A Parallelization Algorithm for Adequacy Assessment of the Electrical Grid

Gabor Riccardi

Università di Pavia

16/02/24

European Commission

Joint Research Centre

8th AIROYoung WORKSHOP
February 14-16, 2024
UNIVERSITY OF CALABRIA
"Emerging technologies for decision support systems and innovative optimization paradigms"

Stochastic Capacity Expansion Problem (CEP)

We formulate the Stochastic Capacity Expansion Problem as a two-stage stochastic program.

Stochastic Capacity Expansion Problem (CEP)

We formulate the Stochastic Capacity Expansion Problem as a two-stage stochastic program.

$$\begin{aligned} &\min_{x} \; c'x + \mathbb{E}_{\omega}\left[\mathcal{V}(x,\omega)\right] \\ &s.t. \; 0 \leq x_{n,g} \leq X_{n,g} \end{aligned} \tag{CEP}$$

Stochastic Capacity Expansion Problem (CEP)

We formulate the Stochastic Capacity Expansion Problem as a two-stage stochastic program.

$$\begin{aligned} &\min_{x} c'x + \mathbb{E}_{\omega}\left[\mathcal{V}(x,\omega)\right] \\ &s.t.\ 0 \leq x_{n,g} \leq X_{n,g} \end{aligned} \tag{CEP}$$

- The first stage determines the capacity expansion $x_{n,g}$ for each generator $g \in \mathcal{G}$

Stochastic Capacity Expansion Problem (CEP)

We formulate the Stochastic Capacity Expansion Problem as a two-stage stochastic program.

$$
\begin{aligned}
&\min_{x} \; c'x + \mathbb{E}_{\omega}\left[\mathcal{V}(x,\omega)\right] \\
&s.t. \; 0 \leq x_{n,g} \leq X_{n,g}
\end{aligned}
\tag{CEP}
$$

- The first stage determines the capacity expansion $x_{n,g}$ for each generator $g \in \mathcal{G}$
- The second stage solves the Economic Dispatch (ED).

Stochastic Capacity Expansion Problem (CEP)

We formulate the Stochastic Capacity Expansion Problem as a two-stage stochastic program.

$$\begin{aligned} &\min_{x} c'x + \mathbb{E}_{\omega}\left[\mathcal{V}(x,\omega)\right] \\ &s.t.\ 0 \leq x_{n,g} \leq X_{n,g} \end{aligned} \tag{CEP}$$

- The first stage determines the capacity expansion $x_{n,g}$ for each generator $g \in \mathcal{G}$
- The second stage solves the Economic Dispatch (ED).

Where $\mathcal{V}(x,\omega)$ is the solution to (ED) in function of the expanded capacities x and the scenario ω.

Economic Dispatch (ED) model

Economic Dispatch (ED) model Scary Slide

Economic Dispatch (ED) model

For a fixed scenarios $\omega = (\mathcal{PV}, \mathcal{W}, \mathcal{D})$ comprising of respectively solar power, wind power and loads,

Economic Dispatch (ED) model

For a fixed scenarios $\omega = (\mathcal{PV}, \mathcal{W}, \mathcal{D})$ comprising of respectively solar power, wind power and loads, let $y_\omega = (p_\omega, f_\omega, ls_\omega)'$ be the vector containing the power genaration, power flows and line shedding variables.

Economic Dispatch (ED) model

For a fixed scenarios $\omega = (\mathcal{PV}, \mathcal{W}, \mathcal{D})$ comprising of respectively solar power, wind power and loads, let $y_\omega = (p_\omega, f_\omega, ls_\omega)'$ be the vector containing the power genaration, power flows and line shedding variables.

