Prognostics-based Scheduling to Extend a Platform Useful Life under Service Constraint

Stéphane Chrétien*, Nathalie Herr**, Jean-Marc Nicod** and Christophe Varnier**

** Laboratoire de Mathématiques de Besançon (LMB) – UFC*
*** FEMTO-ST Institute – Besançon – France*

July 2nd, 2014

1. Context and State of the art

Production scheduling

- Heterogeneous, independant, parallel machines
- Production based on a customer demand

1. Context and State of the art

Production scheduling

- Heterogeneous, independant, parallel machines
- Production based on a customer demand

Maintenance

- Wear and tear on machines
- Only one global maintenance allowed

⟹ Production horizon maximization before maintenance

1. Context and State of the art

Production scheduling

- Heterogeneous, independant, parallel machines
- Production based on a customer demand

Maintenance

⟹ Production horizon maximization

Operating conditions

⟹ Consideration of many running profiles

1. Context and State of the art

Production scheduling

- Heterogeneous, independant, parallel machines
- Production based on a customer demand

Maintenance

$\Longrightarrow$ Production horizon maximization

Operating conditions

$\Longrightarrow$ Consideration of many running profiles

$\Longrightarrow$ Taking real wear and tear into consideration (and not average life)

Prognostics and Health Management (PHM)

- Machine monitoring
- Remaining Useful Life (*RUL*) value depending on past and future usage

1. Context and State of the art

Prognostics and Health Management (PHM)

- Maintenance scheduling based on actual health state

- Haddad et al.: maintenance optimization under availability requirement

  [“A real options optimization model to meet availability requirements for offshore wind turbines”, MFPT, Virginia, 2011]

- Vieira et al.: maintenance scheduling based on health limits

  [“New variable health threshold based on the life observed for improving the scheduled maintenance of a wind turbine”, 2nd IFAC Workshop on Advanced Maintenance Engineering, 2012]

- Maintenance scheduling and mission reconfiguration

- Balaban et al.: rover maintenance optimization and mission duration extension

  [“A mobile robot testbed for prognostic-enabled autonomous decision making”, Annual Conference of the Prognostics and Health Management Society, 2011]

1. Context and State of the art

Operating conditions

- Variable-speed machines: control of time used by jobs on machines
  - Trick: single and multiple machine variable-speed scheduling

    ["Scheduling multiple variable-speed machines", Operations Research, 1994, 42, p.234-248]
  - Dietl et al.: derating of cutting tools by reducing the cutting speed

    ["An operating strategy for high-availability multi-station transfer lines", Int. J. of Automation and Computing, 2006, 2, p.125 - 130]

- Voltage/Frequency scaling
  - Kimura et al.: energy consumption reducing without impacting performance

    ["Empirical study on reducing energy of parallel programs using slack reclamation by dvfs in a power-scalable high performance cluster", IEEE Int. Conf. on Cluster Computing, Barcelona, 2006]
  - Semeraro et al.: microprocessor's performance and energy efficiency maximization

    ["Energy-efficient processor design using multiple clock domains with dynamic voltage and frequency scaling", HPCA, Cambridge, 2002]

1. Context and State of the art

Production scheduling

- Heterogeneous, independant, parallel machines
- Production based on a customer demand

Maintenance

$\Longrightarrow$ Production horizon maximization

Operating conditions

$\Longrightarrow$ Consideration of many running profiles

$\Longrightarrow$ Taking real wear and tear into consideration (and not average life)

Prognostics and Health Management (PHM)

$\Longrightarrow$ Use of prognostics results: *RUL*

$\Longrightarrow$ **Prognostics-based scheduling**

Outline

1. State of the art
2. Scheduling with running profiles in a discrete throughput domain
   - Problem statement
   - Complexity results
   - Optimal approach
   - Sub-optimal resolution
   - Results
   - Summary
3. Scheduling with running profiles in a continuous throughput domain
   - Problem statement
   - Convex optimization
   - Summary
4. Conclusion

