A decomposition method by component for the optimization of maintenance scheduling

Mascot-Num 2020

September 18, 2020

T. Bittar[1, 2] P. Carpentier[3] J-Ph. Chancelier[1] J. Lonchampt[2]

[1] CERMICS, École des Ponts ParisTech

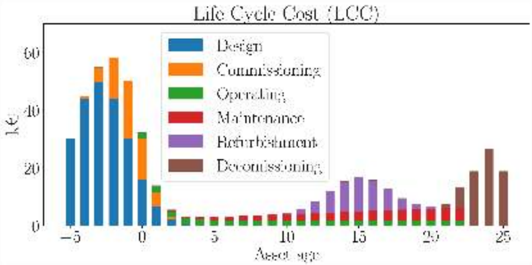
[2] EDF R&D, PRISME Department

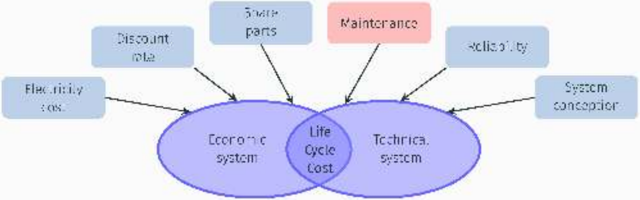
[3] ENSTA Paris

École des Ponts ParisTech

Short introduction to engineering asset management

Engineering asset management focuses on an integrative analysis of the life cycle of physical assets by using tools and methods for the quantification of technical and economic risks in order to support investment decisions.

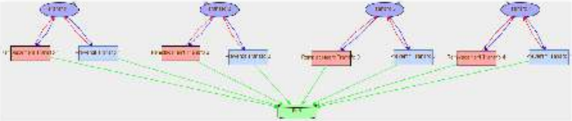




Type of systems considered in this work

- From 2 to 80 components of a hydroelectric power plant: turbines, generators, transformers

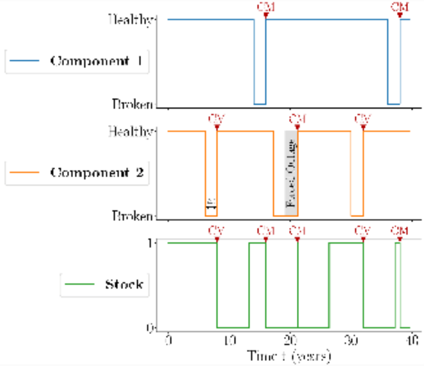
- Common stock of spares, initial stock with a low number of parts



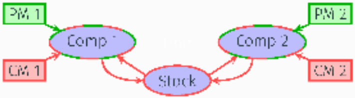
- Horizon of study: 40 years

Maintenance strategies and dynamics of the industrial system

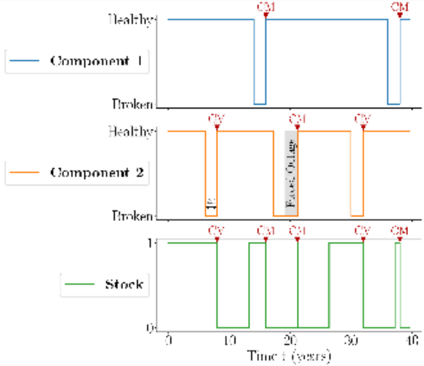
Reference strategy
Corrective maintenance only



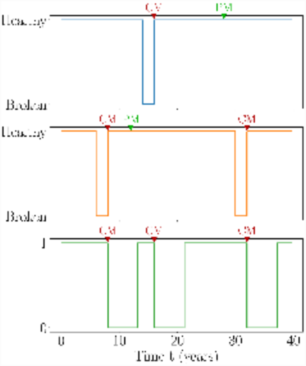
Maintenance strategies and dynamics of the industrial system



