Machine Learning for Probabilistic Power Systems Reliability Management

Louis Wehenkel

joint work with L. Duchesne and E. Karangelos

Lyngby : 23.11.2018

LIÈGE université

Montefiore Institute

Organisation of the talk

- PART I
  Probabilistic Reliability Management: Stakes and Sub-problems

- PART II
  Machine Learning for Reliability Assessment

- PART III
  Machine Learning for Reliability Control

Part I

Probabilistic Reliability Management: Stakes and Sub-problems

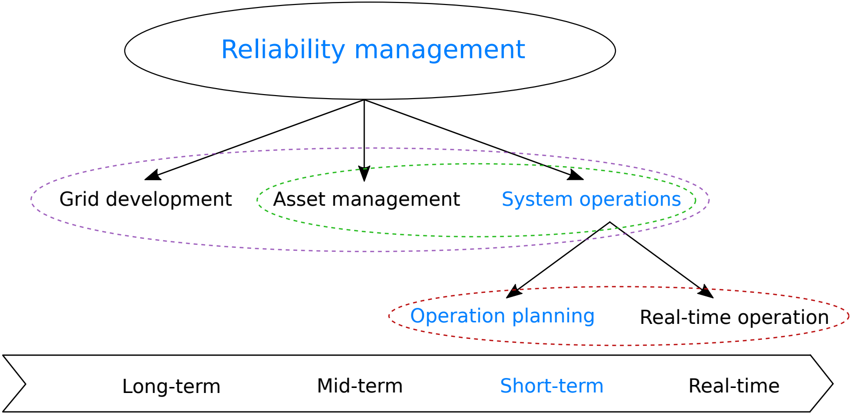
- Reliability Management
- Reliability Assessment vs Reliability Control

Electric power system reliability

- **Requirement:**
  - At sub-second temporal resolution, balance generation/storage/consumption, under network constraints, in spite of various threats
- **Threats faced:**
  - Variations of generation and demand, weather conditions
  - Component failures, human errors, adversarial attacks
- **Problems to avoid:**
  - Component overloads, voltage or frequency deviations
  - Cascading overloads, instabilities, blackouts
- **Opportunities:**
  - Optimisation and control of flows closer to real-time
  - Preventive maintenance and planning of operation
  - Adaptation of the grid structure to market needs

Reliability management contexts

Taking decisions in order to ensure the reliability of the system while minimizing socio-economic costs



The currently used N-1 Reliability Criterion (since 50 years)

"The power system should at any time be able to seamlessly withstand the spontaneous disconnection of any single component (*e.g.* line, transformer, *etc.*)."

- But N-1 can be over-conservative:
  e.g., limiting use of cheap renewables.
- ... can be under-conservative:
  e.g., adverse weather/major sport events, *etc.*.
- ... can be risk averse:
  seeking to avoid even "minor" (sometimes tolerable) consequences.
- and N-1 can be risk taking!
  incetivizes corrective control while neglecting its possible failure.

Need to move towards Probabilistic Reliability Criteria

"To enable the optimization of the overall expected socio-economic performance."

- New models need to be developed
- More complex decision making problems need to solved

Two $\neq$ types of reliability management sub-tasks

- **Reliability assessment (ex ante):**
  Determine the expected level of reliability for a given future period of time and for a certain geographical area
  $\rightarrow$ *large-scale stochastic simulation problem*

- **Reliability control:**
  Determine an optimal set of decisions to take in order to ensure a desired level of reliability over a given time period and for a certain geographical area
  $\rightarrow$ *large-scale multi-stage stochastic optimisation problem*

NB: Both tasks need a suitable physical model of the system and suitable uncertainty models of the exogenous factors acting on it over the considered time period and geographical area.

Part II

Machine Learning for Reliability Assessment

Reliability assessment:

Determine the expected level of reliability for a given future period of time and for a certain geographical area:

- Real-time mode (minutes)
- Short-term look-ahead mode (hours, days)
- Longer-term look-ahead problems (months, years)

Reliability Assessment in Real-Time Mode (Objectives)

Every 5 minutes, based on the real-time situation x_{rt}, assess risk induced by contingencies that could occur over the next hour.

- Remarks:
  - Real-time situation: defined by exogenous and endogenous info
  - Contingencies: big set of external and/or internal threats
  - Contingency response: PF, OPF, time-domain simulation...

Reliability Assessment in Real-Time Mode (Objectives)

Every 5 minutes, based on the real-time situation x_{rt}, assess risk induced by contingencies that could occur over the next hour.

- Based on data and models (we stress dependence on x_{rt}):
  - $C(x_{rt}), \pi_c(x_{rt}, c)$: set of contingencies and their probabilities
  - $f_{cr}(x_{rt}, c)$: measure of the severity of contingency c in state x_{rt}

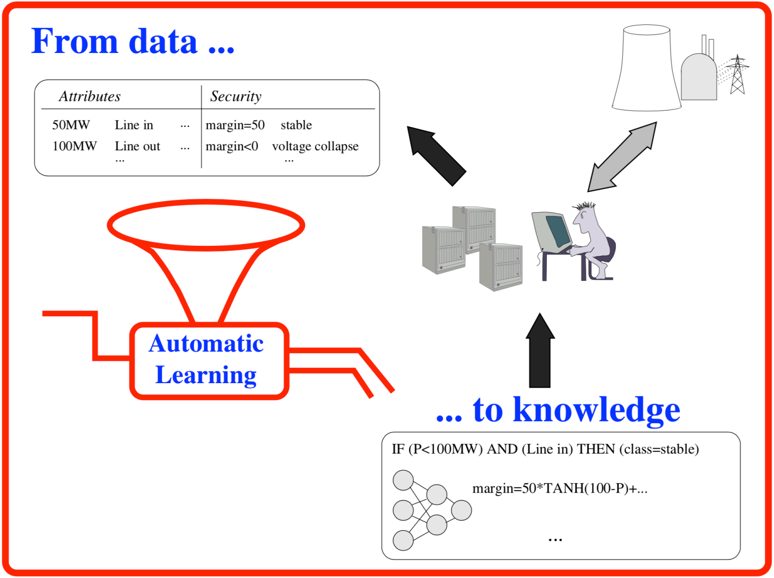
- Remarks:
  - Real-time situation: defined by exogenous and endogenous info
  - Contingencies: big set of external and/or internal threats
  - Contingency response: PF, OPF, time-domain simulation...