2.1. Problem statement

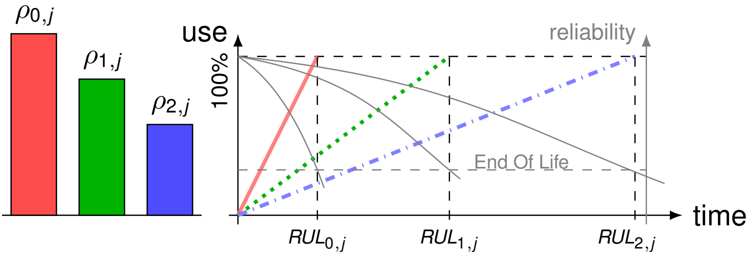
Problem data

- m independant machines (M_j)
- n running profiles (N_i)
- PHM monitoring $\rightarrow (\rho_{i,j}, RUL_{i,j})$

2.1. Problem statement

Problem data

- m independant machines (M_j)
- n running profiles (N_i)
- PHM monitoring $\rightarrow (\rho_{i,j}, RUL_{i,j})$



2.1. Problem statement

Problem data

- m independant machines (M_j)
- n running profiles (N_i)
- PHM monitoring $\rightarrow$ ($\rho_{i,j}$, $RUL_{i,j}$)

Constraints

- No *RUL* overrun
- Mission fulfillment: constant demand in terms of throughput (σ)

Objective

- To fulfill total throughput requirements as long as possible
  MAXK(σ | $\rho_{i,j}$ | $RUL_{i,j}$)

- Time discretization ($T = K \times \Delta T, 1 \leq k \leq K$)

2.1. Problem statement

Problem data

- m independant machines (M_j)
- n running profiles (N_i)
- PHM monitoring $\rightarrow (\rho_{i,j}, RUL_{i,j})$

Constraints

- No *RUL* overrun
- Mission fulfillment: constant demand in terms of throughput (σ)

Objective

- To fulfill total throughput requirements as long as possible
  MAXK($\sigma \mid \rho_{i,j} \mid RUL_{i,j}$)

- Time discretization ($\mathcal{T} = K \times \Delta T$, $1 \leq k \leq K$)

2.1. Problem statement

Problem data

- m independant machines (M_j)
- n running profiles (N_i)
- PHM monitoring $\rightarrow (\rho_{i,j}, RUL_{i,j})$

Constraints

- No *RUL* overrun
- Mission fulfillment: constant demand in terms of throughput (σ)

Objective

- To fulfill total throughput requirements as long as possible
  MAXK($\sigma \mid \rho_{i,j} \mid RUL_{i,j}$)

- Time discretization ($\mathcal{T} = K \times \Delta T$, $1 \leq k \leq K$)

Motivating example

M_1

$N_{0,1} = (\rho_{0,1} = 450W, RUL_{0,1} = 1u.t.)$
$N_{1,1} = (\rho_{1,1} = 125W, RUL_{1,1} = 3u.t.)$

M_2

$N_{0,2} = (\rho_{0,2} = 350W, RUL_{0,2} = 1u.t.)$
$N_{1,2} = (\rho_{1,2} = 100W, RUL_{1,2} = 2u.t.)$

M_3

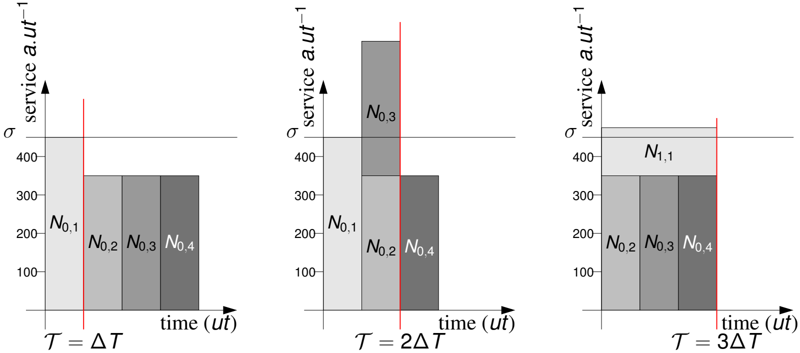
$N_{0,3} = (\rho_{0,3} = 350W, RUL_{0,3} = 1u.t.)$
$N_{1,3} = (\rho_{1,3} = 100W, RUL_{1,3} = 2u.t.)$

M_4

$N_{0,4} = (\rho_{0,4} = 350W, RUL_{0,4} = 1u.t.)$
$N_{1,4} = (\rho_{1,4} = 100W, RUL_{1,4} = 2u.t.)$