Reference strategy
Corrective maintenance only

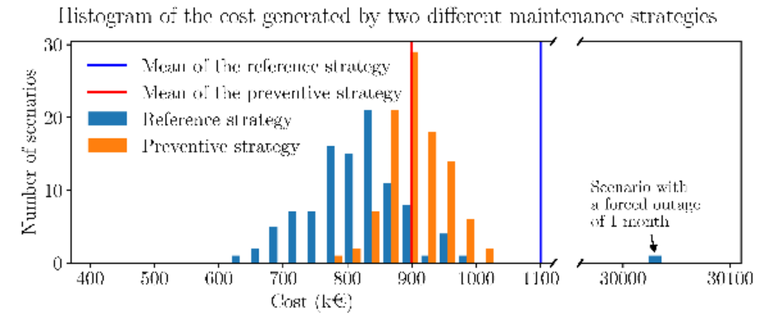


Preventive strategy
Corrective and Preventive maintenance



Order of magnitude of costs

- Costs of maintenance and forced outage have different order of magnitude:
  - Preventive maintenance: ∼ 100 k€
  - Corrective maintenance: ∼ 500 k€
  - Forced outage: ∼ 30000 k€/month
- Failures of the components are random events ⇒ LCC is a random variable
- Expected cost of a strategy estimated with Monte Carlo scenarios



Main goal

Industrial goal

For a given system, find the deterministic (open loop) maintenance strategy that minimizes the expectation of the LCC.

Optimization challenges:

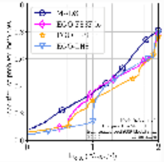
- Large-scale optimization problem (up to 80 components)
- Expected LCC computed with the simulation model VME: blackbox objective function

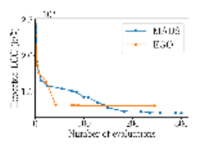
- VME uses Monte-Carlo simulations to estimate the expected LCC: no access to the true value of the objective but only noisy evaluations
- Evaluations of the objective function are expensive
  Industrial case with 80 components: $\sim$ 3h for one evaluation (with 10^5 scenarios)

Work summary

Blackbox optimization

- Use the simulation model VME: blackbox
- New variant of EGO with sequential initial design and metamodel validation
- Comparison with a direct search method





+ Accurate simulation of the dynamics
- Small space of maintenance strategies
- System with few components

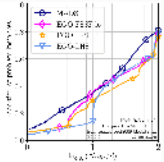
Contribution

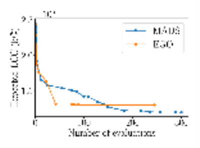
Blackbox approaches for optimal maintenance scheduling

Work summary

Blackbox optimization

- Use the simulation model VME: blackbox
- New variant of EGO with sequential initial design and metamodel validation
- Comparison with a direct search method





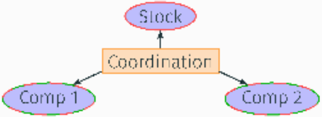
\+ Accurate simulation of the dynamics
\- Small space of maintenance strategies
\- System with few components

Contribution

Blackbox approaches for optimal maintenance scheduling

Stochastic optimal control

- Analytical expression of the dynamics
  → We open the blackbox!
- Maintenance optimization problem solved with a decomposition method



\- Simplified expression of the dynamics
\+ General maintenance strategies
\+ Large-scale systems

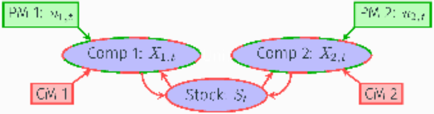
Contribution (submitted paper)

A decomposition method by interaction prediction for the optimization of maintenance scheduling [BCCL20-1]

Outline

Outline

Some important variables to characterize the system



Component i at time t characterized by $X_{i,t} = (E_{i,t}, A_{i,t})$ where:

$E_{i,t} \in \{0, 1\}$: Regime of the component

- $E_{i,t} = 1$: Healthy
- $E_{i,t} = 0$: Broken

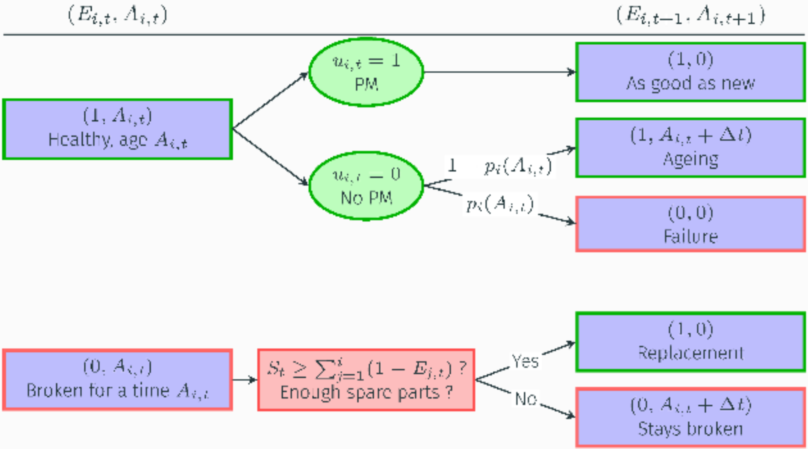
$A_{i,t} \in \mathbb{R}$:

- Age for a healthy component
- Time since last failure for a broken component

- $S_t \in \mathbb{N}$: Number of spare parts in the stock at time t
- $u_{i,t} \in \{0, 1\}$: Control on component i at time t
  - $u_{i,t} = 0$: No preventive maintenance
  - $u_{i,t} = 1$: Preventive maintenance

Dynamics of the components

- Time discretized with time step Δt
- Dynamics of a component from time step t to $t+1$:



Maintenance optimization problem

$$\min_{(X,S,u)\in\mathcal{X}\times\mathcal{S}\times\mathbb{U}} \mathbb{E}\Bigg(\sum_{i=1}^{n} \overbrace{\sum_{t=0}^{T} j_{i,t}(X_{i,t}, u_{i,t})}^{\text{Maintenance cost } j_i(X_i,u_i)} + \overbrace{\sum_{t=0}^{T} j_t^{FO}(X_{1,t},\dots,X_{n,t})}^{\text{Forced outage cost } j^{FO}(X_1,\dots,X_n)} \Bigg)$$

$$\text{s.t.} \quad \underbrace{X_{i,t+1} = f_i^X(X_{i,t}, S_t, u_{i,t}, W_{i,t+1})}_{\text{Dynamics of component } i}, \quad X_{i,0} = x_i \quad \forall t, \forall i$$

$$\underbrace{S_{t+1} = f^S(X_{1,t},\dots,X_{n,t}, S_t)}_{\text{Dynamics of the stock}}, \quad S_0 = s \quad \forall t$$

where $(W_{i,t})_{t=1,\dots,T}^{i=1,\dots,n}$ are random variables that model failure scenarios.

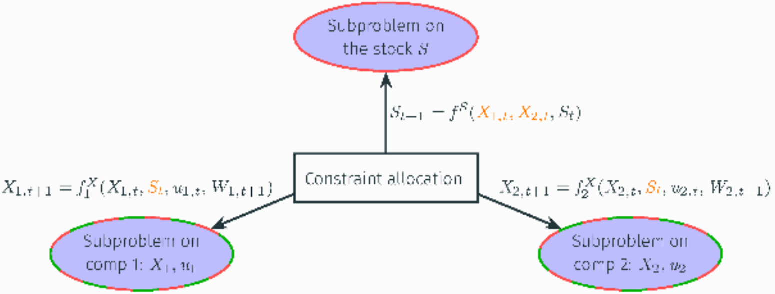
- Maintenance cost: additive in time and components
- Forced outage cost: additive in time, coupling the components

Outline

Sketch of a decomposition method by component

General idea

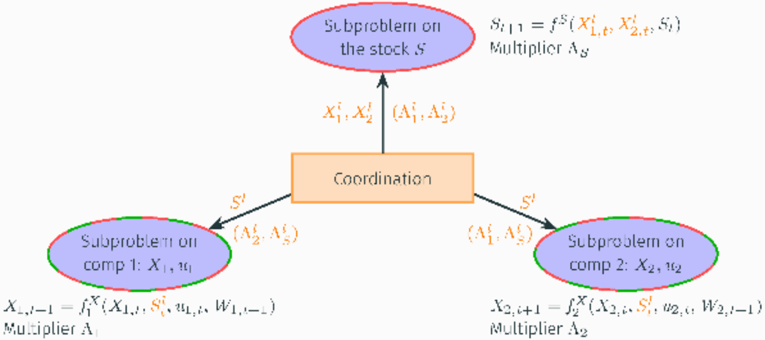
Use a decomposition by interaction prediction [MMT70] to iteratively find the best maintenance policy separately for each component and coordinate the components