Reliability Assessment in Real-Time Mode (Objectives)

Every 5 minutes, based on the real-time situation x_{rt}, assess risk induced by contingencies that could occur over the next hour.

- Based on data and models (we stress dependence on x_{rt}):
  - $C(x_{rt}), \pi_c(x_{rt}, c)$: set of contingencies and their probabilities
  - $f_{cr}(x_{rt}, c)$: measure of the severity of contingency c in state x_{rt}
- We want to asses the expected impact of possible contingencies:
  - $\mathbb{E}\{f_{cr}|x_{rt}\} = \sum_{c \in C(x_{rt})} \pi_c(x_{rt}, c) f_{rt}(x_{rt}, c)$
    (e.g. expected cost of service interruptions)
  - $\mathbb{P}\{f_{cr} > \eta | x_{rt}\} = \sum_{c \in C(x_{rt})} \pi_c(x_{rt}, c) 1(f_{cr}(x_{rt}, c) > \eta)$
    (e.g. probability of large service interruptions)
- Remarks:
  - Real-time situation: defined by exogenous and endogenous info
  - Contingencies: big set of external and/or internal threats
  - Contingency response: PF, OPF, time-domain simulation...

Machine Learning for power systems (in general)



Opportunities for Machine Learning (in Real-Time mode)

- Practical facts:
  - The evaluation of the contingency response function $f_{cr}(x_{rt}, c)$ is generally expensive in CPU time.
  - Still, this function will be evaluated as often as possible by TSO, yielding growing datasets $D = \{(x^i_{rt}, c^i), f_{cr}(x^i_{rt}, c^i)\}_{i=1}^{\cdots}$

Opportunities for Machine Learning (in Real-Time mode)

- Practical facts:
  - The evaluation of the contingency response function $f_{cr}(x_{rt}, c)$ is generally expensive in CPU time.
  - Still, this function will be evaluated as often as possible by TSO, yielding growing datasets $D = \{(x^i_{rt}, c^i), f_{cr}(x^i_{rt}, c^i)\}_{i=1}^{\cdots}$
- Supervised Machine Learning Paradigm:
  - From a sample D of input-output pairs $\{(z^i, y^i)\}_{i=1}^{n}$, we can learn a function $h(\cdot)$ such that $|h(z) - y|$ is small on average.

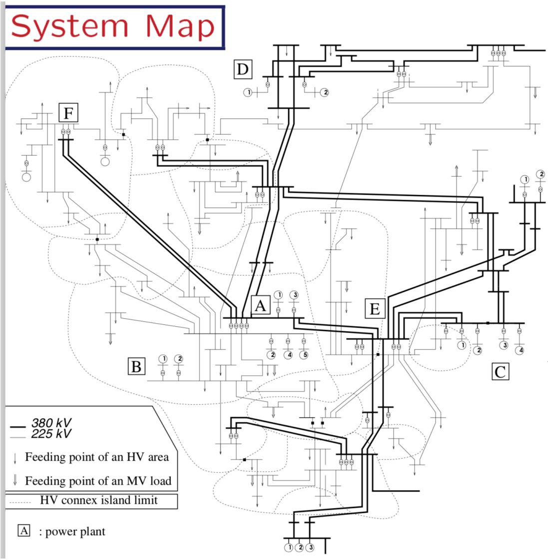
Opportunities for Machine Learning (in Real-Time mode)

- Practical facts:
  - The evaluation of the contingency response function $f_{cr}(x_{rt}, c)$ is generally expensive in CPU time.
  - Still, this function will be evaluated as often as possible by TSO, yielding growing datasets $D = \{(x_{rt}^i, c^i), f_{cr}(x_{rt}^i, c^i)\}_{i=1}^{\cdots}$
- Supervised Machine Learning Paradigm:
  - From a sample D of input-output pairs $\{(z^i, y^i)\}_{i=1}^n$, we can learn a function $h(\cdot)$ such that $|h(z) - y|$ is small on average.
- Application to Real-Time Reliability Assessment:
  - Learn a "regression proxy": $h_{\text{regr}}(x_{rt}, c) \approx f_{cr}(x_{rt}, c)$
  - Learn a "classifier proxy": $h_{\text{class}}(x_{rt}, c) \approx 1(f_{cr}(x_{rt}, c) \geq \eta)$

Opportunities for Machine Learning (in Real-Time mode)

- Practical facts:
  - The evaluation of the contingency response function $f_{cr}(x_{rt}, c)$ is generally expensive in CPU time.
  - Still, this function will be evaluated as often as possible by TSO, yielding growing datasets $D = \{(x_{rt}^i, c^i), f_{cr}(x_{rt}^i, c^i)\}_{i=1}^{\cdots}$
- Supervised Machine Learning Paradigm:
  - From a sample D of input-output pairs $\{(z^i, y^i)\}_{i=1}^n$, we can learn a function $h(\cdot)$ such that $|h(z) - y|$ is small on average.
- Application to Real-Time Reliability Assessment:
  - Learn a "regression proxy": $h_{\text{regr}}(x_{rt}, c) \approx f_{cr}(x_{rt}, c)$
  - Learn a "classifier proxy": $h_{\text{class}}(x_{rt}, c) \approx 1(f_{cr}(x_{rt}, c) \geq \eta)$
- The underlying assumptions are as follows:
  - h-proxies are much faster to evaluate than $f_{cr}(x_{rt}, c)$
  - It is possible to learn accurate enough h-proxies

