmatrix} G\_h_{2,1} \\ G\_th_{2,1} \\ USE_2 \\ v_2 \end{bmatrix} \leq \begin{bmatrix} 100 \\ 100 \\ 160 \\ 100 \end{bmatrix}$$

$$Z_{2,2}(v_{t-1}^*) = min\left[G\_th_{2,2} + 10USE_{2,2} + \alpha_{2,2}\right]$$
$$G\_th_{2,2} + G\_h_{2,2} + USE_{2,2} = 160$$
$$v_{2,2} = v_{1,1} + u \cdot A_{2,2} - 0.6048 \cdot G\_h_{2,2}$$
$$\alpha_{2,2} \geq 0$$
$$\alpha_{2,2} \geq 58240 - 2407.41 \cdot \left(v_{2,2} - 6.05\right)$$
$$\begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \leq x = \begin{bmatrix} G\_h_{2,2} \\ G\_th_{2,2} \\ USE_2 \\ v_2 \end{bmatrix} \leq \begin{bmatrix} 100 \\ 100 \\ 160 \\ 100 \end{bmatrix}$$

$$Z_{2,3}(v_{t-1}^*) = min\left[G\_th_{2,3} + 10USE_{2,3} + \alpha_{2,3}\right]$$
$$G\_th_{2,3} + G\_h_{2,3} + USE_{2,3} = 160$$
$$v_{2,3} = v_{1,1} + u \cdot A_{2,3} - 0.6048 \cdot G\_h_{2,3}$$
$$\alpha_{2,3} \geq 0$$
$$\alpha_{2,3} \geq 28560 - 277.78 \cdot \left(v_{2,3} - 30.24\right)$$

$$\begin{bmatrix}0\\0\\0\\0\end{bmatrix} \le x = \begin{bmatrix}G\_h_{2,3}\\G\_th_{2,3}\\USE_3\\v_3\end{bmatrix} \le \begin{bmatrix}100\\100\\160\\100\end{bmatrix}$$

We have now an approximation for the FCF, the stored volume at the end of stages 1 and 2 have a value different from 0 (see Figure 30).

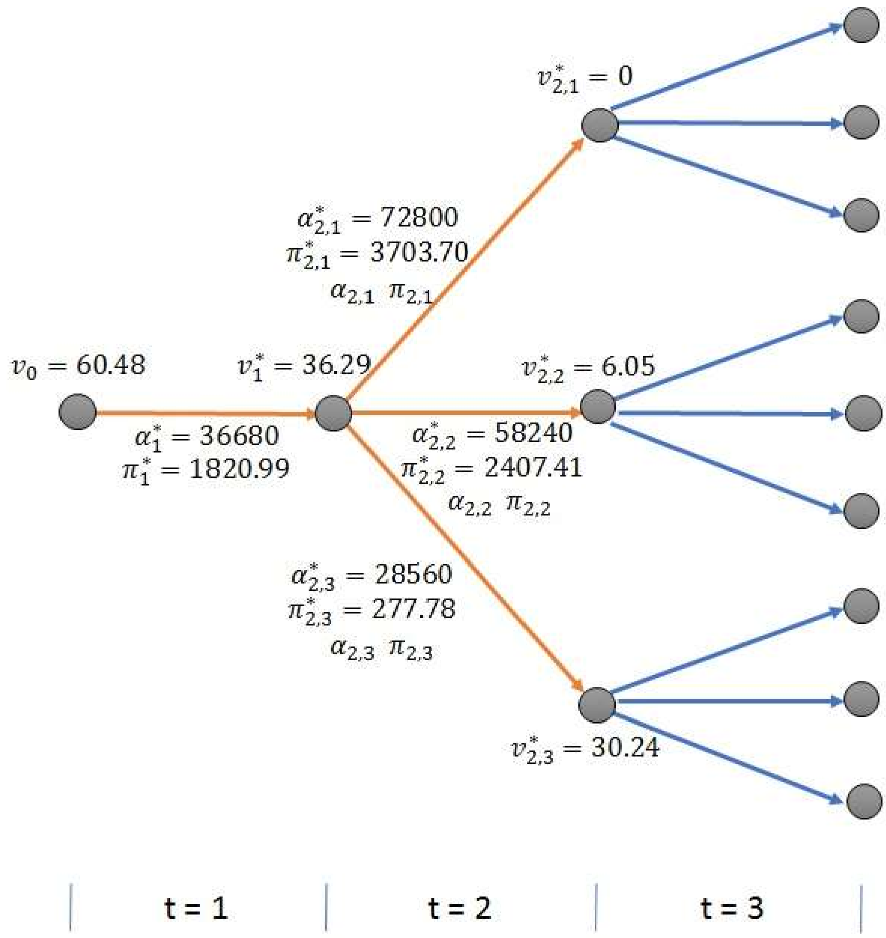


Figure 30. Iteration 2 Forward Pass.

Table 9 shows the results of iteration 2 in the forward pass direction.

