Detection and Mitigation of Corrupted Information in Distributed Model Predictive Control Based on Resource Allocation

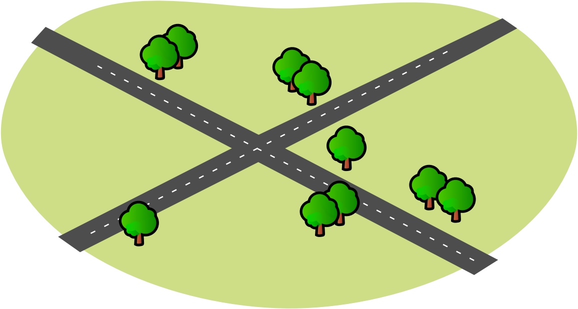
R. A. Nogueira R. Bourdais H. Guéguen
{rafael-accacio.nogueira, romain.bourdais, herve.gueguen} at centralesupelec.fr

AUT Department
IETR — CentraleSupélec

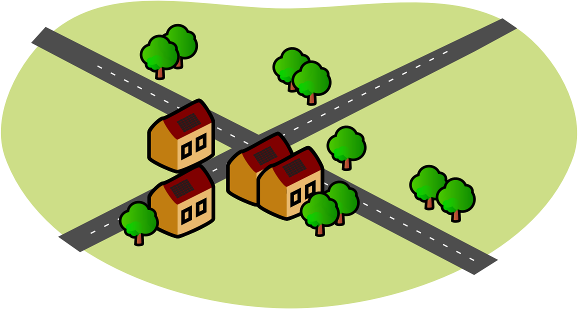
5th International Conference on Control and Fault-Tolerant Systems, 2021

https://git.io/JEFGW

Context



Context



Context

Context

Context

- Geographically distributed
- Coupled by constraints (energy)
- Optimization objectives
  - Energy
  - User satisfaction
  - ...
- Solution → Model Predictive Control

CentraleSupélec

Context

- Geographically distributed
- Coupled by constraints (energy)
- Optimization objectives
  - Energy
  - User satisfaction
  - ...
- Solution → Model Predictive Control

CentraleSupélec

Context

- Geographically distributed
- Coupled by constraints (energy)
- Optimization objectives
  - Energy
  - User satisfaction
  - ...
- Solution → Model Predictive Control

CentraleSupélec

Context

- Geographically distributed
- Coupled by constraints (energy)
- Optimization objectives
  - Energy
  - User satisfaction
  - ...
- Solution → Model Predictive Control

CentraleSupélec

- Geographically distributed
- Coupled by constraints (energy)
- Optimization objectives
  - Energy
  - User satisfaction
  - ...
- Solution → Model Predictive Control

CentraleSupélec

Model Predictive Control

Find control input sequence that optimizes an objective function

$$
\begin{aligned}
&\underset{\boldsymbol{u}(k:k+N_p-1|k)}{\text{minimize}} && \overbrace{\sum_{j=1}^{N_p} \|\boldsymbol{v}(k+j|k)\|_Q^2 + \|\boldsymbol{u}(k+j-1|k)\|_R^2}^{J_G(k)} \\
&\text{subject to} && \left.\begin{array}{c} \boldsymbol{x}(k+1) = f(\boldsymbol{x}(k), \boldsymbol{u}(k)) \\ g_i(\boldsymbol{x}(k), \boldsymbol{u}(k)) = 0 \\ h_k(\boldsymbol{x}(k), \boldsymbol{u}(k)) \leq 0 \end{array}\right\} \begin{array}{l} \forall i \in \{1, \ldots, N_p\} \\ \forall j \in \{1, \ldots, N_p\} \\ \forall k \in \{1, \ldots, N_p\} \end{array}
\end{aligned}
$$