Example: Voltage stability / French system (circa 1993)



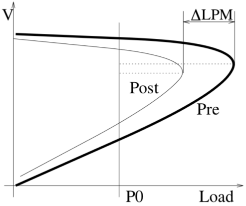
Study region load : 4700-7500MW

Topologies : N, N-1, N-2

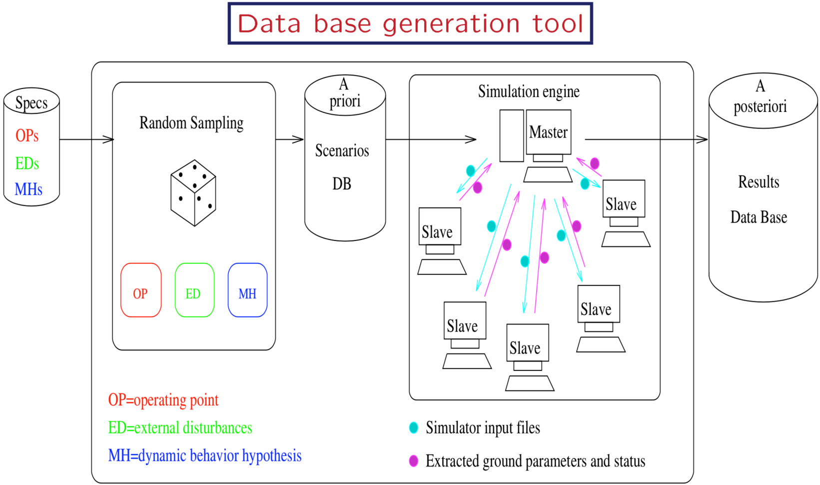
Model : 1200 buses, 450 OLTCs
CSVC, 70 machines

For a given contingency and OP :

ΔLPM : measure of severity

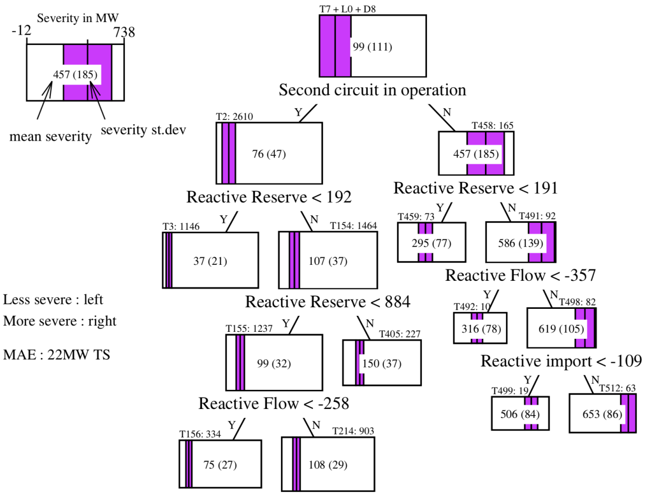


Example: Database generation by Monte-Carlo simulation



Prediction of contingency severity

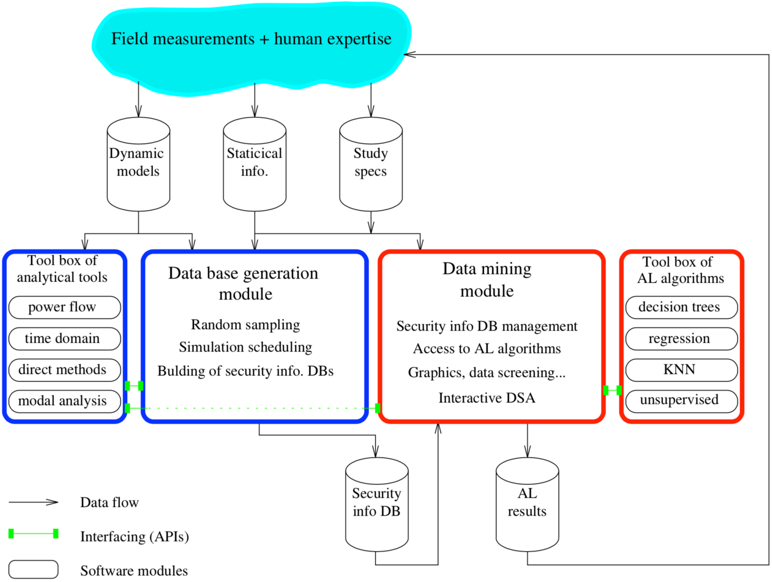
Severity regression tree : loss of a line circuit



ML for RT reliability assessment (practically)

- How often to apply ML to refresh the proxies
  - On the fly in real-time
  - Ahead in time
- How to gather the datasets used for learning the proxies
  - Passively, by exploiting data generated by EMS platforms
  - Actively, by using Monte-Carlo approaches
- How to use the tool-box of available ML techniques
  - Interpretability
  - Computational performances (learning and prediction)
  - Accuracy
- How to use the learnt proxies $h_{r,c}$
  - Stand-alone mode
  - Together with "exact" simulator of f_{cr}

Software framework



Further readings and developments

- Literature of the late 1990'ies
  - Wehenkel, Louis. "Contingency severity assessment for voltage security using non-parametric regression techniques." IEEE Transactions on Power Systems 11.1 (1996): 101-111.
  - Wehenkel, Louis A. Automatic learning techniques in power systems. Springer Science & Business Media, 2012 (first published in 1998)
- More recent machine learning methods
  - Random forests and kernel based methods
  - Gaussian processes
  - Probabilistic graphical models
  - Deep neural networks
- iTESLA European FP7 project: Machine Learning for Dynamic Security Assessment
- GARPUR European FP7 project: Probabilistic reliability management

Reliability Assessment in Look-ahead Mode (Ideally)

Every day (or every few hours), based on probability model $\mathbb{P}\{x_{rt}^{t_0 \cdots t_f}\}$ of trajectories of situations that could show up next day (next hours), evaluate the risk induced by these situations.