Table 9: Results of Iteration 2 Forwards Pass

t	k	v^*_t	AC	FCF
1	1	56.43	5595.25	0
2	1	19.66	14984.75	0
	2	30.24	11204.75	0
	3	74.57	16800	16244.75

Direction: Backward Pass

Table 10 shows the results of iteration 2 in the backwards pass direction. This information can be used to generate new approximations.

Table 10: Results of Iteration 2 Backward Pass.

t	k	AC	FCF	$\pi_{t,k}$

3	1	46200	0	2777.78
	2	31500	0	277.78
	3	28140	0	277.78
	4	31920	0	277.78
	5	28560	0	277.78
	6	25200	0	277.78
	7	19605	0	277.78
	8	16245	0	277.78
	9	12885	0	277.78
2	1	16800	28018.98	111.11
	2	10080	29684.75	277.78
	3	16800	16244.75	277.78

Using the results in Table 10, three new cuts are calculated for the second stage (equation (26)) and one more for the first stage (equation (27)).

$$\alpha_{2,1} \geq 35280 - 111.11 \cdot \left(v_{2,1} - 19.66\right) \quad (26)$$
$$\alpha_{2,2} \geq 28560 - 277.78 \cdot \left(v_{2,2} - 30.24\right)$$
$$\alpha_{2,3} \geq 16244.75 - 277.78 \cdot \left(v_{2,3} - 74.57\right)$$
$$\alpha_1 \geq 39209.49 - 555.56 \cdot (v_1 - 56.43) \quad (27)$$

Convergence

The upper and lower bounds are higher than the desired gap as indicated below.

$$Z_{upper} = 46620$$
$$Z_{lower} = 5595.25$$
$$\varepsilon = 88\% > 1\%$$

ITERATION 3

Direction: Forward Pass

Equations (26) and (27) are added to the sub-problems of the forward pass of iteration 3.

For stage 1:

$$Z_1(0) = min[G\_th_1 + 10USE_1 + \alpha_1]$$
$$G\_th_1 + G\_h_1 + USE_1 = 90$$
$$v_1 = 30.24 - 0.6048 \cdot G\_h_1$$
$$\alpha_1 \geq 0$$
$$\alpha_1 \geq 36680 - 1820.99 \cdot (v_1 - 36.29)$$
$$\alpha_1 \geq 39209.49 - 555.56 \cdot (v_1 - 56.43)$$
$$\begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \leq x = \begin{bmatrix} G\_h_1 \\ G\_th_1 \\ USE_1 \\ v_1 \end{bmatrix} \leq \begin{bmatrix} 100 \\ 100 \\ 90 \\ 100 \end{bmatrix}$$

For stage 2:

$$Z_{2,1}(v_{t-1}^*) = min\left[G\_th_{2,1} + 10USE_{2,1} + \alpha_{2,1}\right]$$
$$G\_th_{2,1} + G\_h_{2,1} + USE_{2,1} = 160$$
$$v_{2,1} = v_{1,1} + u \cdot A_{2,1} - 0.6048 \cdot G\_h_{2,1}$$
$$\alpha_{2,1} \geq 0$$
$$\alpha_{2,1} \geq 72800 - 3703.70 \cdot \left(v_{2,1} - 0\right)$$
$$\alpha_{2,1} \geq 35280 - 111.11 \cdot \left(v_{2,1} - 19.66\right)$$

$$\begin{bmatrix}0\\0\\0\\0\end{bmatrix} \leq x = \begin{bmatrix}G\_h_{2,1}\\G\_th_{2,1}\\USE_2\\v_2\end{bmatrix} \leq \begin{bmatrix}100\\100\\160\\100\end{bmatrix}$$

$$Z_{2,2}(v_{t-1}^*) = min[G\_th_{2,2} + 10USE_{2,2} + \alpha_{2,2}]$$
$$G\_th_{2,2} + G\_h_{2,2} + USE_{2,2} = 160$$
$$v_{2,2} = v_{1,1} + u \cdot A_{2,2} - 0.6048 \cdot G\_h_{2,2}$$
$$\alpha_{2,2} \geq 0$$
$$\alpha_{2,2} \geq 58240 - 2407.41 \cdot (v_{2,2} - 6.05)$$
$$\alpha_{2,2} \geq 28560 - 277.78 \cdot (v_{2,2} - 30.24)$$
$$\begin{bmatrix}0\\0\\0\\0\end{bmatrix} \leq x = \begin{bmatrix}G\_h_{2,2}\\G\_th_{2,2}\\USE_2\\v_2\end{bmatrix} \leq \begin{bmatrix}100\\100\\160\\100\end{bmatrix}$$

$$Z_{2,3}(v_{t-1}^*) = min[G\_th_{2,3} + 10USE_{2,3} + \alpha_{2,3}]$$
$$G\_th_{2,3} + G\_h_{2,3} + USE_{2,3} = 160$$
$$v_{2,3} = v_{1,1} + u \cdot A_{2,3} - 0.6048 \cdot G\_h_{2,3}$$
$$\alpha_{2,3} \geq 0$$
$$\alpha_{2,3} \geq 28560 - 277.78 \cdot (v_{2,3} - 30.24)$$
$$\alpha_{2,3} \geq 16244.75 - 277.78 \cdot (v_{2,3} - 74.57)$$
$$\begin{bmatrix}0\\0\\0\\0\end{bmatrix} \leq x = \begin{bmatrix}G\_h_{2,3}\\G\_th_{2,3}\\USE_3\\v_3\end{bmatrix} \leq \begin{bmatrix}100\\100\\160\\100\end{bmatrix}$$

It can be observed that at the end of first stage, water has a higher opportunity cost in the future stages, so at the end of the first stage the solution stores a higher volume of water, as shown in Table 11.