Sketch of a decomposition method by component

General idea

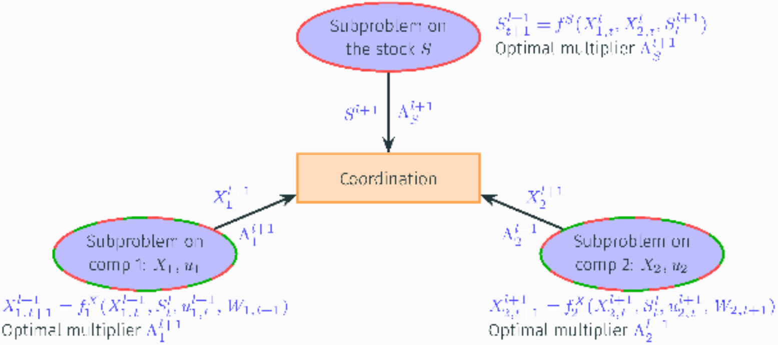
Use a decomposition by interaction prediction [MMT70] to iteratively find the best maintenance policy separately for each component and coordinate the components



Sketch of a decomposition method by component

General idea

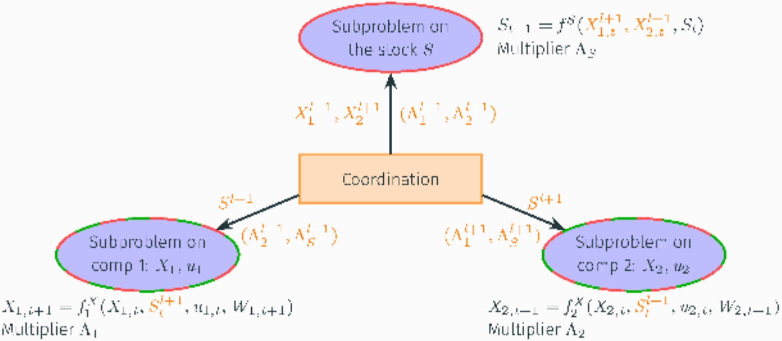
Use a decomposition by interaction prediction [MMT70] to iteratively find the best maintenance policy separately for each component and coordinate the components



Sketch of a decomposition method by component

General idea

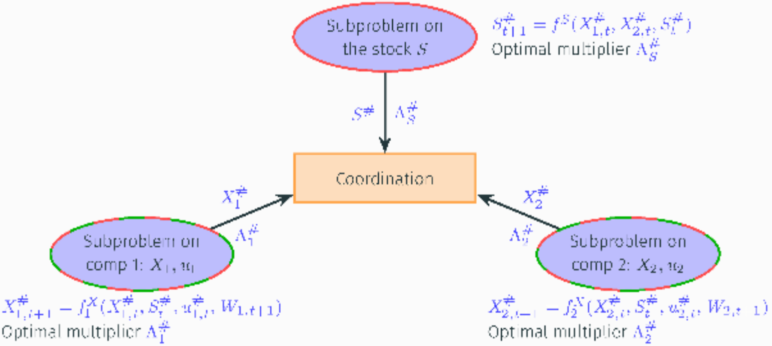
Use a decomposition by interaction prediction [MMT70] to iteratively find the best maintenance policy separately for each component and coordinate the components



Sketch of a decomposition method by component

General idea

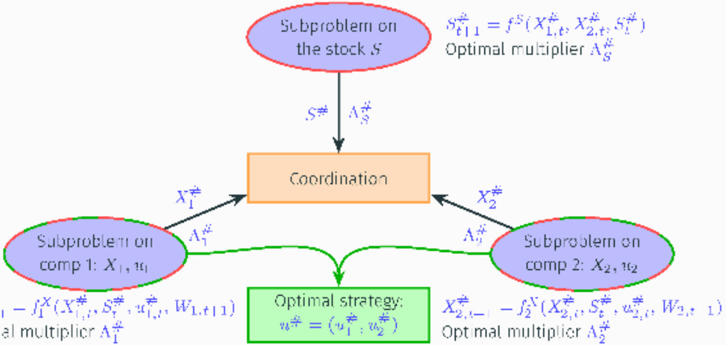
Use a decomposition by interaction prediction [MMT70] to iteratively find the best maintenance policy separately for each component and coordinate the components