Reliability Assessment in Look-ahead Mode (Ideally)

Every day (or every few hours), based on probability model $\mathbb{P}\{x_{rt}^{t_0\cdots t_f}\}$ of trajectories of situations that could show up next day (next hours), evaluate the risk induced by these situations.

- Data and model:
  - $x_{rt}^t = (\xi_{rt}^t, u_{rt}^t(\xi_{rt}^t), c_{rt}(\xi_{rt}^t, u_{rt}^t(\xi_{rt}^t)))$, where ξ_{rt}^t is exogenous (weather, demand, market ...), while the endogenous parts are results of the real-time reliability management process

- Remarks:
  - Exogenous uncertainties $\xi_{rt}^{t_0\cdots t_f}$ are modelled as spatio-temporal stochastic processes conditioned on available information in look-ahead mode
  - Policy $u_{rt}^t(\xi_{rt}^t)$ models how the real-time operator will behave in real-time
  - Function $c_{rt}(\xi_{rt}^t, u_{rt}^t(\xi_{rt}^t))$ expresses the resulting cost per time step of real-time reliability management

Reliability Assessment in Look-ahead Mode (Ideally)

Every day (or every few hours), based on probability model $\mathbb{P}\{x_{rt}^{t_0 \cdots t_f}\}$ of trajectories of situations that could show up next day (next hours), evaluate the risk induced by these situations.

- Data and model:
  - $x_{rt}^t = (\xi_{rt}^t, u_{rt}^t(\xi_{rt}^t), c_{rt}(\xi_{rt}^t, u_{rt}^t(\xi_{rt}^t)))$, where ξ_{rt}^t is exogenous (weather, demand, market ...), while the endogenous parts are results of the real-time reliability management process
- We want to assess expected outcome of real-time operation:
  - $\mathbb{E}\left\{\sum_{t=t_0}^{t_f} c_{rt}(\xi_{rt}^t, u_{rt}^t(\xi_{rt}^t))\right\}$
  - $\mathbb{P}\left\{\sum_{t=t_0}^{t_f} c_{rt}(\xi_{rt}^t, u_{rt}^t(\xi_{rt}^t)) \geq M\right\}$
- Remarks:
  - Exogenous uncertainties $\xi_{rt}^{t_0 \cdots t_f}$ are modelled as spatio-temporal stochastic processes conditioned on available information in look-ahead mode
  - Policy $u_{rt}^t(\xi_{rt}^t)$ models how the real-time operator will behave in real-time
  - Function $c_{rt}(\xi_{rt}^t, u_{rt}^t(\xi_{rt}^t))$ expresses the resulting cost per time step of real-time reliability management

Opportunities for Machine Learning (in Look-ahead mode)

Look-ahead mode probabilistic reliability assessment could be solved by Monte-Carlo simulation. Various possibilities exist to make such a process more effective and practical.

- Better models of $\mathbb{P}\{\xi_{rt}^{t_0 \cdots t_f} | \text{info available in look-ahead mode}\}$
  - from observational datasets, from TSO and DSO
  - using unsupervised learning, e.g. convolutional GANs ?
- Learning about real-time operation strategy $u_{rt}^{t}(\xi_{rt}^{t})$ and/or $c_{rt}(\xi_{rt}^{t})$
  - from observational datasets collected by SCADA and EMS
  - from simulations
- Reducing the number of required Monte-Carlo samples to estimate
  - $\mathbb{E}\left\{\sum_{t=t_0}^{t_f} c_{rt}(\xi_{rt}^{t}, u_{rt}^{t}(\xi_{rt}^{t}))\right\}$
    (Variance reduction via control variates, and/or importance sampling)
  - $\mathbb{P}\left\{\sum_{t=t_0}^{t_f} c_{rt}(\xi_{rt}^{t}, u_{rt}^{t}(\xi_{rt}^{t})) \geq M\right\}$
    (Rare event simulation via importance sampling)

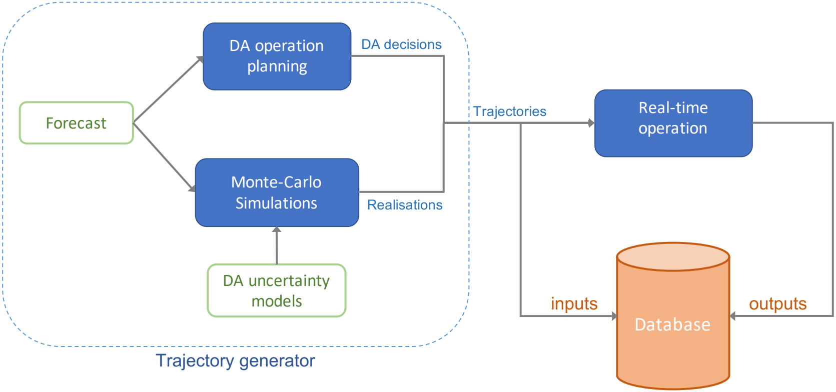
Example: Machine Learning for Day-ahead reliability assessment

Based on ongoing work at ULiège.

