An iterative AC-SCOPF approach managing the contingency and corrective control failure uncertainties with a probabilistic guarantee

Efthymios Karangelos and Louis Wehenkel,
Institut Montefiore,
Universitè de Liège, Belgium.

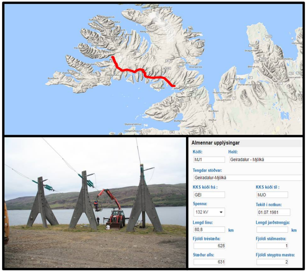
October 18th 2019, KU Leuven, Belgium.

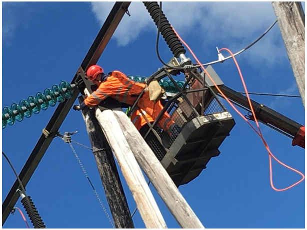
Presentation outline

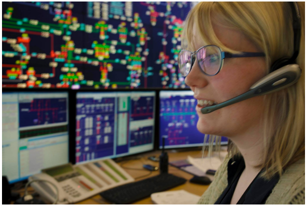
1. Background & motivation
2. Problem formulation
3. Algorithmic solution approach
4. Case study results
5. Wrap-up & future work

Reliability management

- Making decisions under uncertainty, from long-term system development to real-time system operation.







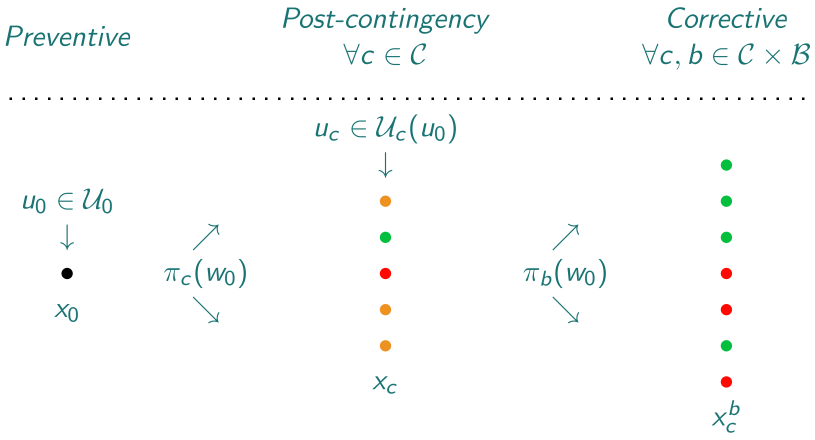
- A reliability criterion sets the basis to determine whether or not the system reliability is acceptable.

Real-time operation reliability management

Horizon: (5' $\sim$ 15')

- Power injections assumed relatively predictable.
- Uncertainty on:
  - $\rightarrow$ occurrence of contingencies $c \in \mathcal{C}$;
  - $\rightarrow$ behavior of post-contingency corrective controls $b \in \mathcal{B}$.
- Decisions to:
  - $\rightarrow$ apply preventive (pre-contingency) control $u_0 \in \mathcal{U}_0(x_0)$?
  - $\rightarrow$ prepare post-contingency corrective controls $u_c \in \mathcal{U}_c\,(u_0)\,\forall c \in \mathcal{C}$?

Transitions of the system state



w_0: spatial/temporal correlation in transition probabilites.

Security Constrained Optimal Power Flow

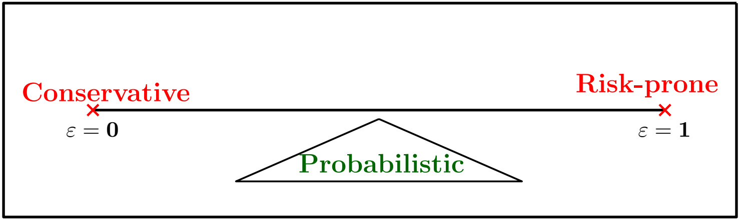
The N-1 approach

- Maintain stable equilibrium (system operational limits) following any single outage,
- but, how to rely on uncertain corrective control?
  - → don't, since it may fail (conservative)?
  - → do, just neglect failure (risk-prone)?

Security Constrained Optimal Power Flow

The probabilistic approach

- Maintain stable equilibrium (system operational limits), at least with a certain confidence,
  - $\rightarrow$ so that the joint probability of violating operational limits remains below a tolerance $\varepsilon \in [0, 1]$.



