GEMSTONE: a stochastic GEM

EPOC Winter Workshop 2011

Dr. Pierre Girardeau[1,2]
Joint work with Prof. Andy Philpott[2] and Jean Coussirou[3]

[1]EDF R&D, France

[2]Electric Power Optimization Centre

[3]ENSTA, France

September 2nd, 2011

Outline

Introduction

Long-term investment model for energy markets

We consider a **power producer** that has to face two types of decision:

1. **day-to-day power management** decisions,
2. and, once in a while, **investment decisions** for generation and transmission.

His aim is to satisfy the load at **minimum cost** over a long-term (around 30 years) horizon.

Introduction

What is GEM?

The Electricity Authority's Generation Expansion Model is a **capacity expansion model of the NZ electricity sector**. GEM models a wide range of large-scale electricity generation techniques, including: thermal, wind, hydro and wave.[a]

[a] Source: http://www.ea.govt.nz/industry/modelling/in-house-models/gem/

Inside the box

- Optimisation model (**mixed integer program**) solved using the CPLEX solver.
- **Deterministic** model.

Introduction

What is GEM?

The Electricity Authority's Generation Expansion Model is a **capacity expansion model of the NZ electricy sector**. GEM models a wide range of large-scale electricity generation techniques, including: thermal, wind, hydro and wave.[a]

[a] Source: http://www.ea.govt.nz/industry/modelling/in-house-models/gem/

Inside the box

- Optimisation model (**mixed integer program**) solved using the CPLEX solver.
- **Deterministic** model.

Introduction

GEMSTONE

GEM with STOchastic Network Expansion.

- Extend GEM to take **randomness** into account.
- Also motivated by a project on long-term investments in Europe with EDF R&D.

Stochastics

Two kinds of randomness

Every day randomness

- Hydrology, power demand, breakdowns, etc.
- Kind of randomness that affects the system **every minute**.
- Often called **fine grain randomness**.

Sporadic randomness

- Public policies, recession, earthquake, etc.
- Often called **coarse grain randomness**.
- These are **the types we will be modelling here**.

Stochastics

Two kinds of randomness

Every day randomness

- Hydrology, power demand, breakdowns, etc.
- Kind of randomness that affects the system **every minute**.
- Often called **fine grain randomness**.

Sporadic randomness

- Public policies, recession, earthquake, etc.
- Often called **coarse grain randomness**.
- These are **the types we will be modelling here**.

Stochastics

Why model it?

- A **deterministic** model assumes the **decision maker (DM) knows what is going to happen (optimistic)**.
- Note that it is the same when one solves several scenarios in parallel.
- Two features are important:
  1. You don't know what is going to happen in the future
- We will observe this in a small example shortly.

Stochastics

Why model it?

- A **deterministic** model assumes the **decision maker (DM) knows what is going to happen (optimistic)**.
- Note that it is the same when one solves several scenarios in parallel.
- Two features are important:
  1. You don't know what is going to happen in the future.
  2. Your decisions can adapt to randomness.
- We will observe this in a small example shortly.

Stochastics

Why model it?

- A **deterministic** model assumes the **decision maker (DM) knows what is going to happen (optimistic)**.
- Note that it is the same when one solves several scenarios in parallel.
- Two features are important:
  1. You **don't know what is going to happen** in the future.
  2. Your **decisions can adapt** to randomness.
- We will observe this in a small example shortly.

Stochastics

Why model it?

- A **deterministic** model assumes the **decision maker (DM) knows what is going to happen (optimistic)**.
- Note that it is the same when one solves several scenarios in parallel.
- Two features are important:
  1. You **don't know what is going to happen** in the future.
  2. Your **decisions can adapt** to randomness.
- We will observe this in a small example shortly.

Stochastics

Why model it?

- A **deterministic** model assumes the **decision maker (DM) knows what is going to happen (optimistic)**.
- Note that it is the same when one solves several scenarios in parallel.
- Two features are important:
  1. You **don't know what is going to happen** in the future.
  2. Your **decisions can adapt** to randomness.
- We will observe this in a small example shortly.