Table 11: Results of Iteration 3 Forward Pass.

t	k	$v_{t,k}^*$	AC	FCF
1	1	90.72	15120	20160
2	1	36.29	10080	57120
	2	60.48	10080	36960
	3	84.67	10080	36960

Direction: Backward Pass

Table 12 shows the results of iteration 3 in the backward pass direction.

Table 12: Results of Iteration 3 Backward Pass.

t	k	AC	FCF	$\pi_{t,k}$
3	1	30240	0	277.78
	2	26880	0	277.78
	3	23520	0	277.78
	4	23520	0	277.78
	5	20160	0	277.78
	6	16800	0	277.78
	7	16800	0	277.78
	8	13440	0	277.78

	9	10080	0	277.78
2	1	10080	36960	0
	2	10080	20160	277.78
	3	14337.78	9182.22	277.78

There is information to calculate three more approximations for the second stage FCF (equation (28)) and one more for the first stage (equation (29)).

$$\alpha_{2,1} \geq 26880 - 277.78 \cdot \left(v_{2,1} - 36.29\right) \tag{28}$$
$$\alpha_{2,2} \geq 20160 - 277.78 \cdot \left(v_{2,2} - 60.48\right)$$
$$\alpha_{2,3} \geq 13440 - 277.78 \cdot \left(v_{2,3} - 84.67\right)$$
$$\alpha_1 \geq 33600 - 185.18 \cdot (v_1 - 90.72) \tag{29}$$

Convergence

The upper and lower bounds are higher than the desired gap as indicated below.

$$Z_{upper} = 45360$$
$$Z_{lower} = 35280$$
$$\varepsilon = 22.22\% > 1\%$$

ITERATION 4

Direction: Forward Pass

Equations (15) and (16) are added to the sub-problems of the forward pass of iteration 4.

For stage 1:

$$Z_1(0) = min[G\_th_1 + 10USE_1 + \alpha_1]$$
$$G\_th_1 + G\_h_1 + USE_1 = 90$$
$$v_1 = 30.24 - 0.6048 \cdot G\_h_1$$
$$\alpha_1 \geq 0$$
$$\alpha_1 \geq 36680 - 1820.99 \cdot (v_1 - 36.29)$$
$$\alpha_1 \geq 39209.49 - 555.56 \cdot (v_1 - 56.43)$$
$$\alpha_1 \geq 33600 - 185.18 \cdot (v_1 - 90.72)$$
$$\begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \leq x = \begin{bmatrix} G\_h_1 \\ G\_th_1 \\ USE_1 \\ v_1 \end{bmatrix} \leq \begin{bmatrix} 100 \\ 100 \\ 90 \\ 100 \end{bmatrix}$$

For stage 2:

$$Z_{2,1}(v_{t-1}^*) = min\left[G\_th_{2,1} + 10USE_{2,1} + \alpha_{2,1}\right]$$
$$G\_th_{2,1} + G\_h_{2,1} + USE_{2,1} = 160$$
$$v_{2,1} = v_{1,1} + u \cdot A_{2,1} - 0.6048 \cdot G\_h_{2,1}$$
$$\alpha_{2,1} \geq 0$$
$$\alpha_{2,1} \geq 72800 - 3703.70 \cdot \left(v_{2,1} - 0\right)$$
$$\alpha_{2,1} \geq 35280 - 111.11 \cdot \left(v_{2,1} - 19.66\right)$$
$$\alpha_{2,1} \geq 26880 - 277.78 \cdot \left(v_{2,1} - 36.29\right)$$
$$\begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \leq x = \begin{bmatrix} G\_h_{2,1} \\ G\_th_{2,1} \\ USE_2 \\ v_2 \end{bmatrix} \leq \begin{bmatrix} 100 \\ 100 \\ 160 \\ 100 \end{bmatrix}$$

$$Z_{2,2}(v_{t-1}^*) = min\left[G\_th_{2,2} + 10USE_{2,2} + \alpha_{2,2}\right]$$
$$G\_th_{2,2} + G\_h_{2,2} + USE_{2,2} = 160$$