$$\min_y \; q' y_\omega \tag{1}$$

$$s.t. \; p_{n,g,t,\omega} + bd_{n,t,\omega} + \sum_{l \in \mathcal{L}(n)} f_{n,l,t,\omega} + ls_{n,t,\omega} + \mathcal{PV}_{n,t,\omega} + \mathcal{W}_{n,t,\omega} = \tag{2}$$

$$= \mathcal{D}_{n,t,\omega} + ps_{nt.\omega} + bc_{n,t,\omega}$$

$$v_{n,t,\omega} = v_{n,t-1,\omega} + BCE \cdot bc_{n,t,\omega} - BDE \cdot bd_{n,t,\omega} + A_{n,t,\omega} \tag{3}$$

$$(v_{n,t,\omega}, bc_{n,t,\omega}, bd_{n,t,\omega}) \leq (BV, BC, BD) \tag{4}$$

$$p_{n,g,t,\omega} \leq p_{n,g}^{\max} + x_{n,g} \tag{5}$$

$$L_{n,l}^{\min} \leq f_{n,l,t,\omega} \leq L_{n,l}^{\max} \tag{6}$$

Where $v_{n,t,w}$ is the power stored at bus n at time t.

Model relaxation description: Intermediate Economic Dispatches (ED-k)

- We divide the time horizon into K intervals, $\{t_0 = 0 := 1, \ldots, t_1\}, \{t_1 + 1, \ldots, t_2\}, \ldots, \{t_{K-1} + 1, \ldots, t_K := T\}$

Model relaxation description: Intermediate Economic Dispatches (ED-k)

- We divide the time horizon into K intervals, $\{t_0 = 0 := 1, \ldots, t_1\}, \{t_1 + 1, \ldots, t_2\}, \ldots, \{t_{K-1} + 1, \ldots, t_K := T\}$
- We fix a priori the intermediate storage values v_{t_k} for $k = 1, \ldots, K$.

Model relaxation description: Intermediate Economic Dispatches (ED-k)

- We divide the time horizon into K intervals, $\{t_0 = 0 := 1, \ldots, t_1\}, \{t_1 + 1, \ldots, t_2\}, \ldots, \{t_{K-1} + 1, \ldots, t_K := T\}$
- We fix a priori the intermediate storage values v_{t_k} for $k = 1, \ldots, K$.
- We refer to the (ED) problems restricted to each time interval with fixed initial and final storage values as **(ED-k)** and to its optimal values as and with optimal value $\mathcal{V}_{\mathbf{k}}(\mathbf{x}, \mathbf{v}_{\mathbf{t_k}}, \mathbf{v}_{\mathbf{t_{k+1}},\omega})$

Model relaxation description: Intermediate Economic Dispatches (ED-k)

- We divide the time horizon into K intervals, $\{t_0 = 0 := 1, \dots, t_1\}, \{t_1 + 1, \dots, t_2\}, \dots, \{t_{K-1} + 1, \dots, t_K := T\}$
- We fix a priori the intermediate storage values v_{t_k} for $k = 1, \dots, K$.
- We refer to the (ED) problems restricted to each time interval with fixed initial and final storage values as **(ED-k)** and to its optimal values as and with optimal value $\mathcal{V}_{\mathbf{k}}(\mathbf{x}, \mathbf{v}_{\mathbf{t_k}}, \mathbf{v}_{\mathbf{t_{k+1}},\omega})$

Observation

$$\mathcal{V}(x, \omega) = \min_{\{v_{t_k}\}_{k=1}^{K}} \sum_{k=0}^{K-1} \mathcal{V}_k(x, v_{t_k}, v_{t_k+1}, \omega) \tag{7}$$

Model relaxation description: Lower Approximation of (ED)

Since each function $\mathcal{V}_k$ is piecewise linear convex in $x, v_{t_K}, v_{t_{K+1}}$, it can be approximated by a collection of supporting hyperplanes $\{\pi^w_{i,k}(x, v_{t_k}, v_{t_{k+1}})\}$ of each $\mathcal{V}_k$.