2.1. Problem statement

Motivating example



2.2. Complexity results

Complexity map

	Homogeneous machines	Heterogeneous machines
1 running profile	$\rho_{i,j} = \rho$ $\textsc{MaxK}(\sigma \mid \rho \mid RUL_j)$ $\Longrightarrow$ polynomial	$\rho_{i,j} = \rho_j$ $\textsc{MaxK}(\sigma \mid \rho_j \mid RUL_j)$ $\Longrightarrow$ NP-complete
n running profiles	$\rho_{i,j} = \rho_i$ $\textsc{MaxK}(\sigma \mid \rho_i \mid RUL_{i,j})$ $\Longrightarrow$?	$\rho_{i,j} = \rho_{i,j}$ $\textsc{MaxK}(\sigma \mid \rho_{i,j} \mid RUL_{i,j})$ $\Longrightarrow$ NP-complete

2.3. Optimal approach

Binary Integer Linear Program (BILP)

$a_{i,j,k} = $ 1 if machine M_j is used with running profile N_i during period k,
0 otherwise

$$\begin{cases} \forall k,\ \forall j,\ \sum_{i=0}^{n-1} a_{i,j,k} \leq 1 & \text{(machines)} \\ \forall j,\ \sum_{i=0}^{n-1} \frac{\sum_{k=1}^{K} a_{i,j,k} \times \Delta T}{RUL_{i,j}} \leq 1 & (RUL) \\ \forall k,\ \sum_{j=1}^{m} \sum_{i=0}^{n-1} a_{i,j,k} \times \rho_{i,j} \geq \sigma & \text{(service)} \end{cases}$$

- Binary search to find maximal value of k

$\Longrightarrow$ Limited to small instances: $\approx$ 5 machines, 2 running profiles, 20 time periods

2.3. Optimal approach

Binary Integer Linear Program (BILP)

$a_{i,j,k} = 1$ if machine M_j is used with running profile N_i during period k,
0 otherwise

$$\begin{cases} \forall k,\ \forall j,\ \sum_{i=0}^{n-1} a_{i,j,k} \leq 1 & \text{(machines)} \\ \forall j,\ \sum_{i=0}^{n-1} \frac{\sum_{k=1}^{K} a_{i,j,k} \times \Delta T}{RUL_{i,j}} \leq 1 & (RUL) \\ \forall k,\ \sum_{j=1}^{m} \sum_{i=0}^{n-1} a_{i,j,k} \times \rho_{i,j} \geq \sigma & \text{(service)} \end{cases}$$

- Binary search to find maximal value of k

$\Longrightarrow$ Limited to small instances: $\approx$ 5 machines, 2 running profiles, 20 time periods

2.4. Sub-optimal resolution

Basic heuristics

- Assignment of machines to reach the demand σ as long as possible
- Selection of one running profile for each machine and each time period

- H–LRF: Largest *RUL* First
- H–HOF: Highest Output First
- H–DP: Dynamic Programming based

Enhancement: repair

- Revision of the schedules obtained with basic heuristics
- Use of available machines

- H–LRF-R, H–HOF-R, H–DP-R

2.4. Sub-optimal resolution

Basic heuristics

- Assignment of machines to reach the demand σ as long as possible
- Selection of one running profile for each machine and each time period

- H–LRF: Largest *RUL* First
- H–HOF: Highest Output First
- H–DP: Dynamic Programming based

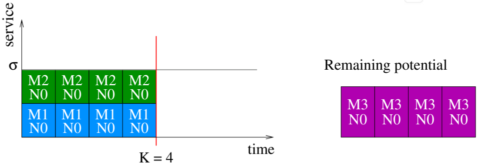
Enhancement: repair

- Revision of the schedules obtained with basic heuristics
- Use of available machines