Presentation outline

1. Background & motivation
2. Problem formulation
3. Algorithmic solution approach
4. Case study results
5. Wrap-up & future work

Steady-state operational limits

- AC power flow (rectangular coordinates);
- voltage magnitude bounds per node;
- voltage angle difference & apparent power flow bounds per branch;
  - → less restrictive for the intermediate problem stage;
- active & reactive power generation bounds per unit;
  - → ramping restrictions between preventive & corrective active power dispatch;
- voltage set-points per unit;
- no loss of load.

State-of-the-art determinstic-constrained problem

$$\min_{\mathbf{u}\in\mathbf{U}} CP(x_0, u_0) \tag{1}$$

$$h_0(x_0, u_0) \leq 0; \tag{2}$$

$$h_c^s(x_c, u_0) \leq 0 \quad \forall c \in \mathcal{C}; \tag{3}$$

$$h_c(x_c^w, u_c) \leq 0 \quad \forall c \in \mathcal{C}; \tag{4}$$

$$\mathbf{u} \in \mathbf{U} \equiv \{u_0 \in \mathcal{U}_0(x_0); u_c \in \mathcal{U}_c(x_0, u_0, c) \forall c \in \mathcal{C}\}. \tag{5}$$

- Minimizing the cost of preventive operation (**??**);
- $h_.(x_., u_.)$ groups all operational limits for preventive, intermediate and corrective stage (**??** - **??**);
- only for perfectly working corrective controls (x_c^w);
- preventive & corrective decisions are coupled (**??**).

Our chance-constrained problem

$$\min_{\mathbf{u}\in\mathbf{U}} CP(x_0, u_0) + \sum_{c\in\mathcal{C}} \pi_c \cdot CC(x_0, u_0, c, u_c); \tag{6}$$

$$h_0(x_0, u_0) \leq 0; \tag{7}$$

$$\mathbb{P}\left\{ \begin{array}{l} h_c^s(x_c, u_0) \leq 0 \\ h_c(x_c^b, u_c) \leq 0 \end{array} \middle| (c, b) \in \mathcal{C} \times \mathcal{B} \right\} \geq 1 - \varepsilon; \tag{8}$$

$$\mathbf{u} \in \mathbf{U} \equiv \{u_0 \in \mathcal{U}_0(x_0); u_c \in \mathcal{U}_c(x_0, u_0, c) \forall c \in \mathcal{C}\}. \tag{9}$$

- Also includes expectation of corrective stage costs (**??**);
- post-contingency operational limits to hold at least with $(1 - \varepsilon)$ probability (**??**);
- taking into account contingency occurrence probabilities & corrective control behavior probabilities.

Chance-constraint reformulation – step one

$$\mathbb{P}\left\{ \begin{array}{l} h_c^s(x_c, u_0) \leq 0 \\ h_c(x_c^b, u_c) \leq 0 \end{array} \middle| (c, b) \in \mathcal{C} \times \mathcal{B} \right\} \geq 1 - \varepsilon;$$

- LH-side measures the probability of all post-contingency stages meeting operational constraints;

Chance-constraint reformulation – step one

$$\mathbb{P}\left\{ \begin{array}{l} h_c^s(x_c, u_0) \leq 0 \\ h_c(x_c^b, u_c) \leq 0 \end{array} \middle| (c, b) \in \mathcal{C} \times \mathcal{B} \right\} \geq 1 - \varepsilon;$$

- LH-side measures the probability of all post-contingency stages meeting operational constraints;
- introducing indicator function $\mathcal{I}(x_0, u_0, c, u_c, b)$ to show post-contingency constraint violations,

$$\mathcal{I}(x_0, u_0, c, u_c, b) = \begin{cases} 1 \equiv \{h_c^s(x_c, u_0) \not\leq 0 \vee h_c(x_c^b, u_c) \not\leq 0\} \\ 0 \equiv \{h_c^s(x_c, u_0) \leq 0 \wedge h_c(x_c^b, u_c) \leq 0\} \end{cases}.$$

Chance-constraint reformulation – step one

$$\mathbb{P}\left\{ \begin{array}{l} h_c^s(x_c, u_0) \leq 0 \\ h_c(x_c^b, u_c) \leq 0 \end{array} \middle| (c, b) \in \mathcal{C} \times \mathcal{B} \right\} \geq 1 - \varepsilon;$$