Using

CentraleSupélec

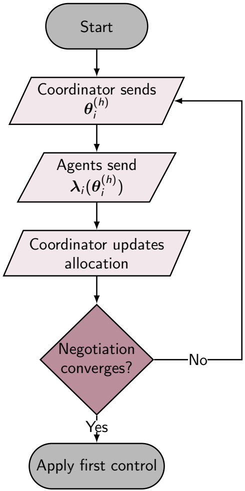


Figure 1: Quantity decomposition based DMPC

CentraleSupélec

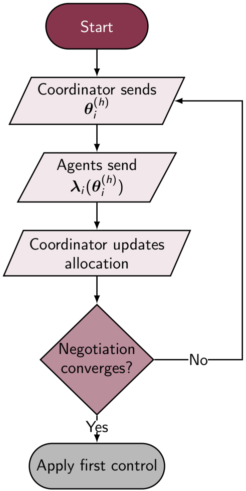


Figure 1: Quantity decomposition based DMPC

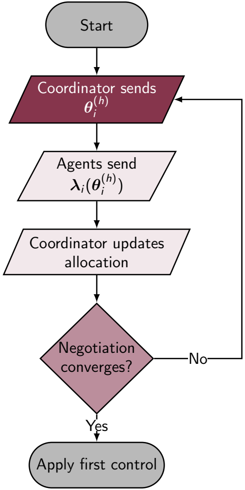


Figure 1: Quantity decomposition based DMPC

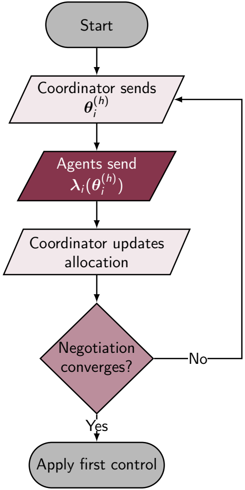


Figure 1: Quantity decomposition based DMPC

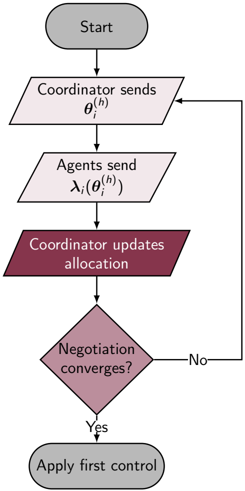


Figure 1: Quantity decomposition based DMPC

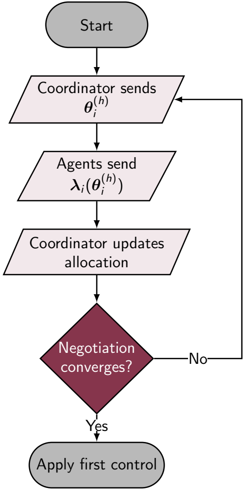


Figure 1: Quantity decomposition based DMPC

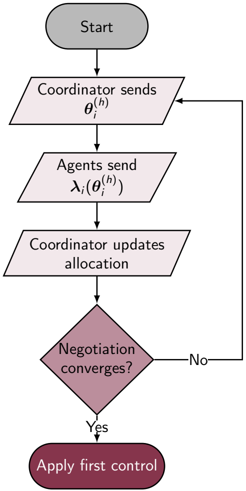


Figure 1: Quantity decomposition based DMPC

CentraleSupélec

Outline

Outline

CentraleSupélec

Make Titles Informative. Use Uppercase Letters.

Subtitles are optional.

- Use `itemize` a lot.
- Use very short sentences or short phrases.

Make Titles Informative.

te

You can create overlays...

- using the pause command:
  - First item.
  - Second item.
- using overlay specifications:
  - First item.
  - Second item.
- using the general uncover command:
  - First item.
  - Second item.

CentraleSupélec

Make Titles Informative.

te

You can create overlays...

- using the pause command:
  - First item.
  - Second item.
- using overlay specifications:
  - First item.
  - Second item.
- using the general uncover command:
  - First item.
  - Second item.

Make Titles Informative.

te

You can create overlays. . .

- using the `pause` command:
  - First item.
  - Second item.
- using overlay specifications:
  - First item.
  - Second item.
- using the general `uncover` command:
  - First item.
  - Second item.

Make Titles Informative.

te

You can create overlays...

- using the `pause` command:
  - First item.
  - Second item.
- using overlay specifications:
  - First item.
  - Second item.
- using the general `uncover` command:
  - First item.
  - Second item.

Make Titles Informative.

te

You can create overlays. . .