- H–LRF-R, H–HOF-R, H–DP-R

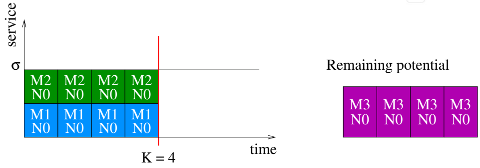
2.4. Sub-optimal resolution

- H–DP schedule

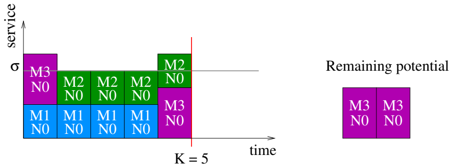


2.4. Sub-optimal resolution

- H–DP schedule

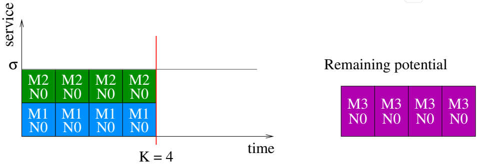


- H–DP-R Step 1

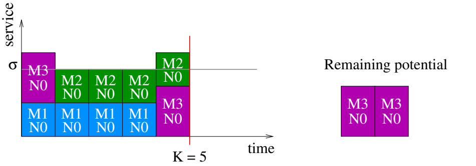


2.4. Sub-optimal resolution

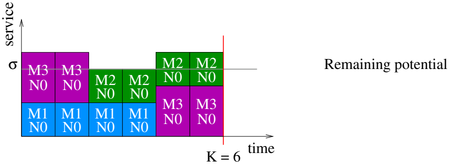
- H–DP schedule



- H–DP-R Step 1



- H–DP-R Step 2



2.4. Sub-optimal resolution

Basic heuristics

- Assignment of machines to reach the demand σ as long as possible
- Selection of one running profile for each machine and each time period

⌑ H–LRF: Largest *RUL* First

⌑ H–HOF: Highest Output First

⌑ H–DP: Dynamic Programming based

Enhancement: repair

- Revision of the schedules obtained with basic heuristics
- Use of available machines

⌑ H–LRF-R, H–HOF-R, H–DP-R

2.5. Results

Simulations

- Validation of heuristics on random problem instances
- Consideration of an increasing output $Q_{i,j} = \rho_{i,j} \times RUL_{i,j}$ with ρ such that:

$$Q_{0,j} > Q_{1,j} > \ldots > Q_{n-1,j}$$

$$\text{with} \;\; \rho_{0,j} > \rho_{1,j} > \ldots > \rho_{n-1,j}$$

$$\text{and} \;\; RUL_{0,j} < RUL_{1,j} < \ldots < RUL_{n-1,j}$$

- Constant demand $\sigma_k = \sigma$, with:

$$\sigma = \alpha \times \sum_{1 \leq j \leq m} \rho max_j$$

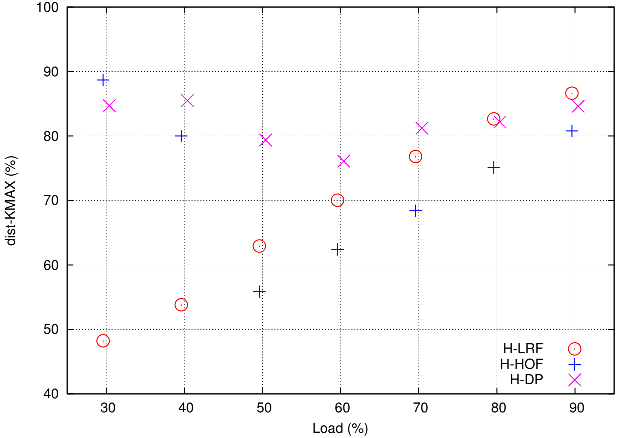
$$\text{with} \;\; 30\% \leq \sigma \leq 90\%$$

- Average value of 20 instances with same parameters values

2.5. Results

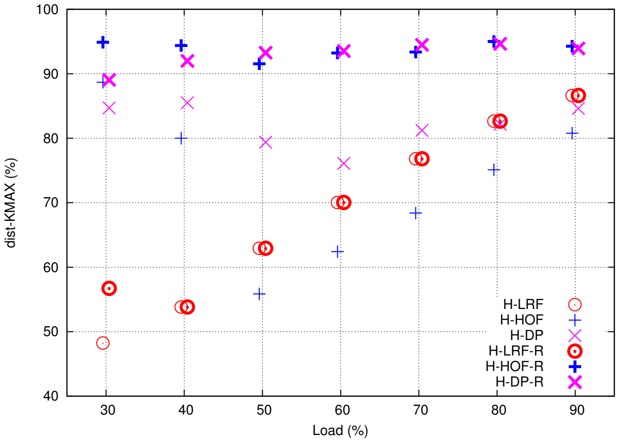
Comparison to an upper bound(n=5, m=25)

$$\text{KMAX} = \left\lfloor \sum_j \max_i \left(\rho_{i,j} \times RUL_{i,j}\right) / \sigma \right\rfloor$$