Model relaxation description: Lower Approximation of (ED)

Since each function $\mathcal{V}_k$ is piecewise linear convex in $x, v_{t_K}, v_{t_{K+1}}$, it can be approximated by a collection of supporting hyperplanes $\{\pi^w_{i,k}(x, v_{t_k}, v_{t_{k+1}})\}$ of each $\mathcal{V}_k$.
An approximation of (ED) is given by:

Model relaxation description: Lower Approximation of (ED)

Since each function $\mathcal{V}_k$ is piecewise linear convex in $x, v_{t_K}, v_{t_{K+1}}$, it can be approximated by a collection of supporting hyperplanes $\{\pi^w_{i,k}(x, v_{t_k}, v_{t_{k+1}})\}$ of each $\mathcal{V}_k$.
An approximation of (ED) is given by:

$$
\begin{aligned}
\hat{\mathcal{V}}(x,\omega) = \min_{\{v_{t_k}\}_{k=1}^K} \sum_{k=0}^{K} \hat{\mathcal{V}}_k(x, v_{t_k}, v_{t_{k+1}}) = \\
= \min_{\{v_{t_k}\}_{k=1}^K} \sum_{k=0}^{K} \theta_k^\omega \\
\text{s.t.} \quad \theta_k^\omega \geq \pi^\omega_{i,k}(x, v_{t_k}, v_{t_{k+1}}) \quad \forall i, k
\end{aligned}
\tag{ISP}
$$

We refer to this problem as the **Intermediate Storage Problem (ISP)**

Model relaxation description: Lower Approximation of (ED)

Since each function $\mathcal{V}_k$ is piecewise linear convex in $x, v_{t_K}, v_{t_{K+1}}$, it can be approximated by a collection of supporting hyperplanes $\{\pi^w_{i,k}(x, v_{t_k}, v_{t_{k+1}})\}$ of each $\mathcal{V}_k$.
An approximation of (ED) is given by:

$$\begin{aligned}
\hat{\mathcal{V}}(x,\omega) = \min_{\{v_{t_k}\}_{k=1}^K} \sum_{k=0}^{K} \hat{\mathcal{V}}_k(x, v_{t_k}, v_{t_{k+1}}) = \\
= \min_{\{v_{t_k}\}_{k=1}^K} \sum_{k=0}^{K} \theta_k^\omega \\
\text{s.t.} \quad \theta_k^\omega \geq \pi^\omega_{i,k}(x, v_{t_k}, v_{t_{k+1}}) \quad \forall i, k
\end{aligned} \tag{ISP}$$

We refer to this problem as the **Intermediate Storage Problem (ISP)**
(I know, very original)

Model description: Relaxed Capacity Expansion(CEP-R)

$$\begin{aligned} &\min_{x} c'x + \mathbb{E}_{\omega}\left[\mathcal{V}(x,\omega)\right] \\ &s.t.\ 0 \leq x_{n,g} \leq X_{n,g} \end{aligned} \tag{CEP}$$

Model description: Relaxed Capacity Expansion(CEP-R)

$$\min_{x} c'x + \mathbb{E}_{\omega}\left[\mathcal{V}(x,\omega)\right]$$
$$s.t.\ 0 \leq x_{n,g} \leq X_{n,g} \tag{CEP}$$

$$\min_{x} c'x + \mathbb{E}_{w}\left[\hat{\mathcal{V}}(x,w)\right]$$
$$s.t.\ 0 \leq x_{n,g} \leq X_{n,g} \tag{CEP-R}$$

