**Kovács, A.; Erdős, G.; Viharos, Zs. J.; Monostori, L.**: Szélparkok hibadetektálása és karbantartás ütemezése (Failure detection and maintenance scheduling for wind farms, in Hungarian), *BJMT Alkalmazott Matematikai Konferencia 2012*, Győr, 2012, presentation available



# Failure detection and maintenance scheduling for wind farms

by

A. Kovács[1], G. Erdős[1], Zs.J. Viharos [1], L. Monostori [1,2]

[1] Fraunhofer Project Center for Production Management and Informatics, Computer and Automation Research Institute, Hungary

[2] Dept. of Manufacturing Science and Technology, BME, Hungary

MTA SZTAKI

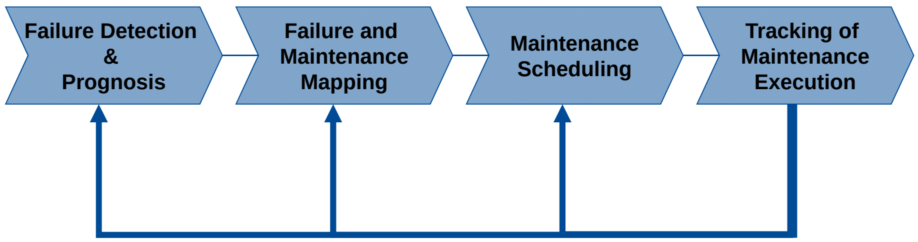
## Introduction

- Motivation
  - Wind is often regarded as “energy for free”
  - Not true: O&M contributes ~20% to the final price of energy
  - Investors’ expectations are high on turbine availability
  - Failures must be detected and fixed as quickly and efficiently as possible
- Objectives
  - Integrated system for monitoring, diagnosis and maintenance
  - The focus of the talk is on maintenance scheduling



MTA SZTAKI

## Workflow covered

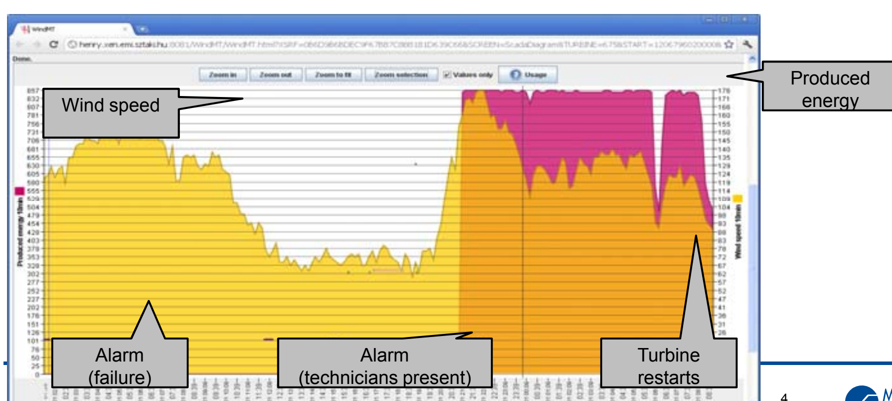





MTA SZTAKI

## Failure detection & prognosis

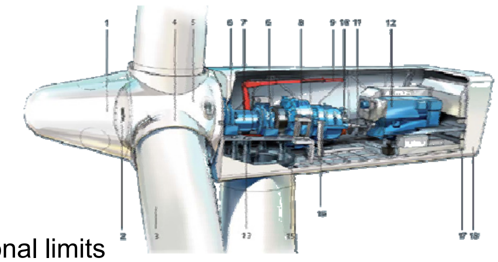
- Based on input from the turbine's SCADA system
  - Automated mapping of SCADA alarms & signals
  - Decision support to human experts by diagrams & reports






MTA SZTAKI

# Failure detection & prognosis

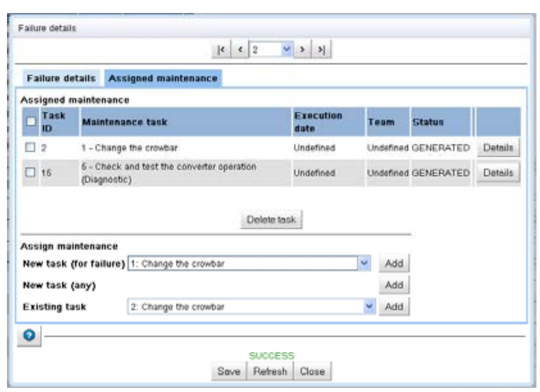


- Failure detection
  - Identify an existing failure
  - Measured signal falls out of operational limits
  - SCADA issues a warning, may stop the turbine
- Failure prognosis
  - Recognize failures before they actually occur
  - By physical measurements (e.g., vibration) or life data analysis
  - Current precision sufficient for controlling spare part logistics