Comparison to an upper bound(n=5, m=25)

$$\text{KMAX} = \left\lfloor \sum_j \max_i \left(\rho_{i,j} \times RUL_{i,j}\right) / \sigma \right\rfloor$$



2.6. Summary

Adressed problem: maximizing the production horizon under service constraint

- Scheduling using prognostics results (*RUL*)
- Choose of running profiles in a discrete throughput domain
- Extension of a platform operational time
- Efficient sub-optimal heuristics (6% from upper bound)

[MIM2013], [PHM2014*], [CASE2014]

⟹ Application on wind turbines, cutting tools

Second adressed problem

- Choose of running profiles in a continuous throughput domain

⟹ Application on fuel cells

- Continuous use of machines

* Best Paper Award

2.6. Summary

Adressed problem: maximizing the production horizon under service constraint

- Scheduling using prognostics results (*RUL*)
- Choose of running profiles in a discrete throughput domain
- Extension of a platform operational time
- Efficient sub-optimal heuristics (6% from upper bound)

[MIM2013], [PHM2014*], [CASE2014]

⟹ Application on wind turbines, cutting tools

Second adressed problem

- Choose of running profiles in a continuous throughput domain

⟹ Application on fuel cells

- Continuous use of machines

* Best Paper Award

Outline

1. State of the art
2. Scheduling with running profiles in a discrete throughput domain
   - Problem statement
   - Complexity results
   - Optimal approach
   - Sub-optimal resolution
   - Results
   - Summary
3. Scheduling with running profiles in a continuous throughput domain
   - Problem statement
   - Convex optimization
   - Summary
4. Conclusion

3.1. Problem statement

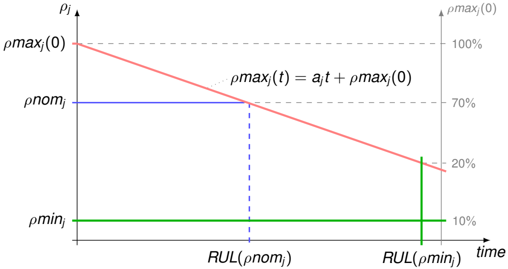
Problem data

- m independant machines (M_j)
- Running profiles **in a continuous throughput domain** ($\rho min_j \leq \rho_j(t) \leq \rho max_j(t)$)
- PHM monitoring $\rightarrow (\rho_j(t), RUL_j(\rho_j(t), t))$
- **Constant minimal throughput** (ρmin_j)
- **Maximal throughput decreasing with time** ($\rho max_j(t)$)

3.1. Problem statement

Problem data

- m independant machines (M_j)
- Running profiles **in a continuous throughput domain** ($\rho min_j \leq \rho_j(t) \leq \rho max_j(t)$)
- PHM monitoring $\rightarrow (\rho_j(t), RUL_j(\rho_j(t), t))$
- **Constant minimal throughput** (ρmin_j)
- **Maximal throughput decreasing with time** ($\rho max_j(t)$)



3.1. Problem statement

Problem data

- m independant machines (M_j)
- Running profiles **in a continuous throughput domain** ($\rho min_j \leq \rho_j(t) \leq \rho max_j(t)$)
- PHM monitoring $\rightarrow (\rho_j(t), RUL_j(\rho_j(t), t))$
- **Constant minimal throughput** (ρmin_j)
- **Maximal throughput decreasing with time** ($\rho max_j(t)$)

Constraints

- No *RUL* overrun
- Mission fulfillment: constant demand in terms of throughput (σ)
- **Avoid temporary machine shutdowns**

Objective

- To fulfill total throughput requirements as long as possible MAXK($\sigma \mid \rho_j(t) \mid RUL_j(\rho_j(t), t)$)