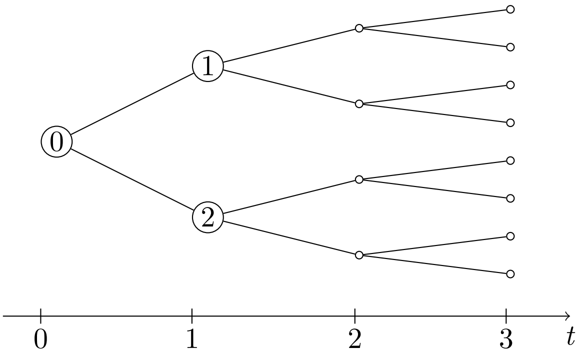
Stochastics

Why model it?

- A **deterministic** model assumes the **decision maker (DM) knows what is going to happen (optimistic)**.
- Note that it is the same when one solves several scenarios in parallel.
- Two features are important:
  1. You **don't know what is going to happen** in the future.
  2. Your **decisions can adapt** to randomness.
- We will observe this in a small example shortly.

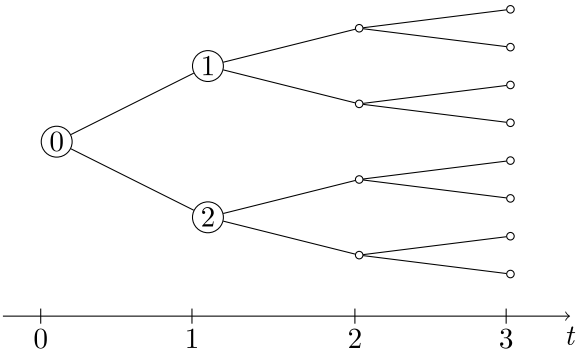
Outline

Scenario tree



- One has to choose the feature that changes if you fall into case 1 or case 2.
- In our case, the random events will be a sudden increase or decrease of the forecast load.

Scenario tree



- One has to choose the feature that changes if you fall into case 1 or case 2.
- In our case, the random events will be a sudden increase or decrease of the forecast load.

Motivation

A simple example of what we are looking for

Suppose we model:

- the **NZ power system over 21 years** (from 2010 to 2030),
- with **39 possible plants** to be built, including coal, geothermal, wind, peakers, etc.
- **1 possible upgrade of the HDVC** line (to add 400MW of capacity in both directions).

Moreover,

- the stochastic model branches three times: in 2015, 2020 and 2030;
- each time the load forecast can be multiplied either by 1.02 or 0.98.

Motivation

Answer of the deterministic model

Technology	Total investments	Possible investments
Coal, IGCC with CCS	0	3
Combined cycle gas turbine with CCS	0	2
Geothermal	2	3
Hydro, peaking	5	5
Hydro, pumped storage	0	1
Lignite, IGCC with CCS	0	1
Peaker, diesel-fired OCGT	8	8
Peaker, fast start gas-fired peaker	2	3
Price responsive load curtailment	1	4
Wind	3	9

Motivation

Answer of the stochastic model

Technology	Total investments	Possible investments
Coal, IGCC with CCS	0	3
Combined cycle gas turbine with CCS	0	2
Geothermal	1.875	3
Hydro, peaking	5	5
Hydro, pumped storage	0	1
Lignite, IGCC with CCS	0	1
Peaker, diesel-fired OCGT	8	8
Peaker, fast start gas-fired peaker	2	3
Price responsive load curtailment	1	4
Wind	3	9

Motivation
Answer of the stochastic model

Let us take a closer look at the differences.

Stochastic case

- Station GGeo2 is to be built only
  - in 2020 if the load goes up in 2015 and 2020,
  - or in 2022 if the load goes up in 2020.
- Station GGeo3 is to be built only
  - if the load goes up in 2015, 2020 and 2025.

Deterministic case

- Station GGeo2 is to be built regardless in 2022.
- Station GGeo3 is never built.

(HDVC upgrade is built anyway in 2016.)