- introducing indicator function $\mathcal{I}(x_0, u_0, c, u_c, b)$ to show post-contingency constraint violations,

$$\mathcal{I}(x_0, u_0, c, u_c, b) = \begin{cases} 1 \equiv \{h_c^s(x_c, u_0) \not\leq 0 \vee h_c(x_c^b, u_c) \not\leq 0\} \\ \\ 0 \equiv \{h_c^s(x_c, u_0) \leq 0 \wedge h_c(x_c^b, u_c) \leq 0\} \end{cases},$$

- when sets of contingencies and corrective control behaviors are discrete, one may re-write the chance-constraint as,

$$1 - \sum_{c \in \mathcal{C}} \pi_c \sum_{b \in \mathcal{B}} \pi_c^b \cdot \mathcal{I}(x_0, u_0, c, u_c, b) \geq 1 - \varepsilon.$$

Our chance-constrained problem

$$\min_{\mathbf{u}\in\mathbf{U}} CP(x_0, u_0) + \sum_{c\in\mathcal{C}} \pi_c \cdot CC(x_0, u_0, c, u_c); \quad (10)$$

$$h_0(x_0, u_0) \leq 0; \quad (11)$$

$$1 - \sum_{c\in\mathcal{C}} \pi_c \sum_{b\in\mathcal{B}} \pi_c^b \cdot \mathcal{I}(x_0, u_0, c, u_c, b) \geq 1 - \varepsilon; \quad (12)$$

$$\mathbf{u} \in \mathbf{U} \equiv \{u_0 \in \mathcal{U}_0(x_0); u_c \in \mathcal{U}_c(x_0, u_0, c) \forall c \in \mathcal{C}\}. \quad (13)$$

- Reformulated chance-constraint (**??**) sums the indicator function over all contingencies & corrective control behaviors!

Chance-constraint reformulation – step two

$$\mathbb{P}\left\{ \begin{array}{l} h_c^s(x_c, u_0) \leq 0 \\ h_c(x_c^b, u_c) \leq 0 \end{array} \middle| (c, b) \in \mathcal{C} \times \mathcal{B} \right\} \geq 1 - \varepsilon;$$

- Intermediate stage constraints need to hold to keep the system functional;
- we can partly replace indicator function $\mathcal{I}(x_0, u_0, c, u_c, b)$ with auxiliary binary variables ($p_c \in [0; 1]$) relaxing these constraints,

$$h_c^s(x_c, u_0) \leq p_c \cdot M, \forall c \in \mathcal{C}, \tag{14}$$

with M being a sufficiently large constant.

Chance-constraint reformulation – step two

$$\mathbb{P}\left\{\begin{array}{l} h^s_c(x_c, u_0) \leq 0 \\ h_c(x^b_c, u_c) \leq 0 \end{array} \middle| (c, b) \in \mathcal{C} \times \mathcal{B}\right\} \geq 1 - \varepsilon;$$

- In a cost minimization context, corrective control would only be selected when needed;
- that is, to alleviate some post-contingency constraint violation;
- when corrective control doesn't work, we'd have constraint violations;
- hence, every post-contingency stage with corrective actions contributes to the sum appearing in the LH-side of the chance constraint (**??**).

Chance-constraint reformulation – step two

- Every post-contingency stage with corrective actions contributes to the sum appearing in the LH-side of the chance constraint (**??**);
- we can partly replace indicator function $\mathcal{I}(x_0, u_0, c, u_c, b)$ with auxiliary binary variables ($i_c \in [0;1]$) showing the use of post-contingency corrective controls;

$$\mathcal{I}(x_0, u_0, c, u_c, b) \equiv \left\{ \begin{array}{l} h_c^s(x_c, u_0) \leq p_c \cdot M \\ h_c(x_c^w, u_c) \leq 0 \\ |u_0 - u_c| \leq i_c \cdot M \\ i_c \in [0;1] \end{array} \right\}, \forall c \in \mathcal{C}, \qquad (15)$$

with M being a sufficiently large constant.