3.1. Problem statement

Problem data

- m independant machines (M_j)
- Running profiles **in a continuous throughput domain** ($\rho min_j \leq \rho_j(t) \leq \rho max_j(t)$)
- PHM monitoring $\rightarrow (\rho_j(t), RUL_j(\rho_j(t), t))$
- **Constant minimal throughput** (ρmin_j)
- **Maximal throughput decreasing with time** ($\rho max_j(t)$)

Constraints

- No *RUL* overrun
- Mission fulfillment: constant demand in terms of throughput (σ)
- **Avoid temporary machine shutdowns**

Objective

- To fulfill total throughput requirements as long as possible MAXK($\sigma \mid \rho_j(t) \mid RUL_j(\rho_j(t), t)$)

3.2. Convex optimization – Model

Problem statement

- Machine throughput

- Solution: schedule

Model

$\mathbf{f_j(t)} = \rho_\mathbf{j}(\mathbf{t})$ if the machine is used during the period t,

$= \mathbf{0}$ otherwise

$\forall j = 1, \ldots, m$ and $\forall t = 0, \ldots, T$

$$F = \begin{bmatrix} f_1(0) \\ f_2(0) \\ \vdots \\ f_m(0) \\ \vdots \\ f_1(T) \\ \vdots \\ f_m(T) \end{bmatrix}$$

3.2. Convex optimization – Model

Problem statement

- Machine throughput

- Solution: schedule

Model

$\mathbf{f_j(t)} = \rho_\mathbf{j}(\mathbf{t})$ if the machine is used during the period t,

$= \mathbf{0}$ otherwise

$\forall j = 1, \ldots, m$ and $\forall t = 0, \ldots, T$

$$F = \begin{bmatrix} f_1(0) \\ f_2(0) \\ \vdots \\ f_m(0) \\ \vdots \\ f_1(T) \\ \vdots \\ f_m(T) \end{bmatrix}$$

3.2. Convex optimization – Model

Problem statement

- Continuous running profiles
- Constant minimal throughput
- Maximal throughput decreasing with time
- No *RUL* overrun
- Mission fulfillment

Model

$$\mathbf{f_j(t) = f_{1,j}(t) + f_{2,j}(t)}$$

$\forall j = 1, \dots, m$ and $\forall t = 0, \dots, T$

$$\mathbf{f_j(t) \geq fmin_j}$$

$\forall j = 1, \dots, m$ and $\forall t = 0, \dots, T$

$$\mathbf{f_j(t) \leq fmax_j(t)}$$

$\forall j = 1, \dots, m$ and $\forall t = 0, \dots, T$

$$\sum_{\mathbf{t=0}}^{\mathbf{T}} \Gamma(\mathbf{f_j(t)}) \leq \mathbf{1}$$

$\forall j = 1, \dots, m$

$$\sum_{\mathbf{j=1}}^{\mathbf{m}} \mathbf{f_j(t)} \geq \sigma(\mathbf{t})$$

$\forall j = 1, \dots, m$ and $\forall t = 0, \dots, T$

3.2. Convex optimization – Constraints

Model

$$f_j(t) = f_{1,j}(t) + f_{2,j}(t)$$

$$f_j(t) \geq fmin_j$$

$$\forall j = 1, \ldots, m \text{ and } \forall t = 0, \ldots, T$$

$$\mathbf{f_j(t) \leq fmax_j(t)}$$

$$\forall j = 1, \ldots, m \text{ and } \forall t = 0, \ldots, T$$

$$\sum_{\mathbf{t=0}}^{\mathbf{T}} \Gamma(\mathbf{f_j(t)}) \leq \mathbf{1}$$

$$\forall j = 1, \ldots, m$$

$$\sum_{\mathbf{j=1}}^{\mathbf{m}} \mathbf{f_j(t)} \geq \sigma(\mathbf{t})$$

$$\forall j = 1, \ldots, m \text{ and } \forall t = 0, \ldots, T$$

Constraint functions

$$\psi_{\mathbf{1,j}}(\mathbf{F})_{\mathbf{t}} = \mathbf{f_{1,j}(t)}$$

$$\psi_{\mathbf{2,j}}(\mathbf{F})_{\mathbf{t}} = \mathbf{f_{2,j}(t)}$$

$$\forall j = 1, \ldots, m$$

$$\psi_{\mathbf{3,j}}(\mathbf{F})_{\mathbf{t}} = \mathbf{fmax_j(t) - f_j(t)}$$

$$\forall j = 1, \ldots, m$$

$$\psi_{\mathbf{4,j}}(\mathbf{F})_{\mathbf{t}} = \mathbf{1} - \sum_{\mathbf{t=0}}^{\mathbf{T}} \Gamma(\mathbf{f_j(t)})$$