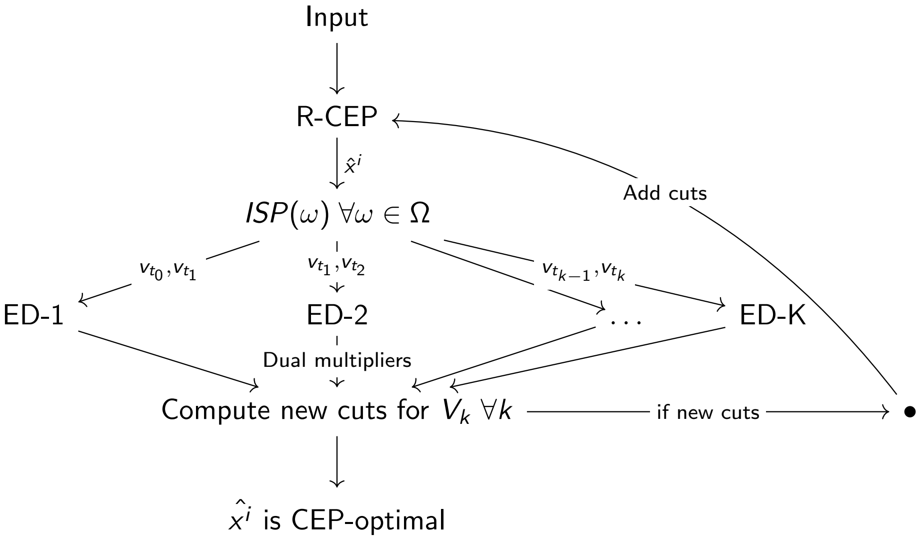
Model description: Relaxed Capacity Expansion(CEP-R)

$$\min_{x} c'x + \mathbb{E}_{\omega}\left[\mathcal{V}(x,\omega)\right]$$
$$s.t.\ 0 \le x_{n,g} \le X_{n,g} \tag{CEP}$$

$$\min_{x} c'x + \mathbb{E}_{w}\left[\hat{\mathcal{V}}(x,w)\right]$$
$$s.t.\ 0 \le x_{n,g} \le X_{n,g} \tag{CEP-R}$$

Since calculating $\hat{\mathcal{V}}$ is straightforward, solving (CEP-R) can be done efficiently with L-shaped or subgradient schemes.

Algorithm


Input
R-CEP
$\hat{x}^i$
$ISP(\omega)\ \forall \omega \in \Omega$
Add cuts
v_{t_0}, v_{t_1}
v_{t_1}, v_{t_2}
$v_{t_{k-1}}, v_{t_k}$
ED-1
ED-2
...
ED-K
Dual multipliers
Compute new cuts for $V_k\ \forall k$
if new cuts
$\hat{x}^i$ is CEP-optimal

Convergence results 2

- Since $(CEP - R) \leq (CEP)$ if a $(CEP - R)$ optimal solution has the same cost for (CEP) then it's also (CEP)-optimal.

Convergence results 2

- Since $(CEP - R) \leq (CEP)$ if a $(CEP - R)$ optimal solution has the same cost for (CEP) then it's also (CEP)-optimal.

Remark 1: It is sufficient to prove that after a finite number of steps (i) of the algorithm we have:

$$\hat{\mathcal{V}}(\hat{x}^i, \omega) = \mathcal{V}(\hat{x}^i, \omega) \text{ for all } \omega \in \Omega \tag{8}$$

Convergence results 3

Observation

After a finite number of iterations no new cuts are found for $\mathcal{V}_k$*.*

Proof.

$$\tag{9}$$

Convergence results 3

Observation

After a finite number of iterations no new cuts are found for $\mathcal{V}_k$*.*

Proof.

$$\#\{p \mid p \text{ is a normal vector of a supporting hyperplane of } \mathcal{V}_k\} \leq \tag{9}$$

Convergence results 3

Observation

After a finite number of iterations no new cuts are found for $\mathcal{V}_k$.

Proof.

$$\#\{p \mid p \text{ is a normal vector of a supporting hyperplane of } \mathcal{V}_k\} \leq$$
$$\#\{\text{dual solutions } p = q' B^{-1} \text{ of (ED-k) for varying } x, v_{t_k}, v_{t_{k+1}}\} \leq \tag{9}$$

Convergence results 3

Observation

After a finite number of iterations no new cuts are found for $\mathcal{V}_k$.