- using the `pause` command:
  - First item.
  - Second item.
- using overlay specifications:
  - First item.
  - Second item.
- using the general `uncover` command:
  - First item.
  - Second item.

Make Titles Informative.

te

You can create overlays. . .

- using the `pause` command:
  - First item.
  - Second item.
- using overlay specifications:
  - First item.
  - Second item.
- using the general `uncover` command:
  - First item.
  - Second item.

CentraleSupélec

Outline

1. Vulnerabilities in distributed MPC based on Resource Allocation
   - The Basic Problem That We Studied
   - Previous Work
2. Securing the DMPC
3. Our Results/Contribution
   - Main Results
   - Basic Ideas for Proofs/Implementation

CentraleSupélec

Make Titles Informative.

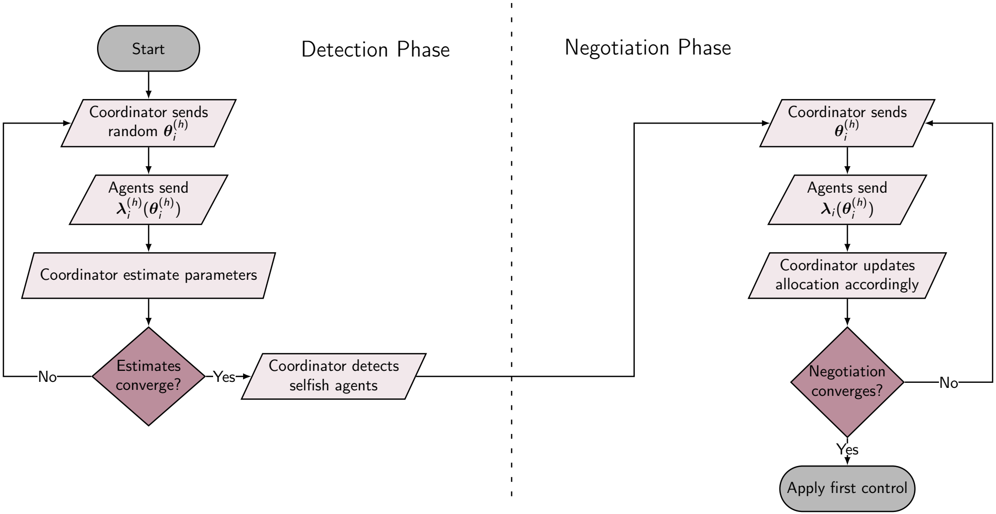


Figure 2: Secure DMPC

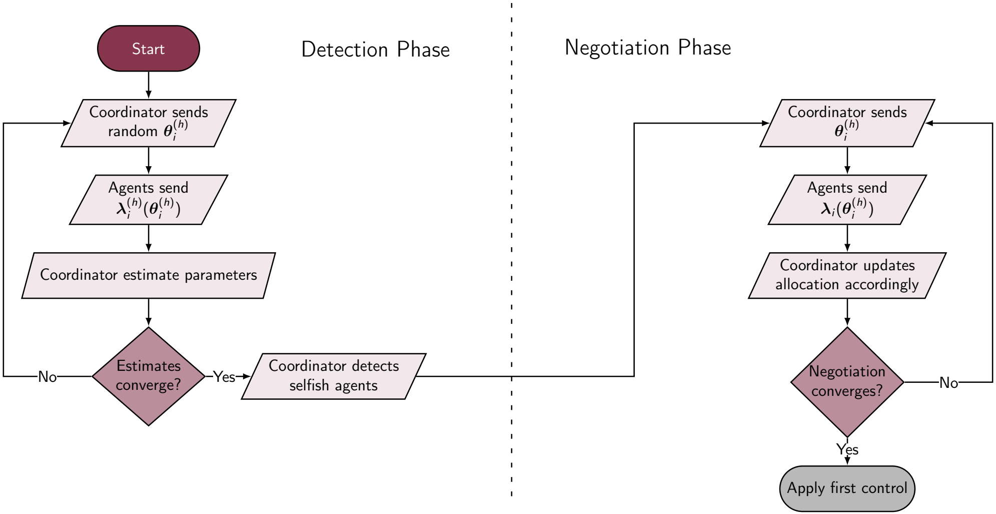


Figure 2: Secure DMPC

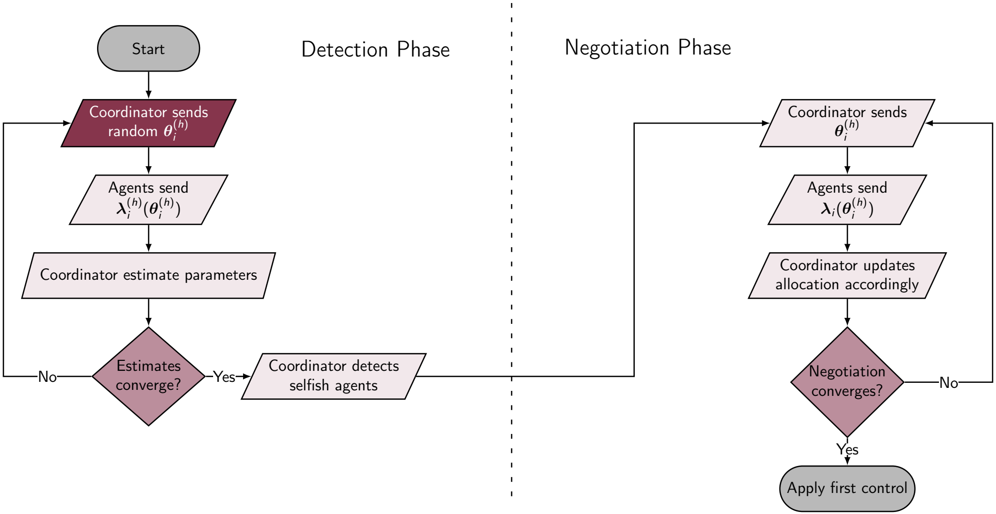


Figure 2: Secure DMPC

CentraleSupélec

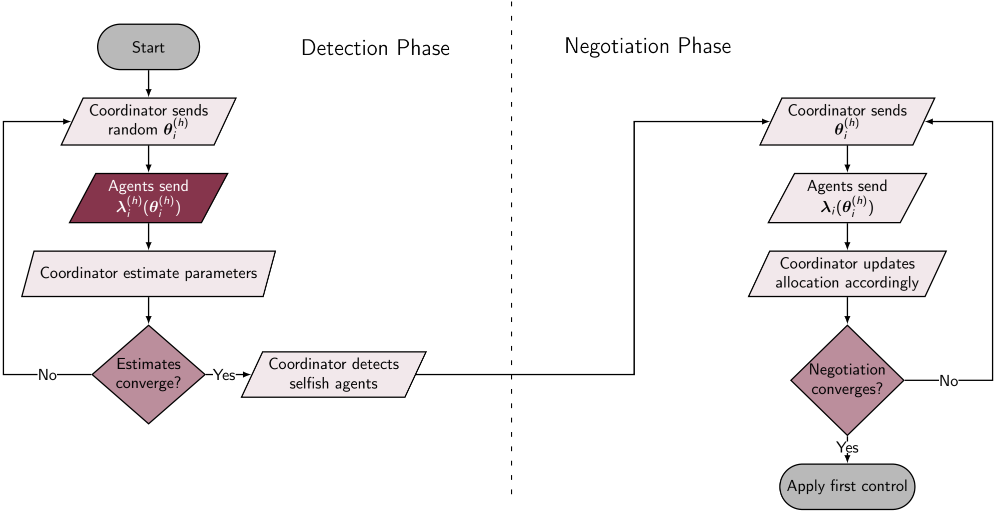


Figure 2: Secure DMPC

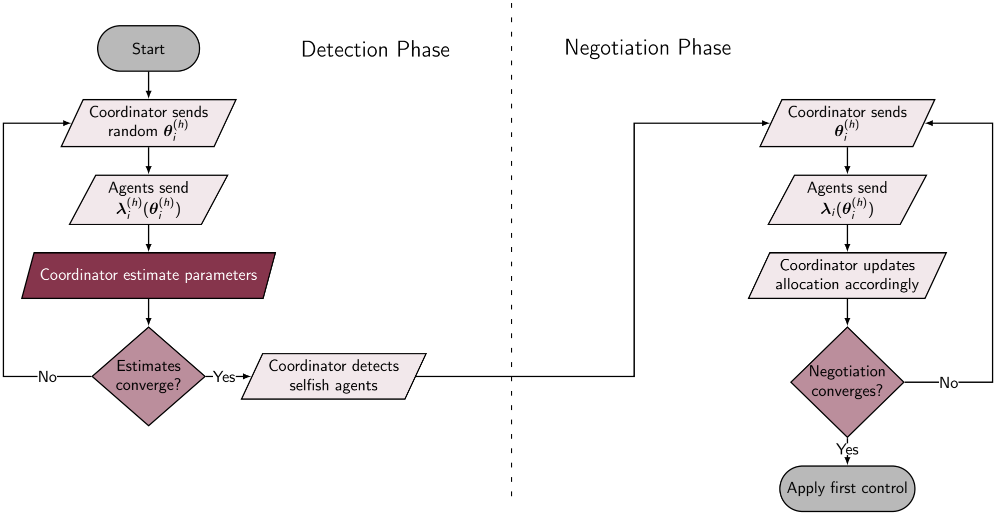


Figure 2: Secure DMPC

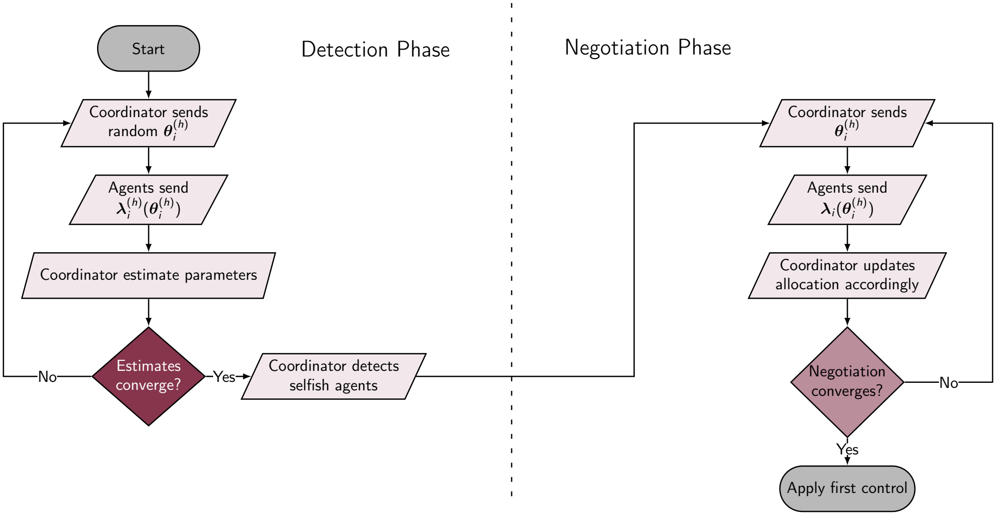


Figure 2: Secure DMPC

CentraleSupélec

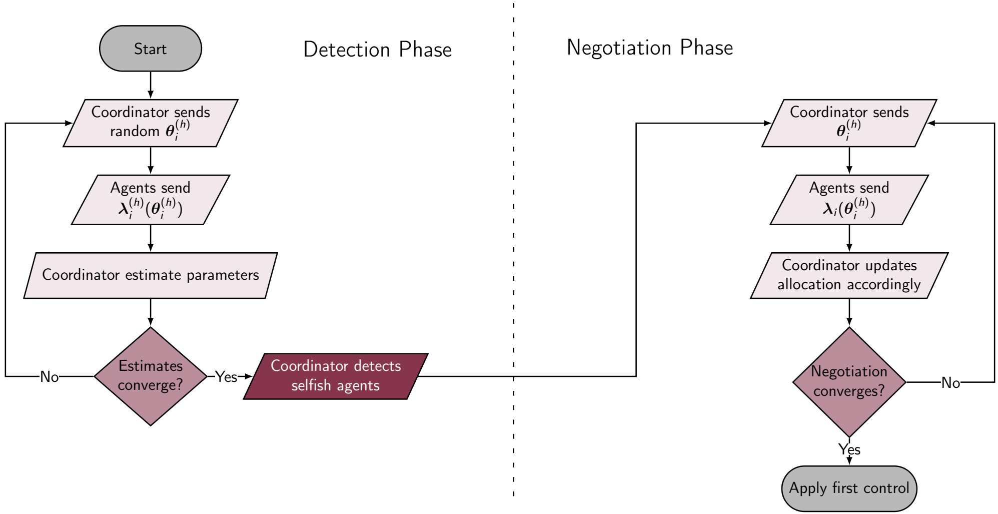


Figure 2: Secure DMPC

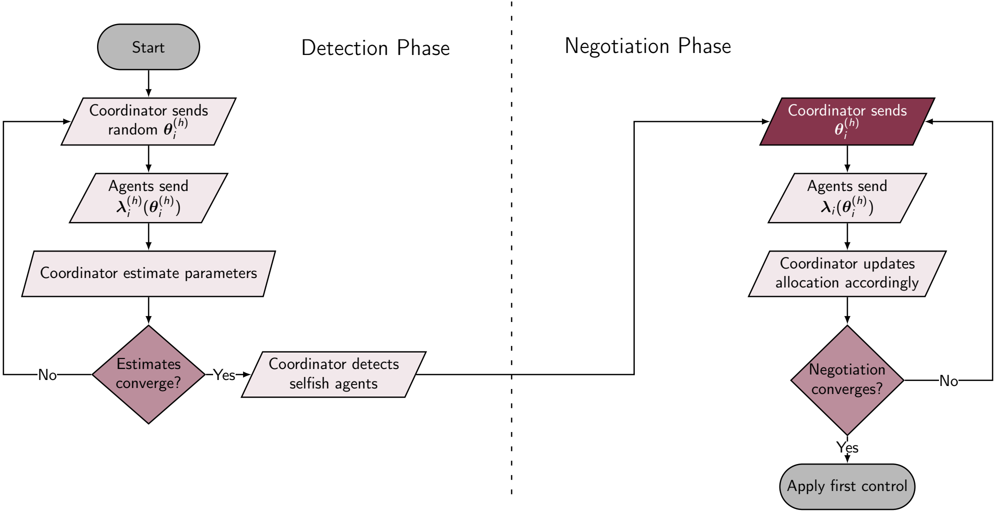


Figure 2: Secure DMPC

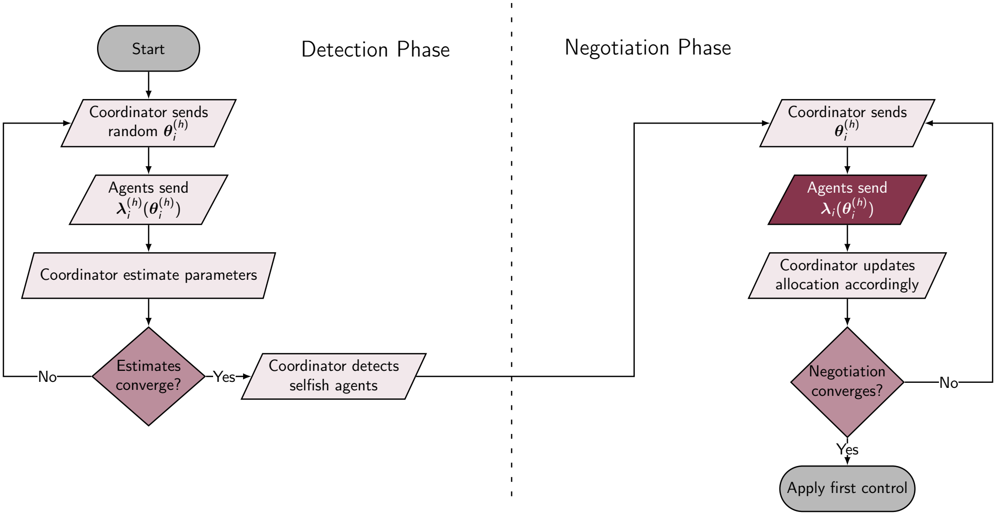


Figure 2: Secure DMPC

CentraleSupélec