Some notations

The set of all nodes in the tree is $\mathcal{N}$ and a particular node is denoted by n. Moreover we have:

- Φ_n Probability of passing through node n.
- $\mathcal{P}_n$ Set of all parents of n including itself.
- x'_n Investment decisions for node n (binary).
- c_n Investment costs at node n.
- y_n Production decisions for node n.
- q_n Production costs at node n.

We assume there is only one possible expansion per unit over the time horizon.

Scenario tree

Stochastic optimisation model

The problem

$$
\begin{aligned}
\min_{x',y} \quad & \sum_{n\in\mathcal{N}} \Phi_n \left(c_n^\top x'_n + q_n^\top y_n\right) \\
\text{s.t.} \quad & y_n \leq u_0 + \sum_{h\in\mathcal{P}_n} U x'_h, \qquad \forall n \in \mathcal{N}, \\
& \sum_{h\in\mathcal{P}_n} x'_h \leq 1, \qquad \forall n \in \mathcal{N}, \\
& y_n \in \mathcal{Y}_n, \qquad \forall n \in \mathcal{N}, \\
& x'_n \in \{0,1\}^F, \qquad \forall n \in \mathcal{N}.
\end{aligned}
$$

$\hookrightarrow$ In our case, $\mathcal{Y}_n$ includes stock constraints for reservoirs and network constraints.

Direct resolution

Size of the problem

- In the cases we considered, this MIP includes approximately:
  - 500 000 variables (1000 binaries),
  - 500 000 constraints,
  - 5 000 000 non-zeros in the constraint matrix.
- Commercial solvers are able to solve it up to a reasonable size, say 50 to 100 nodes (not much for a 20 years model).

Decomposition

General idea

- We are going to replace this big problem by the **iterative resolution of smaller problems** called subproblems.
- First, we introduce a new variable x_n and write an **equivalent formulation** of the initial problem.

Decomposition

Split-variable reformulation based on [Singh et al., 2009]

x_n states whether the plant has been built or not at node n.

Split-variable reformulation

$$
\begin{aligned}
\min_{x,x',y} \quad & \sum_{n\in\mathcal{N}} \Phi_n \left(c_n^\top x'_n + q_n^\top y_n \right) \\
\text{s.t.} \quad & x_n \le \sum_{h\in\mathcal{P}_n} x'_h, \qquad \forall n \in \mathcal{N} \\
& y_n \le u_0 + U x_n, \qquad \forall n \in \mathcal{N}, \\
& \sum_{h\in\mathcal{P}_n} x'_h \le 1, \qquad \forall n \in \mathcal{N}, \\
& y_n \in \mathcal{Y}_n, \qquad \forall n \in \mathcal{N}, \\
& x'_n \in \{0,1\}^F, \qquad \forall n \in \mathcal{N}.
\end{aligned}
$$

Decomposition
Enumeration

- Now any feasible x_n can be written as:

$$x_n = \sum_{j \in \mathcal{J}_n} \hat{x}_n^j w_n^j, \quad \sum_{j \in \mathcal{J}_n} w_n^j = 1, \quad w_n^j \in \{0, 1\}, \quad \forall j \in \mathcal{J}_n,$$

  where:

  - variables $\hat{x}_n^j$ are **feasible expansion plans** (FEP) at node n,
  - we denote by $\hat{y}_n^j$ one associated optimal **operational plan**,
  - $\mathcal{J}_n$ is the set of all possible feasible expansion plans.

- This leads to our master problem...

Decomposition
Enumeration

Master problem

$$\begin{aligned}
\min_{x',w} \quad & \sum_{n\in\mathcal{N}} \Phi_n c_n^\top x'_n + \sum_{n\in\mathcal{N}} \sum_{j\in\mathcal{J}_n} \Phi_n q_n^\top \hat{y}_n^j w_n^j \\
\text{s.t.} \quad & \sum_{j\in\mathcal{J}_n} \hat{x}_n^j w_n^j \le \sum_{h\in\mathcal{P}_n} x'_h, \qquad \forall n\in\mathcal{N} \\
& \sum_{h\in\mathcal{P}_n} x'_h \le 1, \quad x'_n \in \{0,1\}^F, \qquad \forall n\in\mathcal{N}, \\
& \sum_{j\in\mathcal{J}_n} w_n^j = 1, \quad w_n^j \in \{0,1\} \qquad \forall n\in\mathcal{N}.
\end{aligned}$$

$\hookrightarrow$ Problem: $\mathcal{J}_n$ is generally huge!