Proof.

$$
\begin{aligned}
&\#\{p \mid p \text{ is a normal vector of a supporting hyperplane of } \mathcal{V}_k\} \leq \\
&\#\{\text{dual solutions } p = q'B^{-1} \text{ of (ED-k) for varying } x, v_{t_k}, v_{t_{k+1}}\} \leq \\
&\#\{\text{basis matrices of (ED-k)}\} < \infty \qquad (9)
\end{aligned}
$$

Convergence results 3

Observation

After a finite number of iterations no new cuts are found for $\mathcal{V}_k$.

Proof.

$$
\begin{aligned}
&\#\{p \mid p \text{ is a normal vector of a supporting hyperplane of } \mathcal{V}_k\} \leq \\
&\#\{\text{dual solutions } p = q' B^{-1} \text{ of (ED-k) for varying } x, v_{t_k}, v_{t_{k+1}}\} \leq \\
&\#\{\text{basis matrices of (ED-k)}\} < \infty
\end{aligned} \tag{9}
$$

After a finite number of steps:

Convergence results 3

Observation

After a finite number of iterations no new cuts are found for $\mathcal{V}_k$.

Proof.

$$
\begin{aligned}
&\#\{p \mid p \text{ is a normal vector of a supporting hyperplane of } \mathcal{V}_k\} \leq \\
&\#\{\text{dual solutions } p = q' B^{-1} \text{ of (ED-k) for varying } x, v_{t_k}, v_{t_{k+1}}\} \leq \\
&\#\{\text{basis matrices of (ED-k)}\} < \infty
\end{aligned} \tag{9}
$$

After a finite number of steps:

- new cut: $\bar{c}(x, v) = p'(x, v) + b$

Convergence results 3

Observation

After a finite number of iterations no new cuts are found for $\mathcal{V}_k$.

Proof.

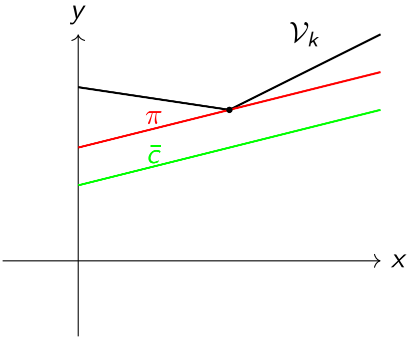
$$
\begin{aligned}
&\#\{p \mid p \text{ is a normal vector of a supporting hyperplane of } \mathcal{V}_k\} \leq \\
&\#\{\text{dual solutions } p = q'B^{-1} \text{ of (ED-k) for varying } x, v_{t_k}, v_{t_{k+1}}\} \leq \\
&\#\{\text{basis matrices of (ED-k)}\} < \infty
\end{aligned} \tag{9}
$$

After a finite number of steps:

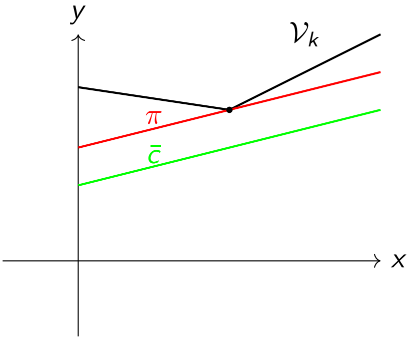
- new cut: $\bar{c}(x, v) = p'(x, v) + b$
- an old cut: $\pi(x, v) = p'(x, v) + \bar{b}$

$\square$

Convergence results 3



Convergence results 3



Since both are supporting hyperplanes it follows that $b = \bar{b}$ (and therefore $\bar{c}$ is not a new cut).

Convergence results 4

Observation

If after the i-iteration no new cuts are added for some i and k then
$\hat{\mathcal{V}}_k(\hat{x}^i, \hat{v}_k, \hat{v}_{k+1}) = \mathcal{V}_k(\hat{x}^i, \hat{v}_k, \hat{v}_{k+1})$.