For further details, please see

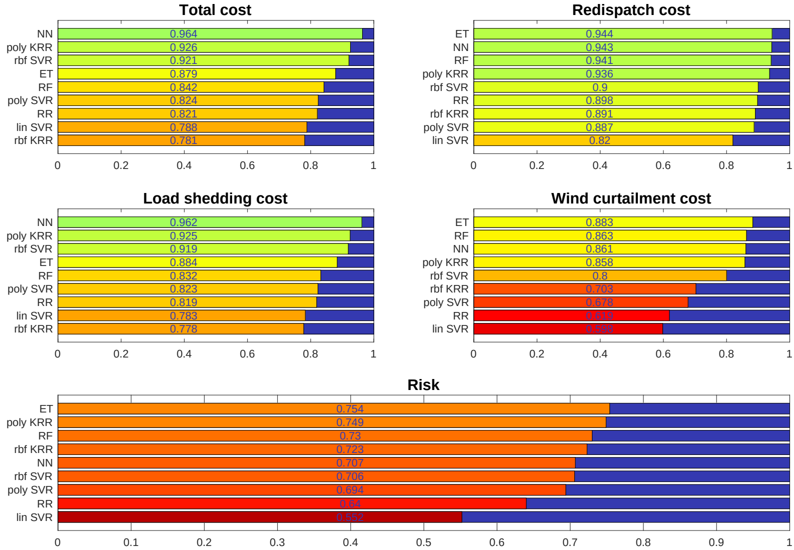
- Machine Learning of Real-time Power Systems Reliability Management Response, L. Duchesne et. al, IEEE PowerTech 2017
- Using Machine Learning to Enable Probabilistic Reliability Assessment in Operation Planning, L. Duchesne et. al, PSCC 2018

Day-ahead learning of RT operator response (PowerTech 2017)

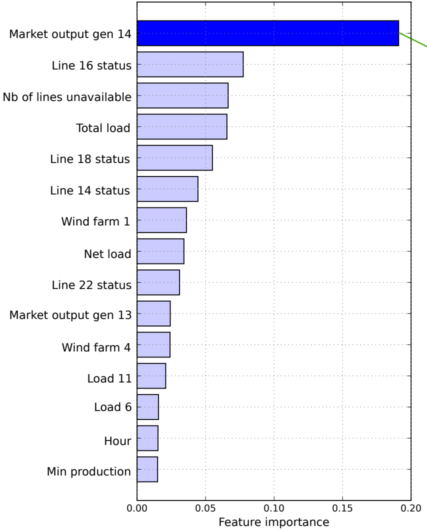


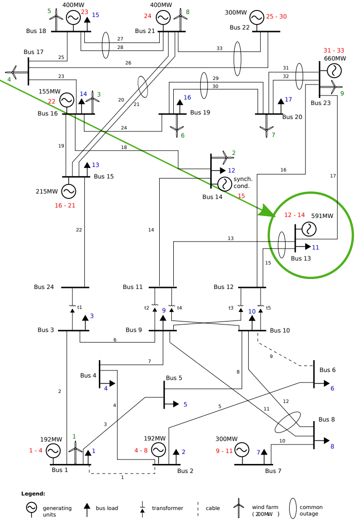
- 4000 samples of uncertainty realizations ξ_{rt} along next day (load, wind, outages)
- Real-time operation simulated by N-1 DC-SCOPF per time-step and trajectory
- Inputs: DA decisions per time-step, ξ_{rt} per trajectory and time-step
- Outputs: the different terms of the cost function c_{rt}, including risk $\mathbb{E}\{f_{rt}|\xi_{rt}, u_{rt}\}$ of service-interruptions upon contingencies, per trajectory and per time-step.

Some Machine Learning results (PowerTech 2017)

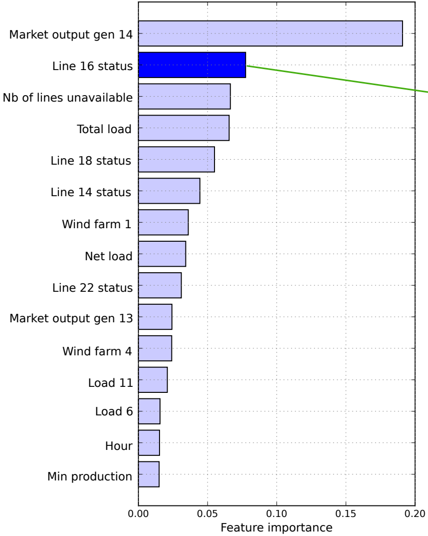


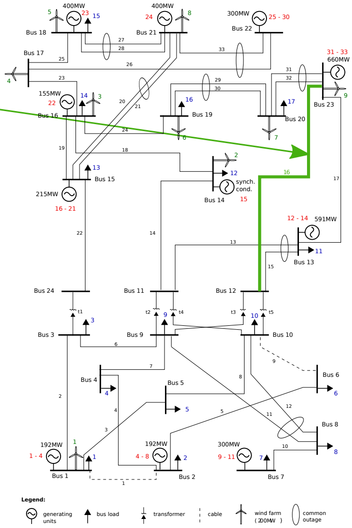
Some Machine Learning results (PowerTech 2017)





Some Machine Learning results (PowerTech 2017)





Synthesis (PowerTech 2017)

- Machine learning can be used day-ahead to build "proxies" $\hat{c}_{rt}$ of the different terms of c_{rt} incurred in real-time reliability management.
- Computationally, evaluating $\hat{c}_{rt}$ is about 10000 times faster than the "exact" evaluation of c_{rt} via SCOPF and contingency simulation
- Random forests and Neural networks are promizing and complementary tools in this context
- Some terms of $\hat{c}_{rt}$ are more difficult to learn than others, in particular the expected risk induced by contingencies
- Open questions for further work:
    - Leverage deep learning to improve accuracy of proxies $\hat{c}_{rt}$
    - Use of machine learning to model the RT decision policy u_{rt}
    - Use of learnt proxies $\hat{c}_{rt}$ for day-ahead reliability assessment