$$v_{2,2} = v_{1,1} + u \cdot A_{2,2} - 0.6048 \cdot G\_h_{2,2}$$
$$\alpha_{2,2} \geq 0$$
$$\alpha_{2,2} \geq 58240 - 2407.41 \cdot (v_{2,2} - 6.05)$$
$$\alpha_{2,2} \geq 28560 - 277.78 \cdot (v_{2,2} - 30.24)$$
$$\alpha_{2,2} \geq 20160 - 277.78 \cdot (v_{2,2} - 60.48)$$
$$\begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \leq x = \begin{bmatrix} G\_h_{2,2} \\ G\_th_{2,2} \\ USE_2 \\ v_2 \end{bmatrix} \leq \begin{bmatrix} 100 \\ 100 \\ 160 \\ 100 \end{bmatrix}$$

$$Z_{2,3}(v_{t-1}^{*}) = min\left[G\_th_{2,3} + 10USE_{2,3} + \alpha_{2,3}\right]$$
$$G\_th_{2,3} + G\_h_{2,3} + USE_{2,3} = 160$$
$$v_{2,3} = v_{1,1} + u \cdot A_{2,3} - 0.6048 \cdot G\_h_{2,3}$$
$$\alpha_{2,3} \geq 0$$
$$\alpha_{2,3} \geq 28560 - 277.78 \cdot (v_{2,3} - 30.24)$$
$$\alpha_{2,3} \geq 16244.75 - 277.78 \cdot (v_{2,3} - 74.57)$$
$$\alpha_{2,3} \geq 13440 - 277.78 \cdot (v_{2,3} - 84.67)$$
$$\begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \leq x = \begin{bmatrix} G\_h_{2,3} \\ G\_th_{2,3} \\ USE_3 \\ v_3 \end{bmatrix} \leq \begin{bmatrix} 100 \\ 100 \\ 160 \\ 100 \end{bmatrix}$$

Table 13 shows the results of iteration 4 in the forward pass direction.

Table 13: Results of iteration 4 Forwards Pass.

t	k	$v_{t,k}^{*}$	AC	FCF
1	1	54.43	5040	40320
2	1	24.19	16800	30240
	2	24.19	10080	30240
	3	72.58	16800	16800

Direction: Backward Pass

Table 14 shows the results of iteration 4 in the backward pass direction.

Table 14: Results of iteration 3 Backward Pass.

t	k	AC	FCF	$\pi_{t,k}$
3	1	33600	0	277.78
	2	30240	0	277.78
	3	26880	0	277.78
	4	33600	0	277.78
	5	30240	0	277.78
	6	26880	0	277.78
	7	20160	0	277.78
	8	16800	0	277.78
	9	13440	0	277.78
2	1	16800	30240	1111.11
	2	10080	30240	277.78
	3	16800	16800	277.78

Convergence

The upper and lower bounds are identical as indicated below, so the process is concluded and this iteration results are the optimal.

$$Z_{upper} = 45360$$
$$Z_{lower} = 45360$$
$$\varepsilon = 0 < 1\%$$

The expected cost of the problem is $45,360. Figure 31 show the optimal end volumes for each sub-problem.