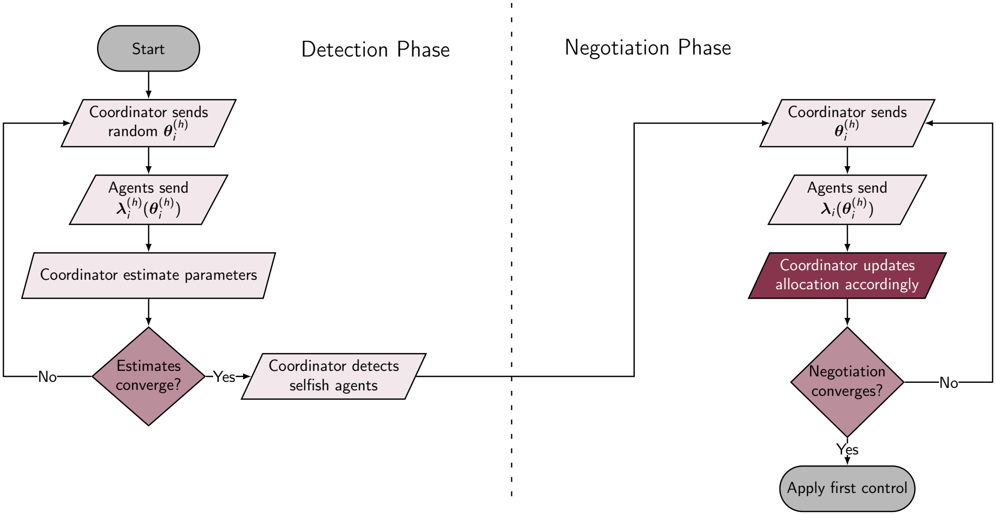


Figure 2: Secure DMPC

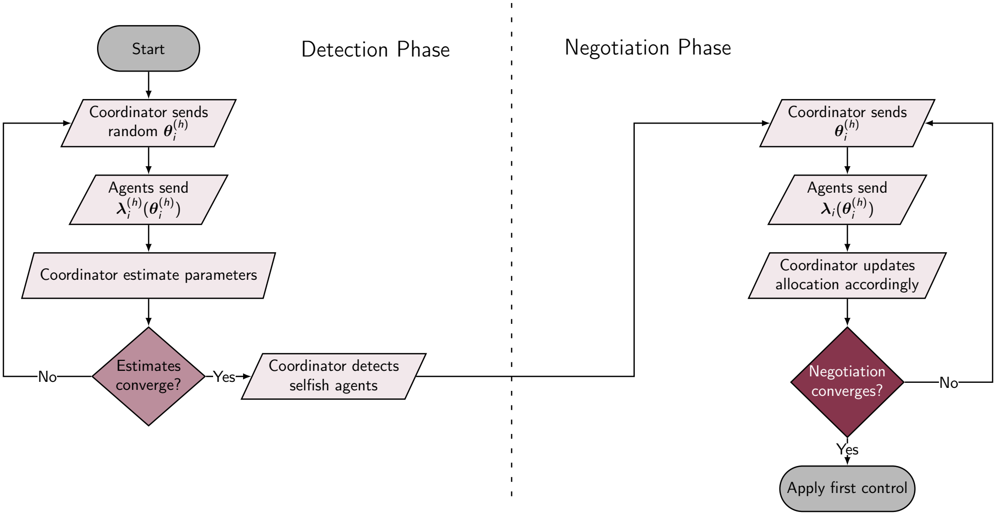


Figure 2: Secure DMPC

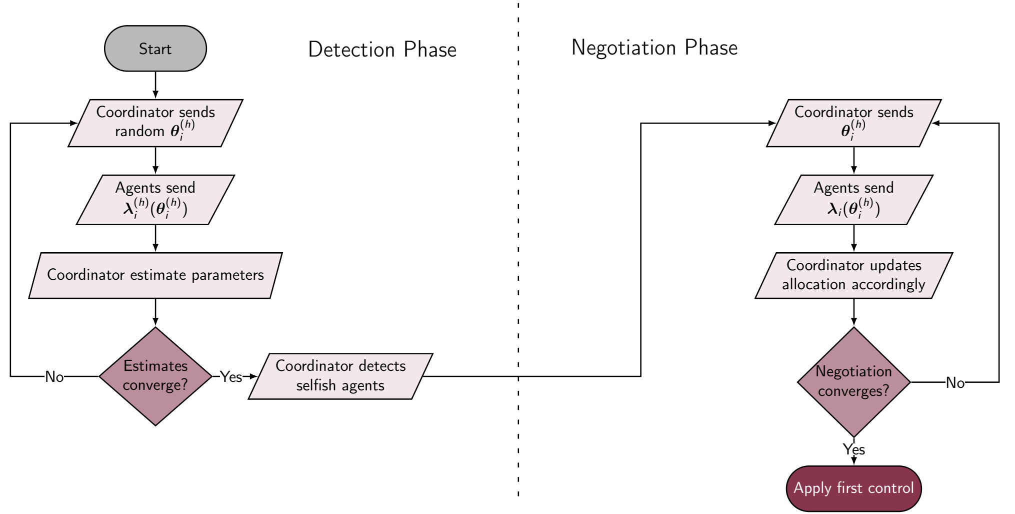


Figure 2: Secure DMPC

Outline

Make Titles Informative.

Outline

Make Titles Informative.

Summary

1. Resource allocation based DMPC is vulnerable to attacks.
2. Sub-problems structure has time invariant parameters.
3. Attacks can be detected using these parameters.

- Outlook
  - Inequality Constraints yield Hybrid behavior
  - Non-linear attack model

CentraleSupélec

Summary

1. Resource allocation based DMPC is vulnerable to attacks.
2. Sub-problems structure has time invariant parameters.