Convergence results 4

Observation

If after the i-iteration no new cuts are added for some i and k then $\hat{\mathcal{V}}_k(\hat{x}^i, \hat{v}_k, \hat{v}_{k+1}) = \mathcal{V}_k(\hat{x}^i, \hat{v}_k, \hat{v}_{k+1})$.
Where $\hat{\mathcal{V}}_k(\hat{x}^i, \hat{v}_{t_k}, \hat{v}_{t_{k+1}}, \omega) = \max_i \pi^{\omega}_{i,k}(\hat{x}^i, \hat{v}_{t_k}, \hat{v}_{t_{k+1}})$ *be the current supporting hyperplane approximation of* $\mathcal{V}_k$.

Convergence results 4

Observation

If after the i-iteration no new cuts are added for some i and k then $\hat{\mathcal{V}}_k(\hat{x}^i, \hat{v}_k, \hat{v}_{k+1}) = \mathcal{V}_k(\hat{x}^i, \hat{v}_k, \hat{v}_{k+1})$.
Where $\hat{\mathcal{V}}_k(\hat{x}^i, \hat{v}_{t_k}, \hat{v}_{t_{k+1}}, \omega) = \max_i \pi^{\omega}_{i,k}(\hat{x}^i, \hat{v}_{t_k}, \hat{v}_{t_{k+1}})$ *be the current supporting hyperplane approximation of* $\mathcal{V}_k$.

Proof.

Let $\bar{c}^{\omega}_k(x, v_{t_k}) := p'(x - \hat{x}^i, v_{t_k} - \hat{v}_{t_k}) + V_k(\hat{x}^i, \hat{v}_{t_k})$ be the new cut found after the i-th iteration.

Convergence results 4

Observation

If after the i-iteration no new cuts are added for some i and k then $\hat{\mathcal{V}}_k(\hat{x}^i, \hat{v}_k, \hat{v}_{k+1}) = \mathcal{V}_k(\hat{x}^i, \hat{v}_k, \hat{v}_{k+1})$.
Where $\hat{\mathcal{V}}_k(\hat{x}^i, \hat{v}_{t_k}, \hat{v}_{t_{k+1}}, \omega) = \max_i \pi^{\omega}_{i,k}(\hat{x}^i, \hat{v}_{t_k}, \hat{v}_{t_{k+1}})$ *be the current supporting hyperplane approximation of* $\mathcal{V}_k$.

Proof.

Let $\bar{c}^{\omega}_k(x, v_{t_k}) := p'(x - \hat{x}^i, v_{t_k} - \hat{v}_{t_k}) + V_k(\hat{x}^i, \hat{v}_{t_k})$ be the new cut found after the i-th iteration.
Since $\bar{c}$ is not a new cut we have $\bar{c}(\hat{x}^i, v_{t_k}) \leq \hat{\mathcal{V}}_k(\hat{x}^i, v_{t_k})$.

Convergence results 4

Observation

If after the i-iteration no new cuts are added for some i and k then $\hat{\mathcal{V}}_k(\hat{x}^i, \hat{v}_k, \hat{v}_{k+1}) = \mathcal{V}_k(\hat{x}^i, \hat{v}_k, \hat{v}_{k+1})$.
Where $\hat{\mathcal{V}}_k(\hat{x}^i, \hat{v}_{t_k}, \hat{v}_{t_{k+1}}, \omega) = \max_i \pi^{\omega}_{i,k}(\hat{x}^i, \hat{v}_{t_k}, \hat{v}_{t_{k+1}})$ *be the current supporting hyperplane approximation of* $\mathcal{V}_k$.

Proof.

Let $\bar{c}^{\omega}_k(x, v_{t_k}) := p'(x - \hat{x}^i, v_{t_k} - \hat{v}_{t_k}) + V_k(\hat{x}^i, \hat{v}_{t_k})$ be the new cut found after the i-th iteration.
Since $\bar{c}$ is not a new cut we have $\bar{c}(\hat{x}^i, v_{t_k}) \leq \hat{\mathcal{V}}_k(\hat{x}^i, v_{t_k})$. We have thus

$$\mathcal{V}_k(\hat{x}^i, \hat{v}_{t_k}) \geq \hat{\mathcal{V}}_k(\hat{x}^i, \hat{v}_{t_k}) \geq \bar{c}(\hat{x}^i, \hat{v}_{t_k}) = \mathcal{V}_k(\hat{x}^i, \hat{v}_{t_k})$$

