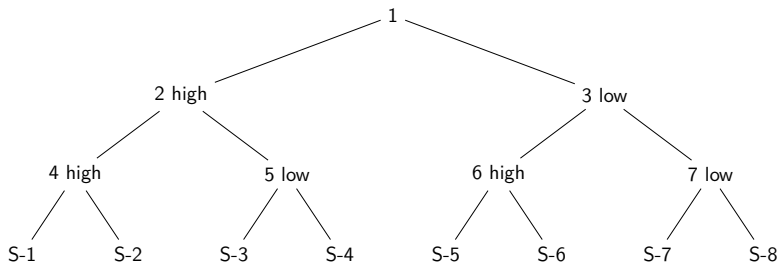


# Programmation Stochastique avec Recours

P. Hénaff

Version: 17 Dec 2024

## Arbre de décision



# Solution

Table 1: Optimal asset allocation at each node in the 3-period tree

Node	Stock	Bond
1	41479	13521
2	65095	2168
3	36743	22368
4	83840	0
5	0	71429
6	0	71429
7	64000	0

The excess or shortfall at horizon  $T$  for each scenario is displayed in Table @ref(tab:es).

## Solution Stochastique vs Miope

Table 3: Expected Value vs. Recourse solutions of the asset allocation problem.

Scenario	EV solution		Recourse solution	
	Shortfall	Surplus	Shortfall	Surplus
1		27422		24800
2		11094		8870
3		11094		1429
4	2752			
5		11094		1429
6	2752			
7	2752			
8	14494		12160	

# Lagrangien Augmenté

---

## Algorithm 1: Augmented Lagrangian algorithm

---

**Input** :  $\rho$ ,  $\text{tol}$ ,  $k_{\max}$

**Output:**  $x^*$

```

1  $k < -0; \lambda_k < -0$ 
2 while  $k < k_{\max}$  and  $t > \text{tol}$  do
3    $x_{k+1} < -\underset{x}{\text{argmin}} \phi_k(x)$ 
4    $\lambda_{k+1} < -\lambda_k - \rho g(x_{k+1})$ 
5    $k < -k + 1$ 
6    $t_1 < -||\lambda_{k+1} - \lambda_k||$ 
7    $t_2 < -||g(x_{k+1})||$ 
8    $t < -t_1 \text{ and } t_2$ 
9 end
```

---

## Algo Progressive Hedging

---

### Algorithm 2: Progressive Hedging algorithm

---

**Input** :  $\rho$ ,  $\text{tol}$ ,  $i_{\max}$

**Output:**  $x^*$

```

1   $i < -0$ ;  $\lambda_i < -0$ ; converged  $< -False$ 
2  while  $k < k_{\max}$  and  $!converged$  do
3      Solve the  $K$  subproblems ?? to obtain  $x_k^{i+1}, k = 1, \dots, K$ 
4      Compute  $\hat{x}^{i+1} = \sum_{k=1}^K \pi_k x_k^{i+1}$ 
5      Update the multipliers:  $\lambda_k^{i+1} = \lambda_k^i - \rho(x_k^{i+1} - \hat{x}^{i+1})$ 
6       $i < -i + 1$ 
7       $t_1 < -||\lambda^{i+1} - \lambda^i|| < \text{tol}_1$ 
8       $t_2 < -||g^1(x^{i+1})|| < \text{tol} < \text{tol}_2$ 
9       $t_3 < -||g^2(x^{i+1})|| < \text{tol} < \text{tol}_3$ 
10     converged  $< -t_1$  and  $t_2$  and  $t_3$ 
11 end
    
```

---

# First Example

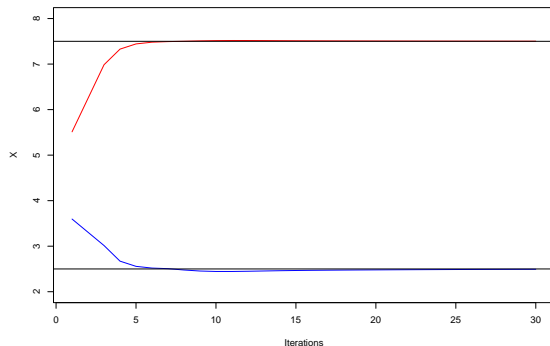


Figure 1: Progressive Hedging Iterations

## Second example: A 3-stage optimization problem