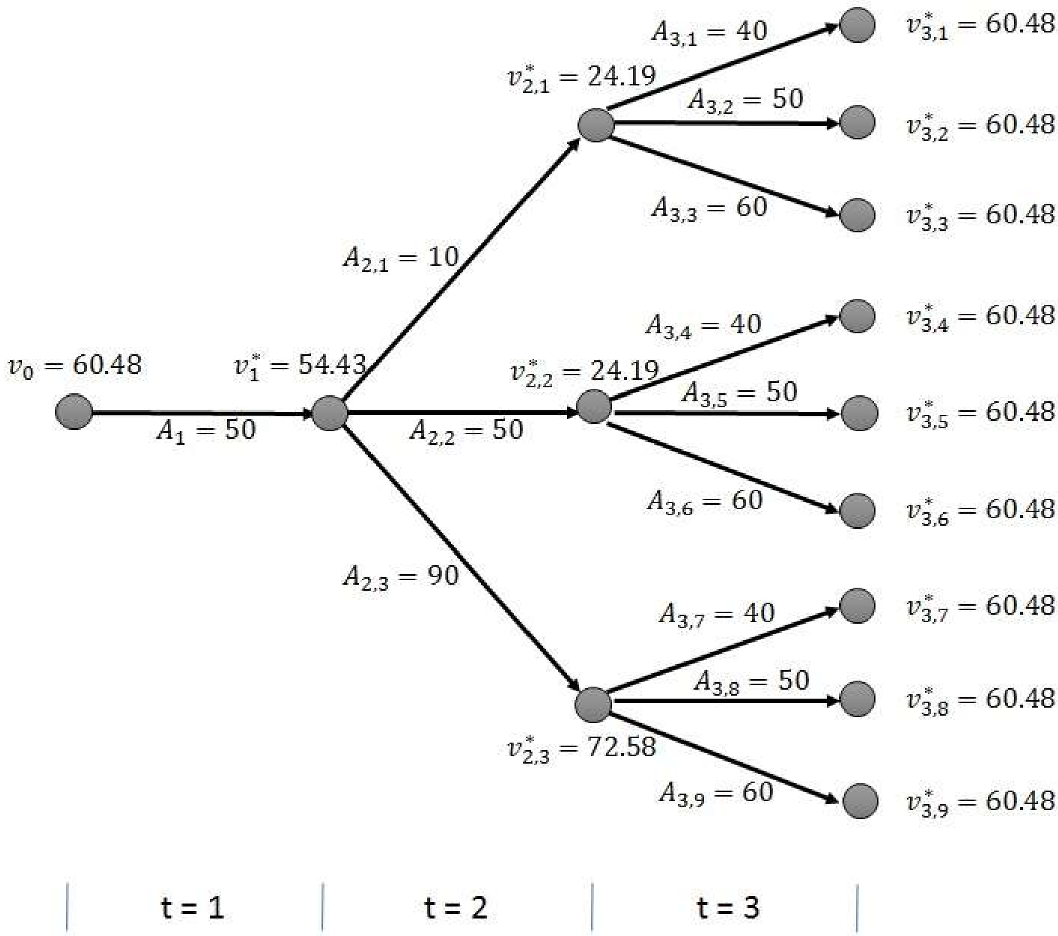


Figure 31. Optimum volumes per sub-problem.

6.3 Model solution using Hanging Branches

The equivalent tree using hanging branches method is showed below:

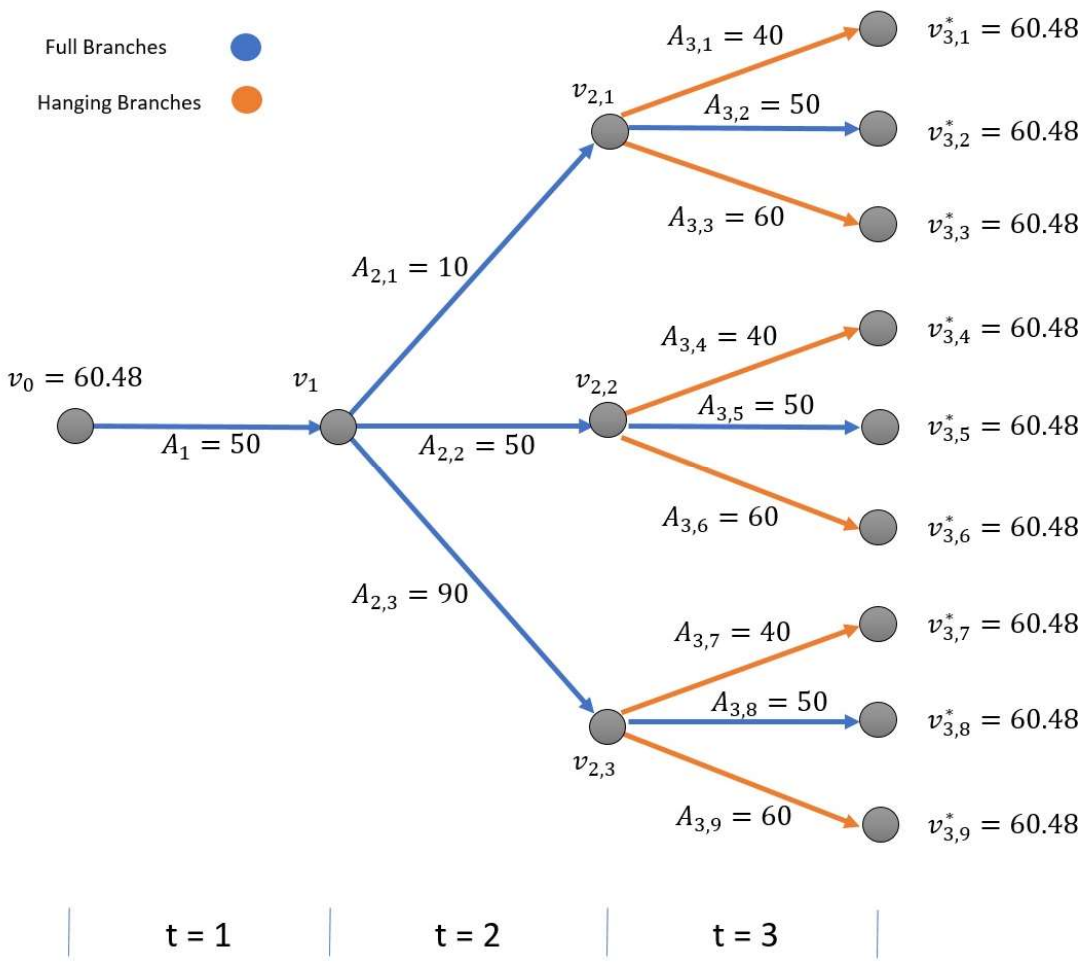


Figure 32: Multi-stage tree with full and hanging branches representation

The problem formulated as hanging branches formulation is the scenario – wise decomposition + non anticipativity constraint version of equation 17. The formulation can be summarized as follows:

$$Z = \min\left[\frac{1}{9}\sum_{k=1}^{9}\left[G\_th_{1,k} + 10 \cdot USE_{1,k} + G\_th_{2,k} + 10 \cdot USE_{2,k} + G\_th_{3,k} + 10 \cdot USE_{3,k}\right]\right]$$