Our chance-constrained problem

$$\min_{\mathbf{u}\in\mathbf{U}} CP(x_0,u_0) + \sum_{c\in\mathcal{C}} \pi_c \cdot CC(x_0,u_0,c,u_c); \tag{16}$$

$$h_0(x_0,u_0) \leq 0; \tag{17}$$

$$h_c^s(x_c,u_0) \leq p_c \cdot M, \forall c \in \mathcal{C}, \tag{18}$$

$$h_c(x_c^w,u_c) \leq 0, \forall c \in \mathcal{C}, \tag{19}$$

$$1 - \sum_{c\in\mathcal{C}} \pi_c \cdot [p_c + (1-\pi_c^w)\cdot i_c] \geq 1-\epsilon; \tag{20}$$

$$|u_0 - u_c| \leq i_c \cdot M, \forall c \in \mathcal{C}, \tag{21}$$

$$i_c + p_c \leq 1, \forall c \in \mathcal{C}, \tag{22}$$

$$i_c, p_c \in [0;1], \forall c \in \mathcal{C}, \tag{23}$$

$$\mathbf{u} \in \mathbf{U} \equiv \{u_0 \in \mathcal{U}_0(x_0); u_c \in \mathcal{U}_c(x_0,u_0,c) \forall c \in \mathcal{C}\}. \tag{24}$$

- Additional coupling constraints and binary vars (**??**,**??**-**??**).

Presentation outline

1. Background & motivation
2. Problem formulation
3. Algorithmic solution approach
4. Case study results
5. Wrap-up & future work

Algorithm motivation

The deterministic-constrained problem

- Optimistic attitude towards corrective control failures;
- a large-scale Mixed-Integer Non-Linear Programming (MINLP) problem;
- state-of-the-art solution approach is contingency filtering.

Algorithm motivation

The deterministic-constrained problem

- Optimistic attitude towards corrective control failures;
- a large-scale Mixed-Integer Non-Linear Programming (MINLP) problem;
- state-of-the-art solution approach is contingency filtering.

Our chance-constrained problem

- Reformulated as a MINLP;
- includes constraints from the optimistic version and then some more;
- how can we adapt contingency filtering schemes?

Solution principle

- Any chosen decision partitions the contingency set ...

Preventive Only $\mathcal{C}_{\mathcal{P}}$	Preventive & Corrective $\mathcal{C}_{\mathcal{C}}$	Not Secured $\mathcal{C}_{\mathcal{X}} = \mathcal{C} \setminus (\mathcal{C}_{\mathcal{C}} \cup \mathcal{C}_{\mathcal{P}})$

Solution principle

- Any chosen decision partitions the contingency set . . .

Preventive Only $\mathcal{C}_{\mathcal{P}}$	Preventive & Corrective $\mathcal{C}_{\mathcal{C}}$	Not Secured $\mathcal{C}_{\mathcal{X}} = \mathcal{C} \setminus (\mathcal{C}_{\mathcal{C}} \cup \mathcal{C}_{\mathcal{P}})$

+ we get a lower-bound for the probability of interest;

$$\mathbb{P}\Big\{ \dots \Big\} \geq 1 - \left(\sum_{c \in \mathcal{C}_{\mathcal{X}}} \pi_c + \sum_{c \in \mathcal{C}_{\mathcal{C}}} \pi_c \cdot \pi_c^f \right),$$

e.g., $\mathbb{P}\Big\{ \dots \Big\} \geq 1$ when all cntgcies are in preventive only.

Solution principle

- What if we grow secured contingency sub-sets $\mathcal{C}_{\mathcal{P}}, \mathcal{C}_{\mathcal{C}}$?

Preventive Only $\mathcal{C}_{\mathcal{P}}$	Preventive & Corrective $\mathcal{C}_{\mathcal{C}}$	Not Secured $\mathcal{C}_{\mathcal{X}} = \mathcal{C} \setminus (\mathcal{C}_{\mathcal{C}} \cup \mathcal{C}_{\mathcal{P}})$

Solution principle

- What if we grow secured contingency sub-sets $\mathcal{C}_{\mathcal{P}}, \mathcal{C}_{\mathcal{C}}$?