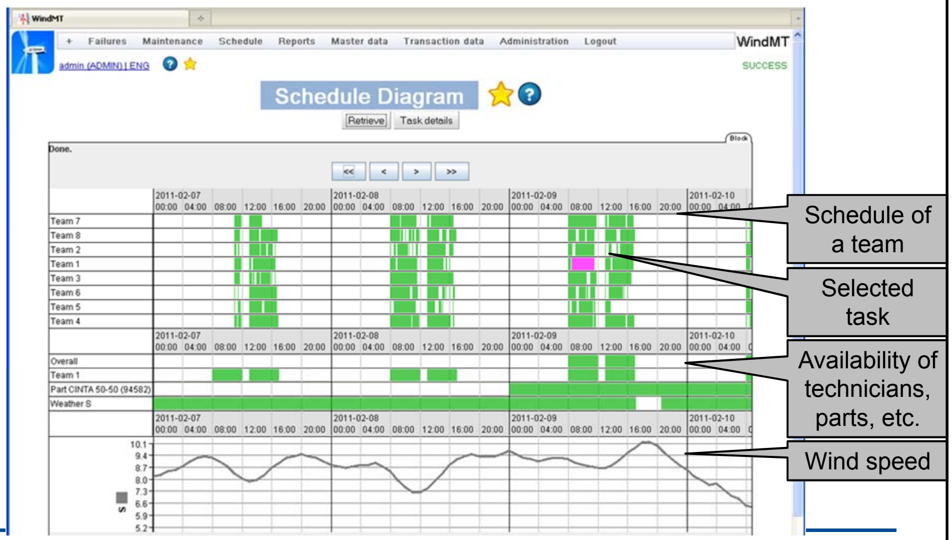


# Failure and maintenance mapping

- Selection of the best maintenance action for each failure
  - Recommendations from Failure Mode and Effects Analysis (FMEA)
  - Historical success rates
  - Interdependencies with other maintenance tasks
  - Cost analysis
- Software provides
  - Decision support to experts
  - Automated task generation
    - Weighted set covering problem



# Maintenance scheduling






# Maintenance execution

- Execution tracking by feedback from technicians
  - Actual failure mode
  - Failed component
  - Task execution, duration, spares, etc.
- Refinement of the applied models
  - Failure detection
  - Failure & maintenance mapping
  - Task definitions
  - Turbine design

## The maintenance scheduling problem

- The model covers all types of field maintenance
  - Corrective (detected failure)
  - Predictive (prognosed failure)
  - Preventive (planned maintenance)
  - Retrofitting, etc.
- One instance of the scheduling problem involves
  - One *zone*: set of farms maintained using common resources
  - Short-term horizon: 3-7 days
  - The set of tasks defined in the input

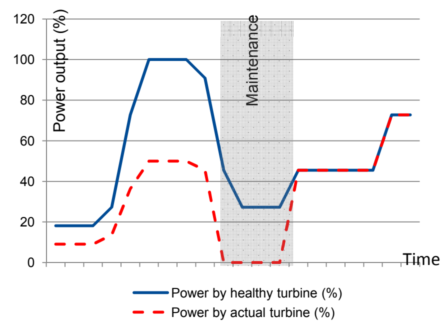


## Maintenance scheduling: tasks

- Tasks
  - Non-preemptive, with known deterministic duration
  - All requirements are assumed to be known a priori
- Requirements of the tasks
  - Skilled technicians
  - Spare parts
  - Hired services (cranes, special trucks)
  - Weather conditions
- Further constraints
  - Travel times between farms
  - Incompatibility of tasks, etc.

# Maintenance scheduling: objectives

- Production of turbine depends on
  - Wind speed (external)
  - Turbine condition (depends on maintenance schedule – internal)
- Production loss due to
  - Failures
  - Maintenance
- Optimization criterion
  - Minimizing total production loss
  - Irregular criterion (potentially worth postponing the tasks)






# Solution approach

- Combination of mixed-integer programming (MIP) and custom heuristics
  - MIP solves the core optimization problem
  - Heuristics reduce the complexity of the MIP and handle some special situations
- The MIP
  - Time-indexed formulation
  - Solved by a commercial MIP solver (ILOG Cplex 11.2)
  - Solved to optimality (unless time limit is hit)

## Solution approach: the MIP

Minimize

$$\sum_{j=1}^{J}\sum_{t=1}^{T} z_{j,t} + \sum_{i=1}^{N} \delta_i y_i \qquad (1)$$