$\hookrightarrow$ Idea: Start with a small set $\mathcal{J}_n$ and add more and more columns as the algorithm progresses.

Decomposition
Prices

- Now suppose **you solved the master problem for a given set $\mathcal{J}_n$** of possible FEP.
- It gives you prices for constraints (dual variables).

Master problem

$$
\begin{aligned}
\min_{x',w} \quad & \sum_{n\in\mathcal{N}} \Phi_n c_n^\top x'_n + \sum_{n\in\mathcal{N}} \sum_{j\in\mathcal{J}_n} \Phi_n q_n^\top \hat{y}_n^j w_n^j \\
\text{s.t.} \quad & \sum_{j\in\mathcal{J}_n} \hat{x}_n^j w_n^j \leq \sum_{h\in\mathcal{P}_n} x'_h, \qquad \forall n\in\mathcal{N} \qquad \to \pi_n \\
& \sum_{h\in\mathcal{P}_n} x'_h \leq 1, \quad x'_n \in \{0,1\}^F, \qquad \forall n\in\mathcal{N}, \\
& \sum_{j\in\mathcal{J}_n} w_n^j = 1, \quad w_n^j \in \{0,1\} \qquad \forall n\in\mathcal{N} \qquad \to \mu_n.
\end{aligned}
$$

Decomposition

Prices

- Now suppose **you solved the master problem for a given set $\mathcal{J}_n$** of possible FEP.
- It gives you prices for constraints (dual variables).

Master problem

$$
\begin{aligned}
\min_{x',w} \quad & \sum_{n\in\mathcal{N}} \Phi_n c_n^\top x'_n + \sum_{n\in\mathcal{N}} \sum_{j\in\mathcal{J}_n} \Phi_n q_n^\top \hat{y}_n^j w_n^j \\
\text{s.t.} \quad & \sum_{j\in\mathcal{J}_n} \hat{x}_n^j w_n^j \le \sum_{h\in\mathcal{P}_n} x'_h, \qquad \forall n \in \mathcal{N} \qquad \to \pi_n \\
& \sum_{h\in\mathcal{P}_n} x'_h \le 1, \quad x'_n \in \{0,1\}^F, \qquad \forall n \in \mathcal{N}, \\
& \sum_{j\in\mathcal{J}_n} w_n^j = 1, \quad w_n^j \in \{0,1\} \qquad \forall n \in \mathcal{N} \qquad \to \mu_n.
\end{aligned}
$$

Decomposition
Subproblems

- Now the question is: for every node, **how to generate a new column** based on the prices for expansions given by the master problem?
- **We use these prices** $\hat{\pi}_n$ and $\hat{\mu}_n$.

Subproblem at node n

$$
\begin{aligned}
\min_{x_n, y_n} \quad & \underbrace{\Phi_n q_n^\top y_n}_{\text{cost of generation}} \underbrace{-\hat{\pi}_n^\top x_n - \hat{\mu}_n}_{\text{cost of changing the plan}} \\
\text{s.t.} \quad & y_n \leq u_0 + U x_n; \\
& y_n \in \mathcal{Y}_n; \\
& x_n \in \{0,1\}^F.
\end{aligned}
$$

Decomposition
Subproblems

- Now the question is: for every node, **how to generate a new column** based on the prices for expansions given by the master problem?
- **We use these prices** $\hat{\pi}_n$ and $\hat{\mu}_n$.