Sketch of a decomposition method by component

General idea

Use a decomposition by interaction prediction [MMT70] to iteratively find the best maintenance policy separately for each component and coordinate the components



The auxiliary problem principle [Coh80] for a decomposition by component

- Choice of an auxiliary problem that can be decomposed in independent subproblems

- Subproblem on component i at iteration $l+1$:

$$\min_{(X_i,u_i)\in\mathcal{X}_i\times\mathcal{U}_i} \mathbb{E}\left(j_i(X_i,u_i)+j^{FO}(X_1^l,\dots,X_i,\dots,X_n^l)\right) + \text{ coordination terms}$$
$$\text{s.t. } X_{i,t+1}=f_i^X(X_{i,t},S_t^l,u_{i,t},W_{i,t+1}),\quad \forall t$$

- Subproblem on the stock S:

$$\min_{S\in\mathcal{S}} \text{coordination terms}$$
$$\text{s.t. } S_{t+1}=f^S(X_{1,t}^l,\dots,X_{n,t}^l,S_t),\quad \forall t$$

Fixed point algorithm for the decomposition by component

- Original problem: dimension nT
- Decomposition: n problems of dimension T per iteration

Algorithm 1 Fixed point algorithm

Start with (X^0, S^0, u^0) and Λ^0, set $l = 0$

At iteration $l+1$:

- For component $i = 1, \ldots, n$ do:
  - Solve
    $$\min_{(X_i, u_i) \in \mathcal{X}_i \times \mathcal{U}_i} \mathbb{E}\Big(j_i(X_i, u_i) + j^{FO}(X_1^l, \ldots, X_i, \ldots, X_n^l)\Big) + \text{coordination terms}$$
    $$\text{s.t. } X_{i,t+1} = f_i^X(X_{i,t}, S_t^l, u_{i,t}, W_{i,t+1}), \quad \forall t$$
    with any method (here with the blackbox optimization algorithm MADS [AD06]), solution (X_i^{l+1}, u_i^{l+1})
  - Compute an optimal multiplier Λ_i^{l+1} for the constraint using the adjoint state
- Similarly for the stock, solution S^{l+1} and optimal multiplier Λ_S^{l+1}

Stop if max number L of iterations reached, else $l \leftarrow l+1$ and start new iteration

---

Using a variational method in a discrete case

The fixed point algorithm is based on variational techniques:

- Gradient of the system dynamics appears in the coordination terms
- Gradient of the cost appears in the multiplier update step

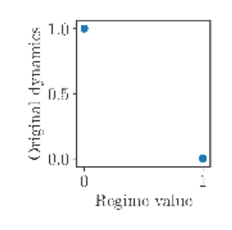
But the system is characterized by integer variables, they are relaxed:

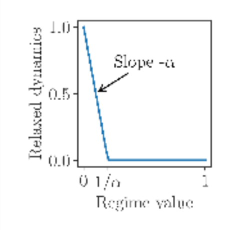
- Regime of the component: $E_{i,t} \in [0, 1]$
- Number of spare parts: $S_i \in \mathbb{R}_-$
- Controls: $u_{i,t} \in [0, 1]$

The dynamics is non-smooth, it is also relaxed:

- Relaxation controlled by a parameter α

Example for the assertion:
`If the component is broken`





Outline

Description of the industrial case

Parameter	Value		
Number of components n	80		
Initial number of spare parts S_0	16		
Horizon T	40 years		
Time of supply for the spare parts	2 years		
Discount factor	0.08		
Yearly forced outage cost	10000 k€/ year		
	Comp. 1	**Comp. 2**	**Comp. $i \geq 3$**
PM cost	50 k€	50 k€	50 k€
CM cost	100 k€	250 k€	200 k€
Failure distribution	$\mathcal{W}(2.3, 10)$	$\mathcal{W}(4, 20)$	$\mathcal{W}(3, 10)$
Mean time to failure	8.85 years	18.13 years	8.93 years