3. Attacks can be detected using these parameters.

- Outlook
  - Inequality Constraints yield Hybrid behavior
  - Non-linear attack model

CentraleSupélec

Summary

1. Resource allocation based DMPC is vulnerable to attacks.
2. Sub-problems structure has time invariant parameters.
3. Attacks can be detected using these parameters.

- Outlook
  - Inequality Constraints yield Hybrid behavior
  - Non-linear attack model

CentraleSupélec

Summary

1. Resource allocation based DMPC is vulnerable to attacks.
2. Sub-problems structure has time invariant parameters.
3. Attacks can be detected using these parameters.

- Outlook
  - Inequality Constraints yield Hybrid behavior
  - Non-linear attack model

CentraleSupélec

Summary

1. Resource allocation based DMPC is vulnerable to attacks.
2. Sub-problems structure has time invariant parameters.
3. Attacks can be detected using these parameters.

- Outlook
  - Inequality Constraints yield Hybrid behavior
  - Non-linear attack model

CentraleSupélec

Summary

1. Resource allocation based DMPC is vulnerable to attacks.
2. Sub-problems structure has time invariant parameters.
3. Attacks can be detected using these parameters.

- Outlook
  - Inequality Constraints yield Hybrid behavior
  - Non-linear attack model

CentraleSupélec

Summary

1. Resource allocation based DMPC is vulnerable to attacks.
2. Sub-problems structure has time invariant parameters.
3. Attacks can be detected using these parameters.

- Outlook
  - Inequality Constraints yield Hybrid behavior
  - Non-linear attack model

CentraleSupélec

For Further Reading I

J. M. Maestre, R. R. Negenborn *et al.*
Distributed Model Predictive Control made easy.
Springer, 2014, vol. 69.

P. Velarde, J. M. Maestre, H. Ishii, and R. R. Negenborn,
"Scenario-based defense mechanism for distributed model predictive control,"
2017 IEEE 56th Annual Conference on Decision and Control (CDC). IEEE, Dec 2017, pp. 6171–6176.

S. Someone.
On this and that.
Journal of This and That, 2(1):50–100, 2000.

Questions?

Repository
https://github.com/Accacio/SysTol-21

Contact
rafael-accacio.nogueira@centralesupelec.fr