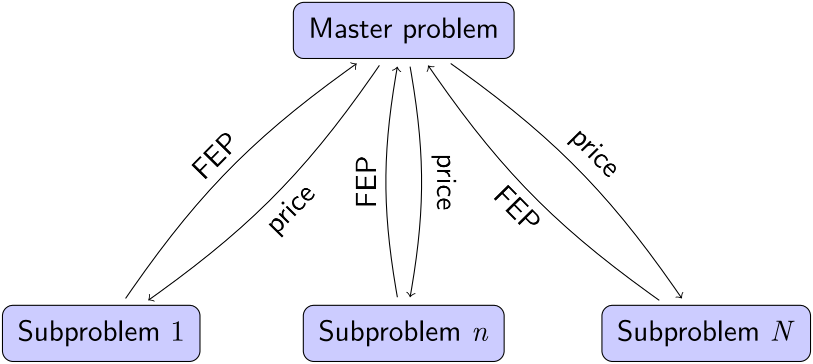
Subproblem at node n

$$\min_{x_n, y_n} \quad \underbrace{\Phi_n q_n^\top y_n}_{\text{cost of generation}} \underbrace{- \hat{\pi}_n{}^\top x_n - \hat{\mu}_n}_{\text{cost of changing the plan}}$$

$$\begin{aligned} \text{s.t.} \quad & y_n \leq u_0 + U x_n, \\ & y_n \in \mathcal{Y}_n, \\ & x_n \in \{0,1\}^F. \end{aligned}$$

Decomposition

The whole process



Decomposition

When does it stop?

Stopping criterion

- It stops **when no more useful FEP** can be added.
- In our case, this is **when no subproblem has a negative cost** anymore.

Decomposition
Strengths and weaknesses

Some remarks

- the approach has a great **flexibility**: no matter how, subproblems just have to send useful expansion plans;
- it **can be parallelized in a straightforward** way;
- however, the approach will be efficient given the **number of expansion possibilities is not too big**.

Outline

Numerical example

Two different models

We considered two different transshipment models for the NZ network.

2 zones

- South Island and North Island.
- Just one link (HDVC) with possible upgrade.
- 67 plants existing or commited to be built before 2030.
- 23 possible creations.

18 zones

- 8 nodes in North Island and 10 in South Island.
- 18 links with possible upgrades.
- 67 plants existing or commited to be built before 2030.
- 23 possible creations.

Numerical example

Cplex vs. Decomposition (on a Core 2 Duo, 2.4GHz, 2Go RAM, under Windows 7)

Scenario tree statistics				Cplex			Decomposition		
zones	branches	leaves	nodes	5%	1%	0%	5%	1%	0%
2	3	8	83	21	22.2	29.2	113.6	395.8	(0.05%)
2	2	4	44	7.4	8	11.1	35.6	173.7	1228.1
2	1	2	27	4.1	4.4	8	20.3	73.7	560.8
18	3	8	83	-	-	-	35	35	1756.9
18	2	4	44	422.3	422.3	(0.15%)	17.7	17.7	1112.8
18	1	2	27	96.1	147	(0.12%)	10.5	10.5	654.3

- Indications of time (in seconds) to achieve a given precision.
- If not finished before an hour, percentages within parenthesis indicate the precision obtained by this time.

Conclusion and perspectives

- **Stochastics are an important feature** to model in a long-term investment problem because
  1. in reality, the DM does not know which events are going to occur,
  2. and decisions can adapt to randomness.
- **It is possible to solve such a model** using dedicated decomposition schemes.

Some perspectives

- Include **loop flow constraints** for the network model.
- Improve the **modelling of hydrology** in the subproblems (uncertain inflows).

Conclusion and perspectives

- **Stochastics are an important feature** to model in a long-term investment problem because
  1. in reality, the DM does not know which events are going to occur,
  2. and decisions can adapt to randomness.
- **It is possible to solve such a model** using dedicated decomposition schemes.

Some perspectives

- Include **loop flow constraints** for the network model.
- Improve the **modelling of hydrology** in the subproblems (uncertain inflows).

Thanks for your attention

Here is a reference if you want to learn more.

Singh, K. J., Philpott, A. B., and Wood, R. K. (2009).
Dantzig-Wolfe Decomposition for Solving Multistage Stochastic Capacity-Planning Problems.
Operations Research, 57(5):1271–1286.