Preventive Only $\mathcal{C}_{\mathcal{P}}$	Preventive & Corrective $\mathcal{C}_{\mathcal{C}}$	Not Secured $\mathcal{C}_{\mathcal{X}} = \mathcal{C} \setminus (\mathcal{C}_{\mathcal{C}} \cup \mathcal{C}_{\mathcal{P}})$

- we could push the probability lower-bound upwards,

$$\mathbb{P}\Big\{ \dots \Big\} \geq 1 - \left(\sum_{c \in \mathcal{C}_{\mathcal{X}}} \pi_c + \sum_{c \in \mathcal{C}_{\mathcal{C}}} \pi_c \cdot \pi_c^f \right),$$

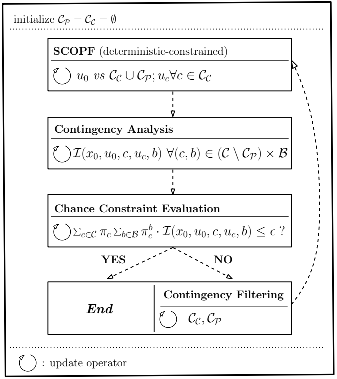
- until the actual probability value grows large enough.

Algorithmic decomposition overview

In a nutshell

1. update decisions *vs* deterministic constraints;
2. evaluate post-contingency violation probability;
3. update contingency subsets;
   - preventive only;
   - preventive & corrective;

– stop when reliability target is OK.



Algorithm components

Deterministic SCOPF

- standard IPOPT implementation *vs* given contingency subsets;

Contingency analysis

- examining both the working & failing behavior of corrective controls;
- per contingency & cc behavior, minimization of fictitious active/reactive power injections;
- returns a zero optimal value for feasible OPF instances;
- non-zero objective indicative of the magnitude of constraint violations implied by the contingency & cc behavior.

Contingency filtering variants

Probability-based (Pb)

- returns the most probable constraint-violating post-contingency stage (*e.g.*, immediately after contingency 14, after contingency 42 and cc failure, *etc.*).

Feasibility-based (Fb)

- returns the most severe constraint-violating post-contingency stage (*e.g.*, immediately after contingency 14, after contingency 42 and cc failure, *etc.*).

Risk-based (Rb)

- blends the former two, ranking post-contingency stages in probability $\times$ severity.

Contingency subset updating rule

- the goal is to push the lower bound on the constraint violation probability;
- we always tighten the security status of the filtered contingency:

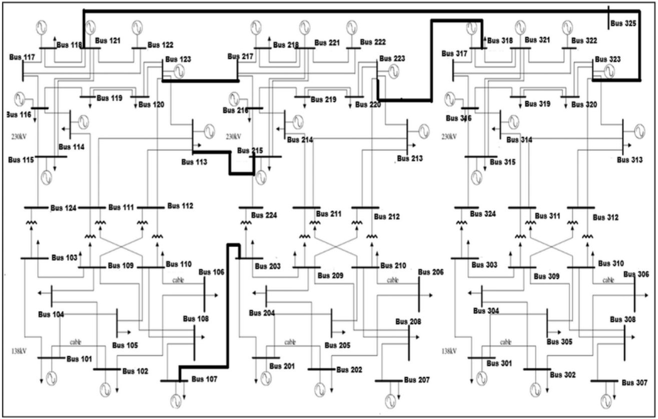
  $c \in \mathcal{C}_{\mathcal{X}} \rightarrow c \in \mathcal{C}_{\mathcal{C}}$: from not secured, to correctively & preventively secured;

  $c \in \mathcal{C}_{\mathcal{C}} \rightarrow c \in \mathcal{C}_{\mathcal{P}}$: from correctively & preventively secured, to preventively only secured;

- that is, make the contingency set partitioning more conservative.

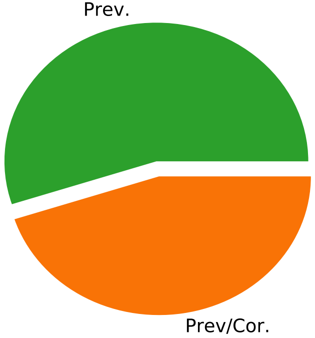
Presentation outline

The test-case



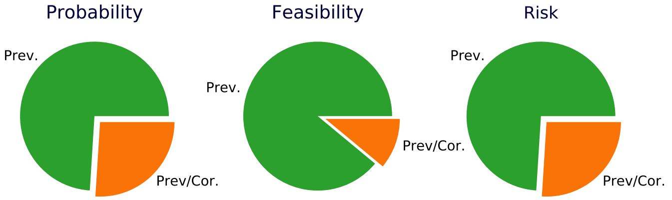
- 111 single component outages;
- Corrective control failure probability assumed 0.01.