1 maintenance decision each year for each component

$\Rightarrow$ Problem in dimension $80 \times 40 = 3200$

Reference algorithm : MADS applied directly to the original optimization problem

Sample Average Approximation

Original problem:

$$\min_{(X,S,u)\in\mathcal{X}\times\mathcal{S}\times\mathbb{U}} \mathbb{E}\left(j(X,u)\right)$$
$$\text{s.t. } \Theta\left(X,S,u,W\right)=0$$

- $j(X,u)$ represents the overall maintenance and forced outage costs
- $\Theta(X,S,u,W)$ represents the dynamics of the system

Sample Average Approximation with Q Monte-Carlo scenarios $\omega_1,\dots,\omega_Q$:

$$\min_{(X,S,u)\in\mathcal{X}\times\mathcal{S}\times\mathbb{U}} \frac{1}{Q}\sum_{q=1}^{Q} j(X(\omega_q),u)$$
$$\text{s.t. } \Theta\left(X(\omega_q),S(\omega_q),u,W(\omega_q)\right)=0 \quad \forall q$$

Outline

Parameter tuning: procedure description

- Relaxation controlled at iteration l by a parameter α^l
- Update of the relaxation parameter at each iteration: $\alpha^{l+1} = \alpha^l + \Delta\alpha$
  As $\alpha \to \infty$, the relaxed dynamics converges to the real one
- Need to tune α^0 and $\Delta\alpha$
- Other parameters to tune: $\gamma^0, \Delta\gamma, r_X, r_S$ (no time for details in this talk)

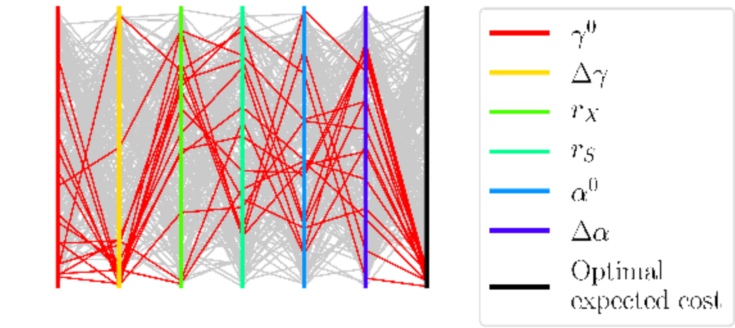
Tuning procedure for the vector of parameters $p = (\alpha^0, \Delta\alpha, \gamma^0, \Delta\gamma, r_X, r_S)$:

- Define bounds for the values of the parameters: $\alpha^0 \in [2, 200]$, $\Delta\alpha \in [0, 200]$, ...
- Draw 200 values of p with an optimized Latin Hypercube Sampling [DCI13]
- Optimization with each of the sampled values (i.e. 200 runs) on a smaller test case (10 components): computation time ~ 4h

Using sensitivity analysis to tune an optimization algorithm I

Qualitative approach: Cobweb plots

→ Visualize the best combinations of parameters for the optimization



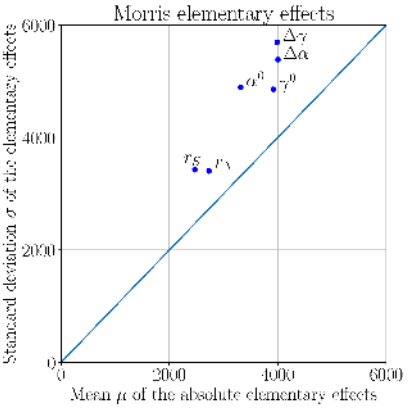
Conclusion

No clear result, except for $\Delta\gamma$ and r_X.

Using sensitivity analysis to tune an optimization algorithm II

Quantitative approach: the Morris method [Mor91].

→ Screening method: sensitivity of the optimization quantified by elementary effects.



- Mean of the elementary effects μ:

  → Quantifies the influence of a parameter on the result of the optimization

- Standard deviation σ of the elementary effects:

  → Measures the non-linear effects and the interactions between parameters on the result of the optimization

Conclusion

No screening possible, all inputs are influential with non linear/interaction effects.