Use of ML-proxies for DA reliability assessment (PSCC 2018)

Problem tackled:

- Given two computer programs
  - a generative model allowing us to sample possible next day trajectories according to $\mathbb{P}\{\xi_{rt}^{t_0 \cdots t_f} | \text{day ahead info}\}$
  - and a SCOPF solver allowing us to compute operating costs $C_{rt}(\xi_{rt}) = \sum_{t=1}^{24} c_{rt}(\xi_{rt}^{t})$ along any trajectory
- Compute an estimate of $\mathbb{E}\{C_{rt} | \text{day ahead info}\}$

Crude Monte Carlo (CMC) approach:

- Sample n trajectories $\xi_{rt}^{i} \sim \mathbb{P}\{\xi_{rt}^{t_0 \cdots t_f} | \text{day ahead info}\}$
- Compute $\bar{C}_{rt} = \frac{1}{n} \sum_{i=1}^{n} C_{rt}(\xi_{rt}^{i}) = \frac{1}{n} \sum_{i=1}^{n} \sum_{t=1}^{24} c_{rt}(\xi_{rt}^{t,i})$
- Needs large sample size n (a few thousand) to be accurate enough
- Requires $24 \times n$ SCOPF computations

Use of ML-proxies for DA reliability assessment (PSCC 2018)

We could (naively) use Machine Learning as follows:

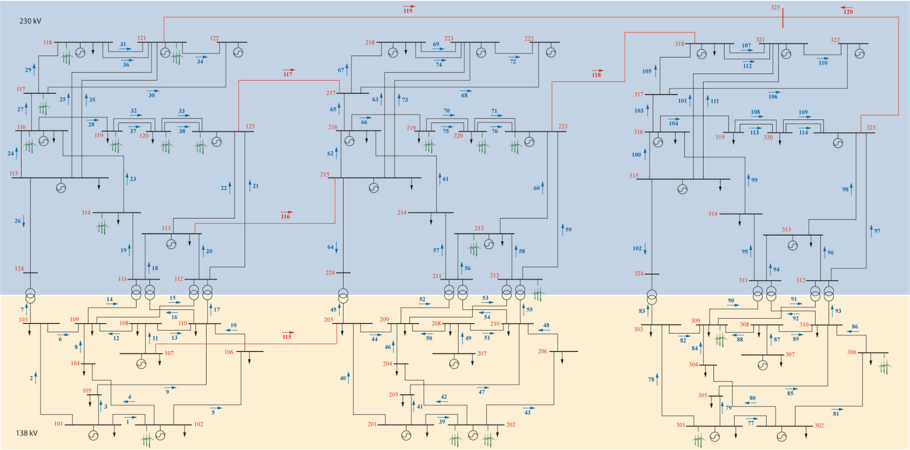
- Machine learning stage:
  - Sample $k \ll n$ trajectories ξ^i_{rt} and use SCOPF to compute the corresponding $k \times 24$ values of $c_{rt}(\xi^{i,t}_{rt})$
  - Use a supervised learning algorithm to build proxy $\hat{c}_{rt}(\cdot)$, much faster to evaluate than SCOPF
- Use CMC with proxy
  - Sample $n' \gg n$ additional trajectories and use proxy to compute $\bar{\hat{C}}_{rt} = \frac{1}{n'} \sum_{i=1}^{n'} \sum_{i=1}^{24} \hat{c}_{rt}(\xi^{i,t}_{rt}) \simeq \mathbb{E}\{\hat{C}_{rt} | \text{day ahead info}\}$.
- Unfortunately, this later quantity is in general not equal to $\mathbb{E}\{C_{rt} | \text{day ahead info}\}$
- Its bias depends both on the problem and on the used machine learning algorithm, and is therefore unpredictable.

Use of ML-proxies for DA reliability assessment (PSCC 2018)

Combining Machine Learnt proxies with Control Variate approach:

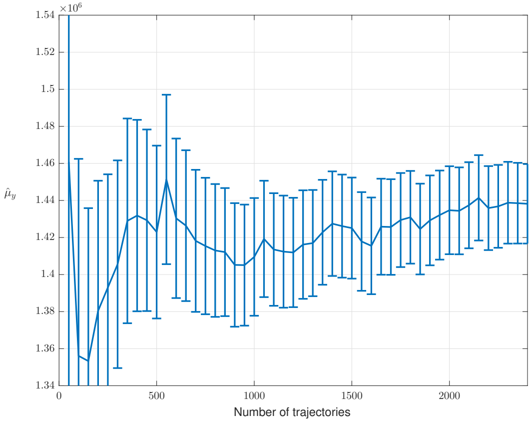
- First do as in the previous slide:
  - Learn proxy $\hat{c}_{rt}(\cdot)$ with k trajectories
  - Estimate $\mathbb{E}\{\hat{C}_{rt}|\text{day ahead info}\}$ with large n' trajectories.
- Then estimate $\mathbb{E}\{\hat{C}_{rt}|\text{day ahead info}\} - \mathbb{E}\{C_{rt}|\text{day ahead info}\}$
  - Sample k' additional trajectories
  - Compute $\bar{\Delta} = \frac{1}{k'}\sum_{j=1}^{k'}\left(\hat{C}_{rt}(\xi_{rt}^{j}) - C_{rt}(\xi_{rt}^{j})\right)$
  - Estimate $\mathbb{E}\{C_{rt}|\text{day ahead info}\} \simeq \mathbb{E}\{\hat{C}_{rt}|\text{day ahead info}\} - \bar{\Delta}$
- This latter estimate is always unbiased
- For a given budget of $(k + k')$ trajectories solved via SCOPF, it is typically more accurate than CMC with $n = k + k'$ trajectories