Convergence results 4

Observation

If after the i-iteration no new cuts are added for some i and k then
$\hat{\mathcal{V}}_k(\hat{x}^i, \hat{v}_k, \hat{v}_{k+1}) = \mathcal{V}_k(\hat{x}^i, \hat{v}_k, \hat{v}_{k+1})$.
Where $\hat{\mathcal{V}}_k(\hat{x}^i, \hat{v}_{t_k}, \hat{v}_{t_{k+1}}, \omega) = \max_i \pi^{\omega}_{i,k}(\hat{x}^i, \hat{v}_{t_k}, \hat{v}_{t_{k+1}})$ *be the current supporting hyperplane approximation of* $\mathcal{V}_k$.

Proof.

Let $\bar{c}^{\omega}_k(x, v_{t_k}) := p'(x - \hat{x}^i, v_{t_k} - \hat{v}_{t_k}) + V_k(\hat{x}^i, \hat{v}_{t_k})$ be the new cut found after the i-th iteration.
Since $\bar{c}$ is not a new cut we have $\bar{c}(\hat{x}^i, v_{t_k}) \leq \hat{\mathcal{V}}_k(\hat{x}^i, v_{t_k})$. We have thus

$$\mathcal{V}_k(\hat{x}^i, \hat{v}_{t_k}) \geq \hat{\mathcal{V}}_k(\hat{x}^i, \hat{v}_{t_k}) \geq \bar{c}(\hat{x}^i, \hat{v}_{t_k}) = \mathcal{V}_k(\hat{x}^i, \hat{v}_{t_k})$$

which concludes the proof. □

Convergence results 5

In conclusion, we have $\hat{\mathcal{V}}_k(\hat{x}^i, v_{t_k}, v_{t_{k+1}}, \omega) = \mathcal{V}_k(\hat{x}^i, v_{t_k}, v_{t_{k+1}}, \omega)$ for all ω, k.

Convergence results 5

In conclusion, we have $\hat{\mathcal{V}}_k(\hat{x}^i, v_{t_k}, v_{t_{k+1}}, \omega) = \mathcal{V}_k(\hat{x}^i, v_{t_k}, v_{t_{k+1}}, \omega)$ for all ω, k.
Thus $\hat{\mathcal{V}}(\hat{x}^i, \omega) = \mathcal{V}(\hat{x}^i, \omega)$.

Convergence results 5

In conclusion, we have $\hat{\mathcal{V}}_k(\hat{x}^i, v_{t_k}, v_{t_{k+1}}, \omega) = \mathcal{V}_k(\hat{x}^i, v_{t_k}, v_{t_{k+1}}, \omega)$ for all ω, k.
Thus $\hat{\mathcal{V}}(\hat{x}^i, \omega) = \mathcal{V}(\hat{x}^i, \omega)$.

Proposition

The algorithm converges after a finite number of iterations and $\hat{x}^i$ is an optimal solution for (CEP).

Future implementation in Pypsa

- We are currently implementing this and other stochastic methods within the Pypsa [BHS18] framework using the Linopy [Hof23] modeling package in Python.

Future implementation in Pypsa

- We are currently implementing this and other stochastic methods within the Pypsa [BHS18] framework using the Linopy [Hof23] modeling package in Python.
- We expect improved convergence speed respect to the L-shaped method, especially when leveraging parallel processing capabilities.

Conclusions and future perspectives.

- The specific structure of the intertemporal constraints makes it possible do develop tailored optimization algorithms for (CEP).

Conclusions and future perspectives.

- The specific structure of the intertemporal constraints makes it possible do develop tailored optimization algorithms for (CEP).
- The presented method can be also applied to other linear models such as DC-OPF.

Conclusions and future perspectives.