$$G\_h_{1,k} + G\_th_{1,k} + USE_{1,k} = 90$$

$$G\_h_{2,k} + G\_th_{2,k} + USE_{2,k} = 160$$

$$G\_h_{3,k} + G\_th_{3,k} + USE_{3,k} = 110$$

$$v_{1,k} = v_{0,k} + 30.24 - 0.6048 \cdot G_{h_{1,k}}\,, \forall k = 1 \ldots 9$$

$$v_{2,k} = v_{1,k} + 6.048 - 0.6048 \cdot G\_h_{2,k}, \forall k = 1 \ldots 3$$

$$v_{2,k} = v_{1,k} + 30.24 - 0.6048 \cdot G\_h_{2,k}, \forall k = 4 \ldots 6$$

$$v_{2,k} = v_{1,k} + 54.432 - 0.6048 \cdot G\_h_{2,k}, \forall k = 7 \ldots 9$$

$$v_{3,k} = v_{2,k} + 48.384 - 1.2096 \cdot G\_h_{2,k}, \forall k = 1, 4, 7$$

$$v_{3,k} = v_{2,k} + 60.48 - 1.2096 \cdot G\_h_{2,k}, \forall k = 2, 5, 8$$

$$v_{3,k} = v_{2,k} + 72.576 - 1.2096 \cdot G\_h_{2,k}, \forall k = 3, 6, 9$$

$$\begin{bmatrix}0\\0\\0\\0\\0\\0\end{bmatrix} \le x = \begin{bmatrix}G\_th_{t,k}\\G\_h_{t,k}\\USE_{1,k}\\USE_{2,k}\\USE_{3,k}\\v_{t,k}\end{bmatrix} \le \begin{bmatrix}100\\100\\90\\160\\110\\100\end{bmatrix}$$

$$v_{1,1} = v_{1,2} = v_{1,3} = \cdots = v_{1,9}$$
$$v_{2,1} = v_{2,2} = v_{2,3}$$
$$v_{2,4} = v_{2,5} = v_{2,6}$$
$$v_{2,7} = v_{2,8} = v_{2,9}$$

To represent the example in PLEXOS is needed the following objects:

- 2 generators
- 1 storage
- 1 region
- 1 variable
- 1 global

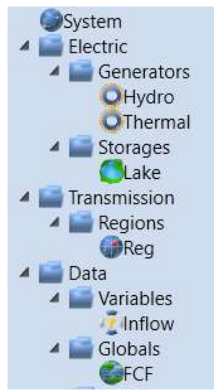
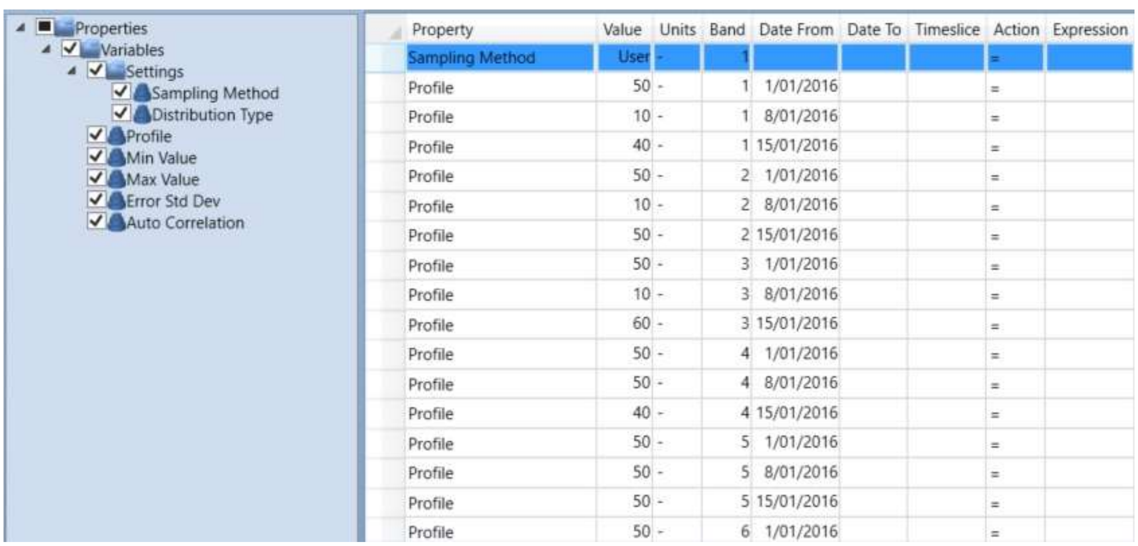


Figure 33: Objects tree representing the example



Property	Value	Units	Band	Date From	Date To	Timeslice	Action	Expression
Sampling Method	User	-	1				=	
Profile	50	-	1	1/01/2016			=	
Profile	10	-	1	8/01/2016			=	
Profile	40	-	1	15/01/2016			=	
Profile	50	-	2	1/01/2016			=	
Profile	10	-	2	8/01/2016			=	
Profile	50	-	2	15/01/2016			=	
Profile	50	-	3	1/01/2016			=	
Profile	10	-	3	8/01/2016			=	
Profile	60	-	3	15/01/2016			=	
Profile	50	-	4	1/01/2016			=	
Profile	50	-	4	8/01/2016			=	
Profile	40	-	4	15/01/2016			=	
Profile	50	-	5	1/01/2016			=	
Profile	50	-	5	8/01/2016			=	
Profile	50	-	5	15/01/2016			=	
Profile	50	-	6	1/01/2016			=	

Figure 34: Variable profile to represent uncertainty in PLEXOS

The global class has to be configured in the following way to have the tree modelled as in Figure 32.