Case study: 3-area RTS system (PSCC 2018)



NB: modified by including lots of wind power plants

Crude Monte-Carlo approach (PSCC 2018)

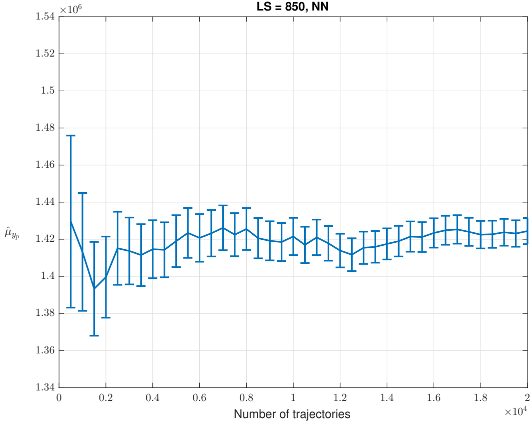


For $n = 2400$ trajectories:

Estimate $= 1.44\ 10^6$
Std.error $= 2\ 10^4 =$ **1.4%**

- Operating cost for one trajectory: 24 successive DC-SCOPF computations
- Sample $n = 2400$ trajectories, and estimate expectation by sample average
- Standard error is estimated by $\sigma/\sqrt{n}$, σ being the sample standard deviation

Naive use of Machine Learnt proxy (PSCC 2018)

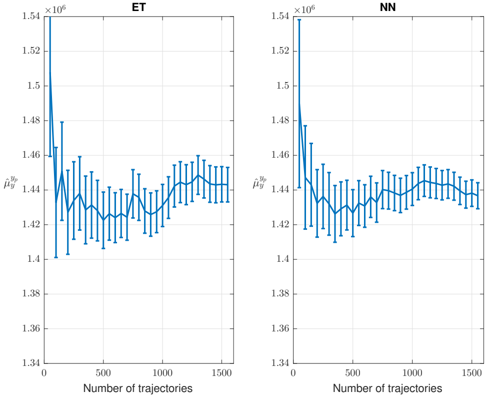


For $n' = 20000$ trajectories:

Estimate $= 1.42\ 10^6$
Std.error $= 4\ 10^{\mathbf{3}} = \mathbf{0.3\%}$

- Proxy of hourly operating cost learnt on $k = 850$ trajectories, using ANN
- Estimate expectation via much larger sample, by using only the proxy
- Unfortunately, using the proxy we get a biased estimate (by about 1.4 %)

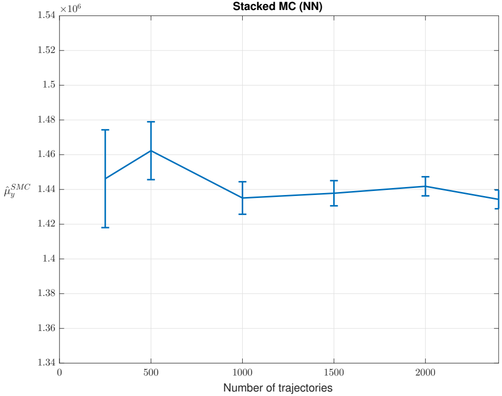
Use of ML proxy as a Control Variate (PSCC 2018)



Estimate $= 1.44\ 10^6$
Std.error $= 1\ 10^4 = \mathbf{0.7}\%$

- Proxy learnt on $k = 850$ trajectories, using resp. ET or ANN
- Estimate on k' up to 1550 additional samples by the control variates approach
- Yields unbiased estimate of reduced std.error (factor 2), for same SCOPF budget

Further refinement: Stacked Monte-Carlo (PSCC 2018)



Estimate $= 1.44\ 10^6$

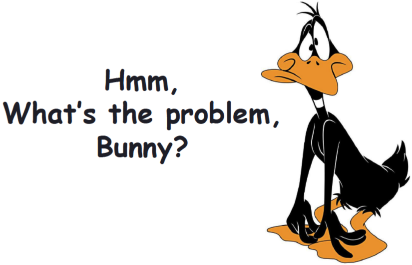
Std.error $= 5\ 10^3 = \mathbf{0.35\%}$

- Yields unbiased estimate of reduced std.error by a factor 4
- Uses SCOPF budget in a more effective way to reduce both bias and variance
- See paper for explanation of the method

Synthesis (PSCC 2018)

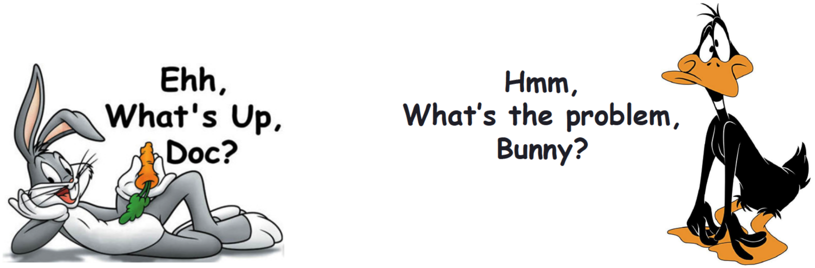
- Machine learning can be used in a sound way to significantly speed up day-ahead reliability assessment under uncertainties
- Computationally, a speed-up of a factor 10-20 with respect to a crude Monte-Carlo approach is certainly reachable
- Further leveraging deep neural networks may help to make the approach even more effective
- Open questions for further work:
  - Adaptation of the proposed framework for estimating probabilities of rare events $\mathbb{P}\left\{\sum_{t=t_0}^{t_f} c_{rt}(\xi_{rt}^t, u_{rt}^t(\xi_{rt}^t)) \geq M\right\}$
  - Combination of this approach with appropriate techniques for finding suitable day-ahead decisions

