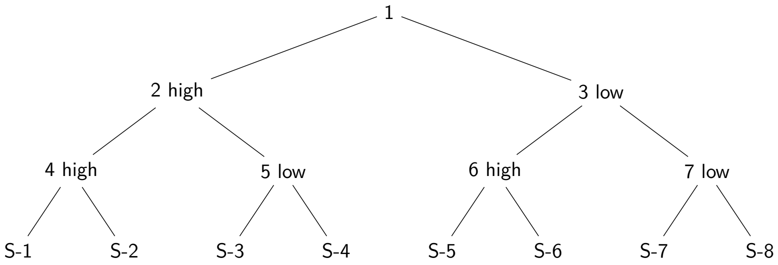
Programmation Stochastique avec Recours

P. Hénaff

Version: 06 Feb 2025

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.

	EV solution		Recourse solution	
Scenario	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 : ρ, tol, k_{max}

Output: x^*

$k < -0; \lambda_k < -0$

while $k < k_{max}$ *and* $t > tol$ **do**

$\quad x_{k+1} < -\underset{x}{\operatorname{argmin}} \phi_k(x)$

$\quad \lambda_{k+1} < -\lambda_k - \rho g(x_{k+1})$

$\quad k < -k + 1$

$\quad t_1 < -||\lambda_{k+1} - \lambda_k||$

$\quad t_2 < -||g(x_{k+1}||$

$\quad t < -t_1 \text{and} t_2$

end

Algo Progressive Hedging

Algorithm 2: Progressive Hedging algorithm

Input : ρ, tol, i_{max}

Output: x^*

```
i < −0; λ_i < −0; converged < −False
while k < k_max and !converged do
    Solve the K subproblems ?? to obtain x_k^{i+1}, k = 1, ..., K
    Compute x̂^{i+1} = ∑_{k=1}^{K} π_k x_k^{i+1}
    Update the multipliers: λ_k^{i+1} = λ_k^i − ρ(x_k^{i+1} − x̂^{i+1})
    i < −i + 1
    t_1 < −||λ^{i+1} − λ^i|| < tol_1
    t_2 < −||g^1(x^{i+1}|| < tol < tol_2
    t_3 < −||g^2(x^{i+1}|| < tol < tol_3
    converged < −t_1  and t_2  and t_3
end
```

First Example

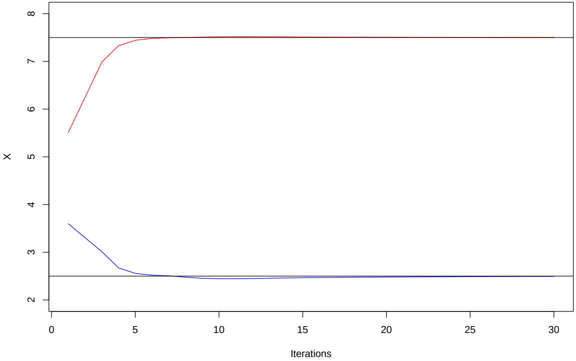


Figure 1: Progressive Hedging Iterations

Second example: A 3-stage optimization problem