Property	Value	Data File	Units	Band	Date From	Date To	Timeslice	Action	Expression	Scenario	Memo
Tree Stage Count	3		-	1				=			
Tree Period Type	interval		-	1				=			
Tree Stages Position	168		-	1				=			
Tree Stages Position	336		-	2				=			
Tree Stages Leaves	3		-	1				=			
Tree Stages Leaves	1		-	2				=			
Tree Stages Hanging Branches	0		-	1				=			
Tree Stages Hanging Branches	2		-	2				=			

Figure 35: Global class configuration

Once the problem is solved in MT, the objective function is 45,360 which is the same objective function value found using SDDP method in section 6.2.

```
MT Schedule Completed. Time: 00:00:02.1
------------------------------------------------------------------------------------------------------------------------
Minimization Objective Function:
    Linear Relaxation:.........................................................................          4.5360000000e+004
    Infeasibilities:...........................................................................                          0
```

Figure 36: Objective function value in log file

7 Conclusions

- The approach to solve hydro-thermal coordination optimization problems is to use stochastic optimization techniques to ensure the user minimizes the cost or alternatively maximizes the benefits of a hydro-thermal portfolio under uncertainty. A stochastic problem can be classified in:

  a) Two-stage or Multi-stages.
  b) Recursive or non-recursive.

  The current trend in the industry to solve a medium - long term stochastic hydro problems is formulating a multi-stage stochastic programming and recursive approach.

- The formulation of a multi-stage stochastic problem has dimensionality issues even when a few number of stages are added so simplifications on the resultant tree have to be made to produce a problem solvable by today computers.

- An equivalent SDDP tree is possible to formulate using scenario-wise decomposition and non-anticipativity constraints. This problem is possible to solve using today's computers.

- Hanging Branches method in PLEXOS looks promising to replace SDDP.

8 References

- *A multi-stage stochastic programming approach for production planning with uncertainty in the quality of raw materials and demand*. ZanjaniM.K.,Nourelfath, M. / Ait-Kadi, D.16, International Journal of Production Research, 4701-4723.
- ConejoA.J.,Carrión, M. & Morales, J.M. *Decision Making Under Uncertainty in Electricity Markerts*. Boston, MA, Springer U.S., 2010.
- NewhamN. *Power System Investment Planning using Stochastic Dual Dynamic Programming*. 2008.
- ReadE.Grant *Power System Optimisation*. , University of Canterbury, 1979.
- *Stochastic Dual Dynamic Programming with CVar Risk Constraints Applied to Hydrothermal Scheduling*. Da Costa JuniorPereira,Granville, Campodónico, Costa Fampa2013. International Conference on Stochastic Programming.
- *Stochastic optimization of a hydro-thermal system including network constraints*. al.Pereiraet2, , IEEE Transactions on Power Systems, 791-797.
- TroncosoC. *Herramienta docente para estudios de coordinación hidrotérmica*. 2010.

9 Annexes

9.1 Annex A: SDDP algorithm fundamentals

SDDP algorithm deterministic approach

Benders cuts take advantage of the following mathematical structure:

$$Min C_1 \cdot x_1 + C_2 \cdot x_2$$

Subject to

$$A_1 \cdot x_1 \geq b_1$$
$$E_1 \cdot x_1 + A_2 \cdot x_2 \geq b_2$$

This can be interpreted as a two-stage sequential decisions process. In the first stage we decide on a trial feasible value for X1 and given the trial value we find the optimal solution of the second stage function:

$$\alpha = Min C_2 \cdot x_2$$
$$A_2 \cdot x_2 \geq b_2 - E_1 \cdot x_1$$

X1 is known value in the second stage problem, and goes to the right hand side of the constraints. The objective then is to minimize the sum of the first – stage and second stage cost functions:

$$Min C_1 \cdot x_1 + \alpha$$

Subject to

$$A_1 \cdot x_1 \geq b_1$$

Where C1X1 represents the "actual cost" and α represents the "future cost" of decision X1. The future cost function translates the second stage costs as a function of the first stage decisions X1. If this function is available the problem can be solved as a one stage problem and then simplify the computation time.

The future cost function is approximated by an analytical function rather than a set of discrete values using a piecewise linear function. The structure of the future cost function can be characterized by taking the dual of the second stage problem:

$$\alpha = Max\left(\pi \cdot (b_2 - E_1 \cdot x_1)\right)$$

Subject to

$$\pi \cdot A_2 \leq C_2$$

π is the row vector of dual variables. From LP theory, optimal solution of dual and the original problem coincide. Since X1 is in the objective function and not in the right hand side of the constraint set as in the original problem, the set of possible solutions can be characterized before knowing the decision X1.