Reliability Assessment in mid/long-term modes (Ideally)



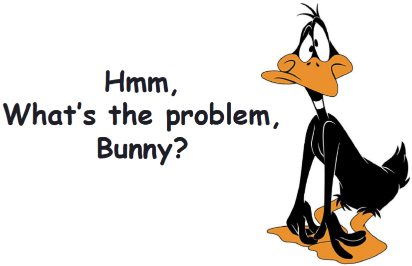
- Maintenance optimization and system development contexts

Reliability Assessment in mid/long-term modes (Ideally)



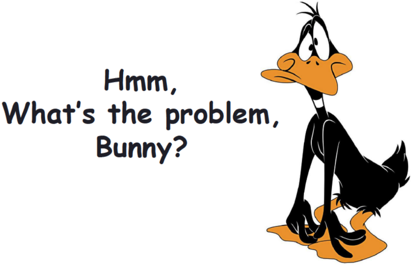
- Maintenance optimization and system development contexts
- Look-ahead horizons: months to years; years to decades

Reliability Assessment in mid/long-term modes (Ideally)



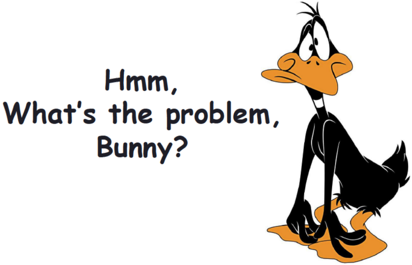
- Maintenance optimization and system development contexts
- Look-ahead horizons: months to years; years to decades
- Complexity multiplied by 8800 hrs $\times$ 30 years

Reliability Assessment in mid/long-term modes (Ideally)



- Maintenance optimization and system development contexts
- Look-ahead horizons: months to years; years to decades
- Complexity multiplied by 8800 hrs $\times$ 30 years
- Uncertainty models even more complex to establish

Reliability Assessment in mid/long-term modes (Ideally)



- Maintenance optimization and system development contexts
- Look-ahead horizons: months to years; years to decades
- Complexity multiplied by 8800 hrs $\times$ 30 years
- Uncertainty models even more complex to establish
- Many opportunities for Machine Learning...

Part III

Machine Learning for Reliability Control

Reliability control:

Determine an optimal decision u^* to take in order to ensure a desired level of reliability over a given time period:

- Real-time mode
- Short-term look-ahead mode
- Longer-term look-ahead problems

Possible Optimal Control Approaches

- Analytical approach: formulate equations and solve them to near-optimality; it is the realm of mathematical optimization; needs lots of approximations to be tractable.

Possible Optimal Control Approaches

- Analytical approach: formulate equations and solve them to near-optimality; it is the realm of mathematical optimization; needs lots of approximations to be tractable.
- Brute force "Trial and error" approach: using 'Reliability assessment module' as an oracle: can work well when small number of alternative decisions have to be compared; can exploit further ideas of variance reduction and machine learning.

Possible Optimal Control Approaches

- Analytical approach: formulate equations and solve them to near-optimality; it is the realm of mathematical optimization; needs lots of approximations to be tractable.
- Brute force "Trial and error" approach: using 'Reliability assessment module' as an oracle: can work well when small number of alternative decisions have to be compared; can exploit further ideas of variance reduction and machine learning.
- Off-line policy search: create structured space of candidate decision policies, and sample them together with scenarios used to assess by simulation the candidate policies. Interleave policy search steps and proxy-learning steps in a suitable way.

Possible Optimal Control Approaches

- Analytical approach: formulate equations and solve them to near-optimality; it is the realm of mathematical optimization; needs lots of approximations to be tractable.
- Brute force "Trial and error" approach: using 'Reliability assessment module' as an oracle: can work well when small number of alternative decisions have to be compared; can exploit further ideas of variance reduction and machine learning.
- Off-line policy search: create structured space of candidate decision policies, and sample them together with scenarios used to assess by simulation the candidate policies. Interleave policy search steps and proxy-learning steps in a suitable way.
- On-line reinforcement learning approach: interleave learning and decision making, while taking advantage of simulators and proxies designed ahead in time.

Some further bibliographical pointers

P. Panciatici, G. Bareux and L. Wehenkel
Operating in the fog - Security management under uncertainty
IEEE Power & Energy Magazine, 2012, September/October, 40-49

E. Karangelos, P. Panciatici and L. Wehenkel
Whither probabilistic security management for real-time operation of power systems ?
Proc. of IREP Symposium, Rethymnon 2013

E. Karangelos and L. Wehenkel
Probabilistic reliability management approach and criteria for power system real-time operation
Proc. of PSCC, Genoa 2016

E. Karangelos and L. Wehenkel
Probabilistic reliability management approach and criteria for power system short-term operational planning
Proc. of IREP Symposium, Porto 2017

L. Duchesne, E. Karangelos, and L. Wehenkel
Machine learning of real-time power systems reliability management response
Proc. of IEEE PowerTech, Manchester 2017

L. Duchesne, E. Karangelos, and L. Wehenkel
Using machine learning to enable probabilistic reliability assessment in operation planning
Proc. of PSCC, Dublin 2018

E. Karangelos, and L. Wehenkel
Post-contingency corrective control failure: a risk to neglect or a risk to control?
Proc. of PMAPS, Boise 2018

Available on-line at: `http://orbi.ulg.ac.be/simple-search?query=wehenkel`