Deterministic-constrained SCOPF



Total Cost ($)	881.62
Explicit Contingencies	4
Violation Probability	$1.91 \cdot 10^{-5}$ (*ex-post*)

Chance-constrained SCOPF ($\varepsilon = 10^{-5}$)



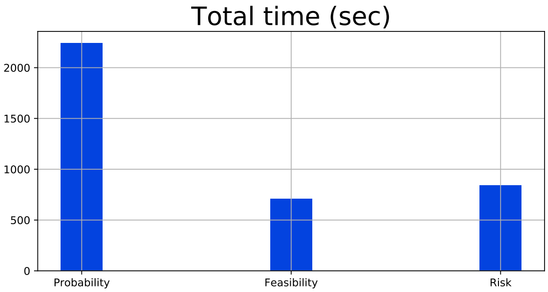
Filter	Probability	Feasibility	Risk
Total Cost ($)	892.37	896.78	892.37
Explicit Contingencies	13	5	7
Chance level	$9.85 \cdot 10^{-6}$	$5.28 \cdot 10^{-6}$	$9.85 \cdot 10^{-6}$

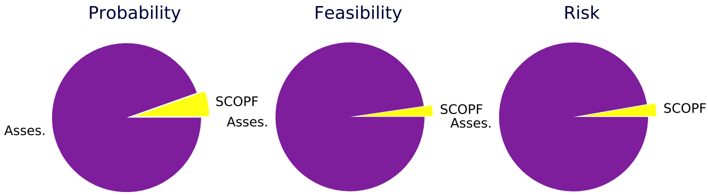
Chance-constrained SCOPF ($\varepsilon = 10^{-5}$)

Filter	Probability	Feasibility	Risk
Total Cost (\$)	892.37	896.78	892.37
Explicit Contingencies	13	5	7
Chance level	$9.85 \cdot 10^{-6}$	$5.28 \cdot 10^{-6}$	$9.85 \cdot 10^{-6}$

- More reliable solutions naturally more costly!
- risk-based filter returns the same solution & a sub-set of the cntgcies filtered by the probability-based;
- feasibility-based filter is more efficient (only 5 explicity cntgcies) yet more conservative (cost & chance-constraint level).

Chance-constrained SCOPF ($\varepsilon = 10^{-5}$)

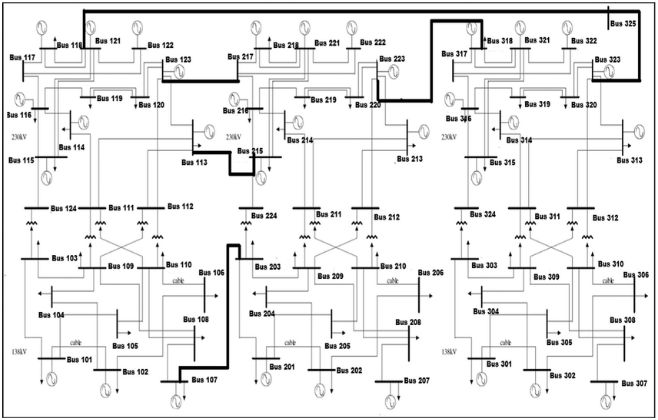




The story so far ...

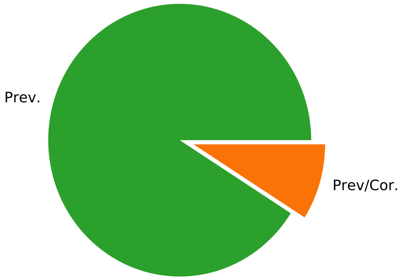
- The probability-based filter is outperformed:
  - → since all probabilities here are exogenous params, it carries no physical information;
  - → careful before generalizing to a different context (*e.g.* when cc failure probability depends on the chosen actions);
- the question of feasibility- *vs* risk-based filtering is open:
  - → feasibility-based is slightly more conservative;
  - → risk-based solves slightly slower;
- demonstrated results verified through sensitivity analysis (see full paper).

3 additional test-cases



- Taking into account weather impact on outage probabilities;
- assuming adverse weather hits any one of the system areas.