Min. production loss

subject to

$$\sum_{k=1}^{K}\sum_{t=1}^{T} x_{i,k,t} + y_i = 1 \qquad \forall i \qquad (2)$$

Task executed in some way or postponed

$$x_{i,k,t} = 0 \qquad \forall i,j,k : \neg\Theta_{i,j,k} \qquad (3)$$

Infeasible assignments excluded

$$\sum_{i=1}^{N}\sum_{t'=t-p_i+1}^{t} x_{i,k,t'} \le 1 \qquad \forall k,t \qquad (4)$$

$$\sum_{i:\, s\in Z_i \wedge F(i)=f}\ \sum_{k=1}^{K}\ \sum_{t'=t-p_i+1}^{t} x_{i,k,t'} \le S_{s,f,t} \qquad \forall s,f,t \qquad (5)$$

Capacity constraints (technicians, services)



## Solution approach: the MIP (contd.)

$$z_{j,t} \ge \left(1 - \sum_{k=1}^{K}\sum_{t'=1}^{t-p_i} x_{i,k,t}\right) w^0_{i,j,t} \qquad \forall i,j,t \qquad (6)$$

$$z_{j,t} \ge \sum_{k=1}^{K}\sum_{t'=t-p_i+1}^{t} x_{i,k,t} w^1_{i,j,t} \qquad \forall i,t,j \qquad (7)$$

Production loss

$$x_{i,k,t} \le a_{k,f,t'} \qquad \forall i,k,t,t' : t \le t' < t + p_i \qquad (8)$$

$$a_{k,f,t} + \sum_{f':\, d_{f,f'} > t'-t} a_{k,f',t'} \le 1 \qquad \forall k,t,t',f : t' \ge t \qquad (9)$$

Technicians must travel

$$\sum_{k=1}^{K}\sum_{t'=t-p_i+1}^{t} x_{i,k,t'} + \sum_{k=1}^{K} x_{i',k,t} \le 1 \qquad \forall i,i' : V_{i,i'} \qquad (10)$$

Task compatibility

$$x_{i,k,t} \in \{0,1\} \qquad \forall i,k,t \qquad (11)$$

$$0 \le y_i, b_{k,t} \le 1, \quad 0 \le z_{j,t} \qquad \forall i,j,k,t \qquad (12)$$

Variables' domains

## Solution approach: the heuristics

- Heuristics for pre-processing
  - Decomposition according to nearby sets of farms
  - In massively oversubscribed problems, omit the least important tasks
  - Segmentation of extremely long tasks
- Heuristics for post-processing
  - Iterative improvement by re-partitioning and re-positioning the segments (necessary due to the segmented long tasks)



## Computational evaluation

- Evaluation of the MIP approach to the core problem
  - Heuristics switched off
  - Random instances generated based on real historical records
  - Up to 50 tasks at 7 wind farms
  - Time limit set to 10 minutes
- Results
  - 95% of the instances solved to proven optimality
  - Average optimality gap of ~0.01% for the remaining instances

| | F=3 | | | 5 | | | 7 | | |
|---|---|---|---|---|---|---|---|---|---|
| | Opt | Time *(s)* | Gap *(ppm)* | Opt | Time *(s)* | Gap *(ppm)* | Opt | Time *(s)* | Gap *(ppm)* |
| N=10 | 5 | 0.18 | - | 5 | 0.12 | - | 5 | 0.13 | - |
| 20 | 5 | 1.96 | - | 5 | 1.26 | - | 5 | 0.55 | - |
| 30 | 5 | 6.39 | - | 5 | 11.40 | - | 5 | 3.36 | - |
| 40 | 5 | 51.40 | - | 5 | 85.31 | - | 5 | 123.35 | - |
| 50 | 3 | 439.48 | 259 | 3 | 311.74 | 280 | 5 | 25.35 | - |

## Evaluation on real-life problems

- Experimental setup
  - 105 turbines at 6 farms, maintained by 12 teams
  - Scheduler was run every morning during 3 weeks
  - Larger problems than expected, with 200-250 tasks
- Results
  - Overwhelmingly positive feedback
  - Optimality gap around ~1%
  - Some additional features requested, e.g.,
    - Certain tasks require multiple teams
    - Preempted tasks must be continued by the same team



## Conclusions

- Integrated system for monitoring, diagnosis and maintenance of wind farms
  - Maps available engineering knowledge in failure detection
  - New models and algorithms for maintenance scheduling
  - Optimizes wind farm operation by minimizing production loss
  - Validated on real-life data
  - Potential industrial exploitation

Thank you for your attention!



MTA SZTAKI