Parameter tuning: conclusion

Tuning procedure for the vector of parameters $p = (\alpha^0, \Delta\alpha, \gamma^0, \Delta\gamma, r_X, r_S)$:

- Define bounds for the values of the parameters: $\alpha^0 \in [2, 200]$, $\Delta\alpha \in [0, 200]$, ...
- Draw 200 values of p with an optimized Latin Hypercube Sampling
- Optimization with each of the sampled values (i.e. 200 runs) on a smaller test case (10 components): computation time ~ 4h

Final choice

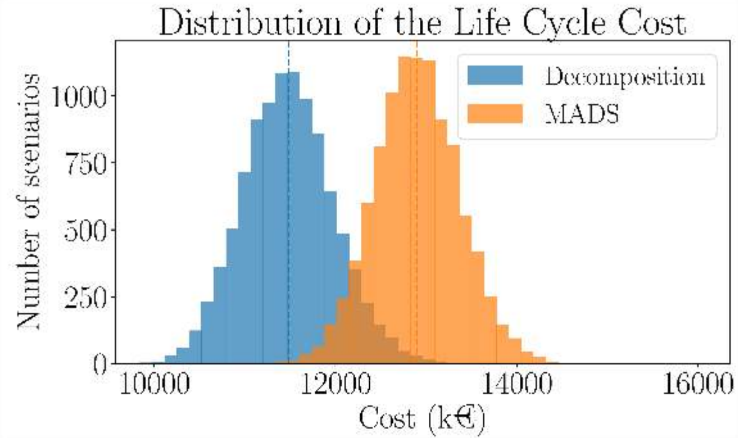
We simply take the best parameter p for the 80-component case.

Outline

Comparison of the Life Cycle Cost

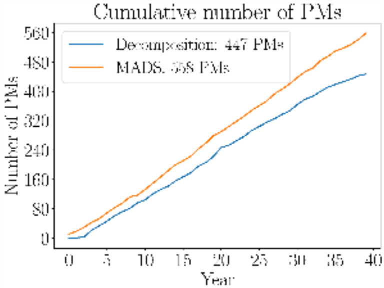
	Only CMs	MADS	Decomposition
Expected cost (k€)	46316	12820	11290

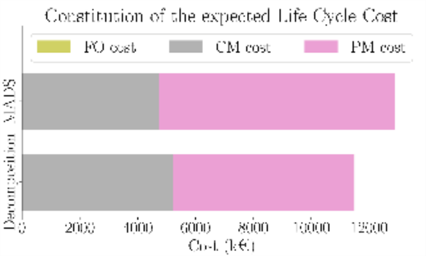
Gap MADS / Decomposition: 12%



Analysis of the maintenance strategies

	Decomposition	MADS
Mean number of PMs/component	5.6	7.0
Mean time between PMs	6.1 years	5.0 years
Mean number of failures/component	1.31	1.13
Number of forced outages	52/10000	1/10000





Outline

Conclusion

Summary:

- Formulation of the maintenance scheduling problem as a stochastic optimal control program
- To our knowledge, first time that a decomposition scheme based on the Auxiliary Problem Principle is applied to maintenance optimization
- Relaxation of the system needed to use the fixed-point algorithm
- Successful application on an industrial case with 80 components

[BCCL20-1] T. Bittar, P. Carpentier, J-Ph. Chancelier, and J. Lonchampt.
A Decomposition Method by Interaction Prediction for the Optimization of Maintenance Scheduling.
Submitted to Annals of Operations Research, 2020.

Perspectives:

- Complexification of the model: add a control for the stock management strategy, consider degraded states for the component
- Could we apply the decomposition methodology in a robust optimization framework?

Related work

In this talk: Sample Average Approximation

- Use a fixed set of scenarios to approximate the expectation
- Algorithm based on the deterministic Auxiliary Problem Principle (APP)

Other possibility: Stochastic Approximation

- Use one different scenario at each iteration of the algorithm
- Algorithm based on the stochastic APP

Theoretical contributions to the stochastic APP:

- Measurability of the iterates of the algorithm in a Banach space
- Extension of convergence results to the Banach case
- Derivation of efficiency estimates

[BCCL20-2] T. Bittar, P. Carpentier, J-Ph. Chancelier, and J. Lonchampt.
The stochastic Auxiliary Problem Principle in Banach spaces: measurability and convergence.
To be submitted soon.