The problem can be solved by enumeration:

$$\alpha(X_1) = Max\left\{\pi^i(B_2 - E_1 X_1), for\ all\ i\right\}$$

This is equivalent to rewrite the problem as:

$$Min\ \alpha$$
$$\alpha \geq \pi^1(b_2 - E_1 \cdot x_1)$$
$$\vdots$$
$$\alpha \geq \pi^v(b_2 - E_1 \cdot x_1)$$

The problem can be rewritten as:

$$Min C_1 \cdot x_1 + \alpha$$

Subject to

$$A_1 \cdot x_1 \geq b_1$$
$$\alpha \geq \pi^1(b_2 - E_1 \cdot x_1)$$
$$\vdots$$
$$\alpha \geq \pi^v(b_2 - E_1 \cdot x_1)$$

We can rewrite the obtained cuts:

$$\alpha \geq \pi^1(b_2 - E_1 \cdot x_1)(*)$$

If $w^*$is optimal solution of:

$$\alpha = Min(c_2 \cdot x_2)$$
$$A_2 \cdot x_2 \geq b_2 - E_1 \cdot x_1$$

Assuming dual and primal has the same value we can write:

$$w^* = \pi^*(b_2 - E_1 \cdot x_1^*) \rightarrow \pi^* \cdot b_2 = w^* + \pi^* \cdot E_1 \cdot x_1^*$$

Substituting previous expression in cuts formulation (*), we can get an alternative expression for FCF:

$$\alpha \geq w^* + \pi^* \cdot E_1 \cdot (x_1^* - x_1)$$

Hydro problems share same mathematical structure described above where the link between stages is the hydro balance equation:

$$EndVolume(2) = EndVolume(1) + Inflow - Release$$

SDDP algorithm stochastic approach:

Stochastic problems can be written as:

$$MinC_1 \cdot x_1 + P_1 \cdot C_2 \cdot X_{21} + P_2 \cdot C_2 \cdot X_{22} + \cdots + P_m \cdot C_2 \cdot X_{2m}$$

Subject to

$$\begin{array}{llllr} A_1 \cdot x_1 & & & & \geq b_1 \\ E_1 \cdot x_1 + A_2 \cdot x_{21} & & & & \geq b_{21} \\ E_1 \cdot x_1 & + A_2 \cdot x_{22} & & & \geq b_{22} \\ & & \cdots & & \\ E_1 \cdot x_1 & & & + A_2 \cdot x_{2m} & \geq b_{2m} \end{array}$$

Were p1 and p2 are the probabilities to obtain b1 and b2. The second stage problem can be written as follow:

$$z = \min p_1 c_2 x_{21} + p_2 c_2 x_{22}$$

Subject to

$$\begin{array}{llr} A_2 x_{21} & & \geq b_{21} - E_1 x_1^* \\ & A_2 x_{22} & \geq b_{21} - E_1 x_1^* \end{array}$$

This problem can be decomposed into two independent problems:

$$\min c_2 x_{21}$$

Subject to

$$A_2 x_{21} \geq b_{21} - E_1 x_1^*$$
$$\min c_2 x_{22}$$

Subject to

$$A_2 x_{22} \geq b_{22} - E_1 x_1^*$$

The original problem can be rewritten as:

$$z = \min c_1 x_1 + \bar{\alpha}_1(x_1)$$

Subject to

$$A_1 x_1 \geq b_1$$

The function $\bar{\alpha}$ represents expected value of future cost function

$$\bar{\alpha}_1(x_1) = p_1 \alpha_{11}(x_1) + p_2 \alpha_{12}(x_1)$$

$$\alpha_{11}(x_1) = \min c_2 x_2$$

Subject to

$$A_2 x_2 \geq b_{21} - E_1 x_1 \rightarrow \pi_1$$

$$\alpha_{12}(x_1) = \min c_2 x_2$$

Subject to

$$A_2 x_2 \geq b_{22} - E_1 x_1 \rightarrow \pi_2$$

The benders cuts associated to this problem are:

$$p_1 \pi_1 (b_{21} - E_1 x_1) + p_2 \pi_2 (b_{22} - E_1 x_1) \leq \bar{\alpha}$$

That can be rewritten as:

$$p_1\big(w_1^* + \pi_1 E_1 (x_1^* - x_1)\big) + p_2\big(w_2^* + \pi_2 E_1 (x_1^* - x_1)\big) \leq \bar{\alpha}$$

Grouping the above equation we obtain the cut expression for stochastic problems:

$$\bar{w}^* + \bar{\pi} E_1 (x_1^* - x_1) \leq \bar{\alpha}$$

$$\bar{w}^* = p_1 w_1^* + p_2 w_2^* \;\; e \;\; \bar{\pi} = p_1 \pi_1 + p_2 \pi_2$$