- The specific structure of the intertemporal constraints makes it possible do develop tailored optimization algorithms for (CEP).
- The presented method can be also applied to other linear models such as DC-OPF.
- Care must be taken for intra-temporal constraints such as ramp up or ramp down constraints.

Conclusions and future perspectives.

- The specific structure of the intertemporal constraints makes it possible do develop tailored optimization algorithms for (CEP).
- The presented method can be also applied to other linear models such as DC-OPF.
- Care must be taken for intra-temporal constraints such as ramp up or ramp down constraints.
- Develop heuristics for the choice of the number of time intervals K.

Conclusions and future perspectives.

- The specific structure of the intertemporal constraints makes it possible do develop tailored optimization algorithms for (CEP).
- The presented method can be also applied to other linear models such as DC-OPF.
- Care must be taken for intra-temporal constraints such as ramp up or ramp down constraints.
- Develop heuristics for the choice of the number of time intervals K.
- Examine generalizations to multistage stochastic programs.

Conclusions and future perspectives.

- The specific structure of the intertemporal constraints makes it possible do develop tailored optimization algorithms for (CEP).
- The presented method can be also applied to other linear models such as DC-OPF.
- Care must be taken for intra-temporal constraints such as ramp up or ramp down constraints.
- Develop heuristics for the choice of the number of time intervals K.
- Examine generalizations to multistage stochastic programs.

Thank you for your attention.

gabor.riccardi01@universitadipavia.it

https://www.compopt.it

Some references:

[Ávi+23] Daniel Ávila, Anthony Papavasiliou, Mauricio Junca, and Lazaros Exizidis. "Applying High-Performance Computing to the European Resource Adequacy Assessment". In: *IEEE Transactions on Power Systems* (2023), pp. 1–13. DOI: 10.1109/TPWRS.2023.3304717.

[Bie+20] Daniel Bienstock, Mauro Escobar, Claudio Gentile, and Leo Liberti. "Mathematical Programming formulations for the Alternating Current Optimal Power Flow problem". In: *4OR* 18.3 (July 2020), pp. 249–292. DOI: 10.1007/s10288-020-00455-w.

[BM14] Daniel Bienstock and Gonzalo Munoz. "On linear relaxations of OPF problems". In: (Nov. 2014).

[BHS18] T. Brown, J. Hörsch, and D. Schlachtberger. "PyPSA: Python for Power System Analysis". In: *Journal of Open Research Software* 6.4 (1 2018). DOI: 10.5334/jors.188. eprint: 1707.09913. URL: https://doi.org/10.5334/jors.188.

[Hof23] Fabian Hofmann. "Linopy: Linear optimization with n-dimensional labeled variables". In: *Journal of Open Source Software* 8.84 (2023), p. 4823. DOI: 10.21105/joss.04823. URL: https://doi.org/10.21105/joss.04823.

Adequacy Assessment of the Electrical Grid

- Measuring the ability of the electric power system to react to adverse uncertain condition has become increasingly importan.
- Member States wishing to introduce capacity mechanisms can do so if an adequacy concern is identified in the ERAA study, a pan-European adequacy assessment for up to 10 years ahead.
- Due to the scale of the ERAA study, ERAA 2022 considered a reduced stochastic problem with three scenarios.
- In [Ávi+23], Daniel A'vila introduced a decomposition algorithm based on subgradient approximations was introduced

Power Grid Optimization

Stochasticity

Time / Exactness

- Optimal Power Flow (OPF) [Bie+20]
  - AC OPF: exact physical model
  - Security-Constrained OPF (SCOPF) – Includes contingencies to guarantee system security under failures.
  - DC OPF and other linearized models [BM14]
  - other relaxations.
- Unit Commitment – Determines on/off status of power units, ignoring grid constraints.
- Economic Dispatch (ED) – Minimizes generation cost, ignoring grid constraints.

Capacity expansion problem: Based on Economic Dispatch models with added flow balance at bus nodes and various scenarios.