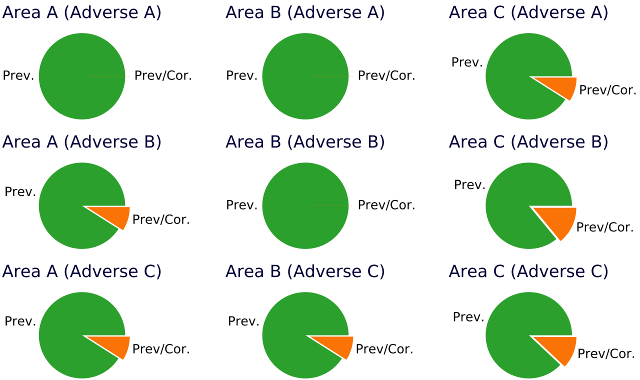
Feasibility-based filtering



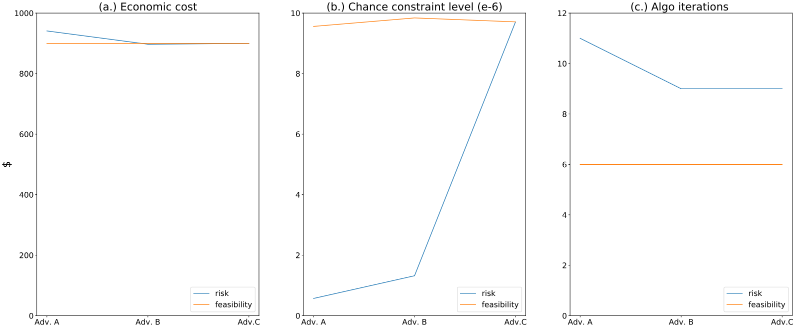
- Contingency filtering unaffected by occurrence probability.

Adverse Weather	Area A	Area B	Area C
Total Cost ($)		899.62	
Explicit Contingencies		6	
Violation Probability	$9.56 \cdot 10^{-6}$	$9.84 \cdot 10^{-6}$	$9.71 \cdot 10^{-6}$

Risk-based filtering



Risk-based filtering



- Feasibility-based filtering more robust w.r.t. the adverse weather.

Presentation outline

1. Background & motivation
2. Problem formulation
3. Algorithmic solution approach
4. Case study results
5. Wrap-up & future work

Wrap-up

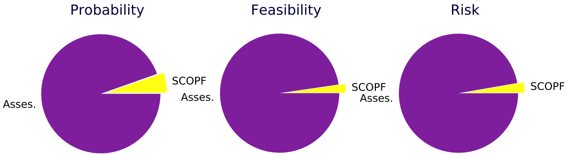
What?

- Practical algorithmic framework for chance-constrained mgmt of operational uncertainties in AC-SCOPF;

Why?

- Post-contingency corrective controls not 100% reliable;
- acknowledging threat explicitly & adopting a tolerance level;
- decision making problem is a slightly more complex variant of the classical AC-SCOPF;
- solution remains understandable & interpretable.

Future work on computational efficiency



- Assessment workload is the computational bottleneck;
- first opportunity to reduce computational times is parallellization;
- the more interesting is machine learning: predicting the objective of the single-contingency OPF problems.

Taking the story forward

How to apply this in practice?

- data collection & models;
- integration in operational practices as complex as any SCOPF variant;
- things are happening :)

How to apply this in other time-horizons?

- On-going work in planning *vs* power injection uncertainties.

Implementation details & full results

E. Karangelos and L. Wehenkel,

"An iterative AC-SCOPF approach managing the contingency and corrective control failure uncertainties with a probabilistic guarantee",

in *IEEE Transactions on Power Systems*, vol. 34, no. 5, pp. 3780-3790, Sept. 2019.

http://hdl.handle.net/2268/233474

Thanks for your attention

e.karangelos@uliege.be

Acknowledgment
The research leading to these results has received funding from the European Union Seventh Framework Programme (FP7/2007-2013) under grant agreement No 608540, project acronym GARPUR (www.garpur-project.eu/). The scientific responsibility lies with the authors. The authors thank the Isaac Newton Institute for Mathematical Sciences, University of Cambridge, for support and hospitality during the programme Mathematics of Energy Systems where work on this paper was completed. This work was supported by EPSRC grant no EP/R014604/1. LW acknowledges the support of F.R.S.-FNRS for his sabbatical year and the Simons Foundation for his fellowship during his stay at the Isaac Newton Institute.