$$\forall j = 1, \ldots, m$$

with $\Gamma(f_j(t)) = \dfrac{a_j \Delta T}{f_j(t) - fmax_j(0)}$

$$\psi_{\mathbf{0}}(\mathbf{F})_{\mathbf{t}} = \sum_{\mathbf{j=1}}^{\mathbf{m}} \mathbf{f_j(t)} - \sigma(\mathbf{t})$$

3.2. Convex optimization – Scheme

Objective function

$$\phi(F) = \sum_{j=1}^{m} \lambda_{1,j} \|\Delta f_{1,j}\|_1 + \lambda_{2,j} \|\Delta f_{2,j}\|_\infty + \lambda_{2',j} \|\Delta^2 f_{2,j}\|_1$$

subject to the previous constraints ψ_0 and $\psi_{K,j}(F)\ \forall\, K = 1, \ldots, 4$

- ℓ_1 penalization approach (convex functions)
- Control of the number of jumps (f_1), of the slope and of the number of breakpoints (f_2)

3.2. Convex optimization – Scheme

Objective function

$$\phi(F) = \sum_{j=1}^{m} \lambda_{1,j} \|\Delta f_{1,j}\|_1 + \lambda_{2,j} \|\Delta f_{2,j}\|_\infty + \lambda_{2',j} \|\Delta^2 f_{2,j}\|_1$$

subject to the previous constraints ψ_0 and $\psi_{K,j}(F) \; \forall \, K = 1, \ldots, 4$

Bregman-proximal method

$\Longrightarrow$ Minimization of the objective function

$\Longrightarrow$ Assures the positivity for each constraint function:
$\psi_0(F) \geq 0$ and $\psi_{K,j}(F) \geq 0 \;\; \forall \, K = 1, \ldots, 4$ and $\; \forall \, j = 1, \ldots, m$

Lagrange function

$$L(F, u) = \|F\|_1 + \phi(F) + \sum_{j=1}^{m} u_j \psi_{4,j}(F)$$

such that $u \leq 0$, $\psi_0(F) \geq 0$ and $\psi_{K,j}(F) \geq 0 \;\; \forall \, K = 1, \ldots, 3$

3.2. Convex optimization – Scheme

Objective function

$$\phi(F) = \sum_{j=1}^{m} \lambda_{1,j} \|\Delta f_{1,j}\|_1 + \lambda_{2,j} \|\Delta f_{2,j}\|_\infty + \lambda_{2',j} \|\Delta^2 f_{2,j}\|_1$$

subject to the previous constraints ψ_0 and $\psi_{K,j}(F)\ \forall\ K = 1, \ldots, 4$

Bregman-proximal method

$\Longrightarrow$ Minimization of the objective function

$\Longrightarrow$ Assures the positivity for each constraint function:
$\psi_0(F) \geq 0$ and $\psi_{K,j}(F) \geq 0\ \ \forall\ K = 1, \ldots, 4$ and $\ \forall\ j = 1, \ldots, m$

Lagrange function

$$L(F, u) = \|F\|_1 + \phi(F) + \sum_{j=1}^{m} u_j \psi_{4,j}(F)$$

such that $u \leq 0$, $\psi_0(F) \geq 0$ and $\psi_{K,j}(F) \geq 0\ \ \forall\ K = 1, \ldots, 3$

3.2. Convex optimization

Coupled ADMM Bregman-proximal scheme

(ADMM: Alternating Direction Method of Multipliers)