[AD06] C. Audet and J. E. Dennis, Jr.
Mesh Adaptive Direct Search Algorithms for Constrained Optimization.
SIAM Journal on Optimization, 17(1):188–217, 2006.

[Coh80] G. Cohen.
Auxiliary problem principle and decomposition of optimization problems.
Journal of Optimization Theory and Applications, 32(3):277–305, 1980.

[DCI13] G. Damblin, M. Couplet, and B. Iooss.
Numerical studies of space-filling designs: optimization of Latin Hypercube Samples and subprojection properties.
Journal of Simulation, 7(4):276–289, 2013.

[MMT70] M. D. Mesarović, D. Macko, and Y. Takahara.
Theory of Hierarchical, Multilevel Systems, volume 68 of Mathematics in Science and Engineering.
Academic Press, 1970.

[Mor91] M. D. Morris.
Factorial Sampling Plans for Preliminary Computational Experiments.
Technometrics, 33(2):161–174, 1991.

Contributions and communications summary

Contributions:

[BCCL20-1] T. Bittar, P. Carpentier, J-Ph. Chancelier, and J. Lonchampt.
A Decomposition Method by Interaction Prediction for the Optimization of Maintenance Scheduling.
Submitted to Annals of Operations Research, 2020.

[BCCL20-2] T. Bittar, P. Carpentier, J-Ph. Chancelier, and J. Lonchampt.
The stochastic Auxiliary Problem Principle in Banach spaces: measurability and convergence.
To be submitted soon.

[BCCL20-3] T. Bittar, P. Carpentier, J-Ph. Chancelier, and J. Lonchampt.
Blackbox approaches for optimal maintenance scheduling.
Within the thesis.

Conference talks:

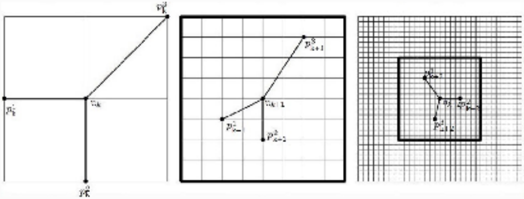
- Journées SMAI-MODE, online (09/2020)
- PGMO Days, Saclay (12/2019)
- International Conference in Stochastic Programming XV, Trondheim (07/2019)
- Optimization Days 2019, HEC Montréal (05/2019)

Thank you for your attention!

The blackbox algorithm MADS [AD06]

At iteration k, for the minimization of a cost function J:

- Current iterate u_k, mesh M_k
- Global search (exploration): Flexible step, possible use of heuristics and user-defined strategies to choose evaluation points $u_k^1, \cdots, u_k^p$ on M_k
- Local search (exploitation): Evaluation points $u_k^{p+1}, \cdots, u_k^{p-l}$ chosen in a neighbourhood $P_k \subset M_k$ of the best incumbent point u_k
- Mesh update:
  - If there exists i such that $J(u_k^i) < J(u_k)$ then $u_{k+1} = u_k^i$ and increase the mesh parameter
  - Else $u_{k+1} = u_k$ and decrease the mesh parameter



Morris method cheat sheet

Denote by $p = (p_1, \ldots, p_l)$ the vector of parameters

- n randomized one-at-a-time experiments
- Elementary effect while perturbating p_i in experiment j:

$$d_i^{(j)}(p^{(j)}) = \frac{\mathcal{A}(p^{(j)} + \delta e_i) - \mathcal{A}(p^{(j)})}{\delta}$$

with $p^{(j)}$ the value of the vector of parameters in the j-th experiment, $\mathcal{A}$ the model output (the optimization output in our case) and e_i the i-th vector of the canonical basis of $\mathbb{R}^l$.

We define two indices for each parameter p_i:

- Mean index:

$$\mu_i = \mathbb{E}\left(|d_i^{(j)}|\right) \simeq \frac{1}{n}\sum_{j=1}^{n}|d_i^{(j)}|$$

- Standard deviation index:

$$\sigma_i = \sqrt{\mathrm{Var}\left(d_i^{(j)}\right)} \simeq \sqrt{\frac{1}{n}\sum_{j=1}^{n}\left(d_i^{(j)} - \frac{1}{n}\sum_{j=1}^{n} d_i^{(j)}\right)^2}$$