Primal step:

$$F^{(l+1)} = \underset{F \in \mathbb{R}^{2m(T+1)}}{\operatorname{argmin}} \Big(L(F, u^{(l)}) + \lambda \big(D_h(\psi_0(F^{(l)}), \psi_0(F)) \big) + < V'^{(l)}, F - F' > + < V''^{(l)}, F - F'' > + < V'''^{(l)}, F - F''' > + \frac{\rho}{2} \|F - F'\|_F^2 + \frac{\rho}{2} \|F - F''\|_F^2 + \frac{\rho}{2} \|F - F'''\|_F^2 \Big)$$

$$F'^{(l+1)} = \underset{\substack{F' \in \mathbb{R}^{2m(T+1)} \\ F = F^{(l+1)}}}{\operatorname{argmin}} \left(\lambda \left(\sum_{j=1}^{m} D_h(\psi_{1,j}(F'^{(l)}), \psi_{1,j}(F')) \right) + < V'^{(l)}, F - F' > + \frac{\rho}{2} \|F - F'\|_F^2 \right)$$

$$F''^{(l+1)} = \underset{\substack{F'' \in \mathbb{R}^{2m(T+1)} \\ F = F^{(l+1)}}}{\operatorname{argmin}} \left(\lambda \left(\sum_{j=1}^{m} D_h(\psi_{2,j}(F''^{(l)}), \psi_{2,j}(F'')) \right) + < V''^{(l)}, F - F'' > + \frac{\rho}{2} \|F - F''\|_F^2 \right)$$

$$F'''^{(l+1)} = \underset{\substack{F''' \in \mathbb{R}^{2m(T+1)} \\ F = F^{(l+1)}}}{\operatorname{argmin}} \left(\lambda \left(\sum_{j=1}^{m} D_h(\psi_{3,j}(F'''^{(l)}), \psi_{3,j}(F''')) \right) + < V'''^{(l)}, F - F''' > + \frac{\rho}{2} \|F - F'''\|_F^2 \right)$$

3.2. Convex optimization

Coupled ADMM Bregman-proximal scheme

(ADMM: Alternating Direction Method of Multipliers)

Primal step: $F^{(l+1)}, F'^{(l+1)}, F''^{(l+1)}, F'''^{(l+1)}$

Dual step:

$$
\begin{aligned}
u^{(l+1)} &= \underset{u}{\text{argmax}} \left(L(F^{(l+1)}, u) + \lambda D_h(u^{(l)}, u) \right) \\
V'^{(l+1)} &= V'^{(l)} + F^{(l+1)} - F'^{(l+1)} \\
V''^{(l+1)} &= V''^{(l)} + F^{(l+1)} - F''^{(l+1)} \\
V'''^{(l+1)} &= V'''^{(l)} + F^{(l+1)} - F'''^{(l+1)}
\end{aligned}
$$

3.2. Convex optimization

Coupled ADMM Bregman-proximal scheme

(ADMM: Alternating Direction Method of Multipliers)

Primal step: $F^{(l+1)}, F'^{(l+1)}, F''^{(l+1)}, F'''^{(l+1)}$

Dual step:

$$
\begin{aligned}
u^{(l+1)} &= \underset{u}{\operatorname{argmax}} \left(L(F^{(l+1)}, u) + \lambda D_h(u^{(l)}, u) \right) \\
V'^{(l+1)} &= V'^{(l)} + F^{(l+1)} - F'^{(l+1)} \\
V''^{(l+1)} &= V''^{(l)} + F^{(l+1)} - F''^{(l+1)} \\
V'''^{(l+1)} &= V'''^{(l)} + F^{(l+1)} - F'''^{(l+1)}
\end{aligned}
$$

Work in progress...

3.3. Summary

Adressed problem: maximizing the production horizon under service constraint

- Scheduling using prognostics results (*RUL*)
- Choose of running profiles in a continuous throughput domain
- Extension of a platform operational time

⇒ Convergence results of the method...

⇒ Experiment results...

4. Conclusion

Adressed problem: maximizing the production horizon under service constraint

- Scheduling using prognostics results (*RUL*)
- Choose of running profiles in a discrete or a continuous domain
- Extension of a platform operational time
- Off-line scheduling
- Constant and variable demand

Future work

- Introduction of storage in the case of variable demand
- Hybrid power production including storage devices

4. Conclusion

Adressed problem: maximizing the production horizon under service constraint

- Scheduling using prognostics results (*RUL*)
- Choose of running profiles in a discrete or a continuous domain
- Extension of a platform operational time
- Off-line scheduling
- Constant and variable demand

Future work

- Introduction of storage in the case of variable demand
- Hybrid power production including storage devices

Thank you for your attention

Any questions ?