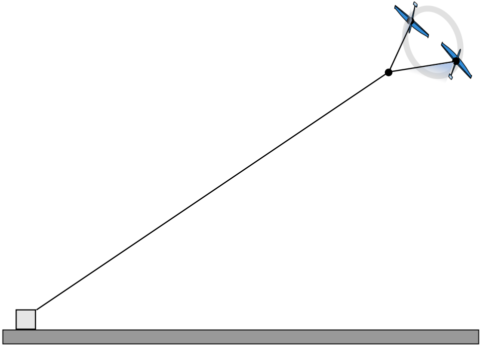
Optimization of Long Trajectories of Dual-Wing AWE Systems with Many Cycles

Jakob Harzer, Jochem De Schutter, Per Rutquist and Moritz Diehl

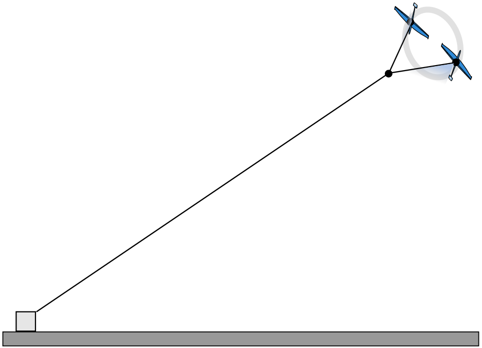
University of Freiburg

AWEC, April 26, 2024

Motivation

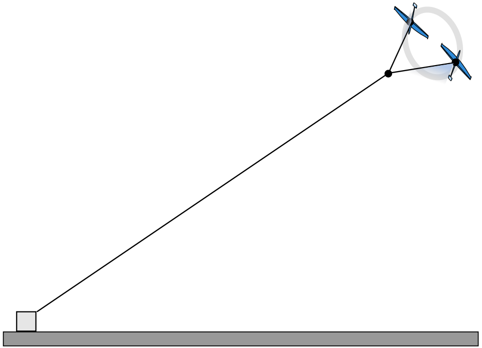


Motivation



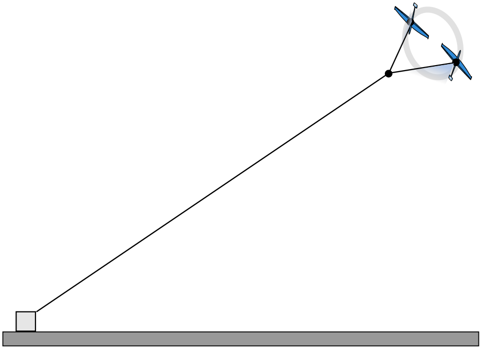
- Optimize a single pumping cycle of a dual-kite AWE system

Motivation



- Optimize a single pumping cycle of a dual-kite AWE system
- Formulate OCP

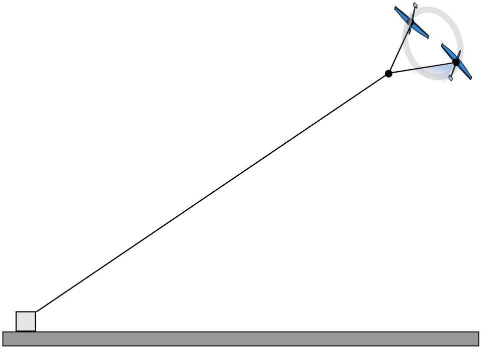
Motivation



- Optimize a single pumping cycle of a dual-kite AWE system
- Formulate OCP

$$\begin{aligned}
\max_{\substack{x(\cdot),u(\cdot),\\ t_\mathrm{f}}} \quad & P_\mathrm{gen}(x,u,t_\mathrm{f}) \\
\text{s.t.} \quad & \dot{x} = f(x(t),u(t)), \quad \forall t \in [0,t_\mathrm{f}], \\
& 0 \leq h(x(t),u(t)), \quad \forall t \in [0,t_\mathrm{f}]
\end{aligned}$$

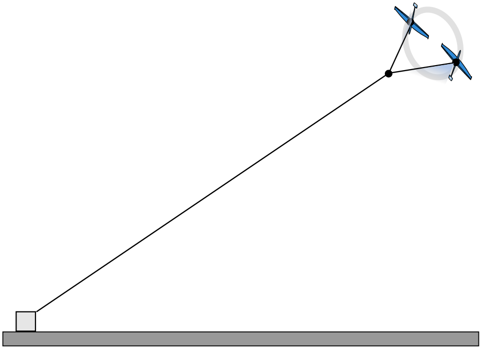
Motivation



- Optimize a single pumping cycle of a dual-kite AWE system
- Formulate OCP $\rightarrow$ discretize to NLP

$$\begin{aligned} \min_{w} \quad & F(w) \\ \text{s.t.} \quad & 0 = G(w), \\ & 0 \leq H(w) \end{aligned}$$

Motivation

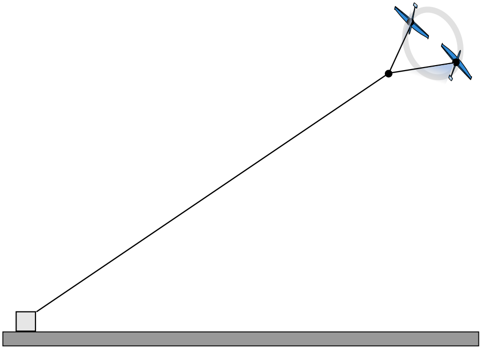


- Optimize a single pumping cycle of a dual-kite AWE system
- Formulate OCP → discretize to NLP

$$\begin{aligned} \min_{w} \quad & F(w) \\ \text{s.t.} \quad & 0 = G(w), \\ & 0 \leq H(w) \end{aligned}$$

- Very large, complicated nonlinear problem, need good strategy and initialization to solve

Motivation

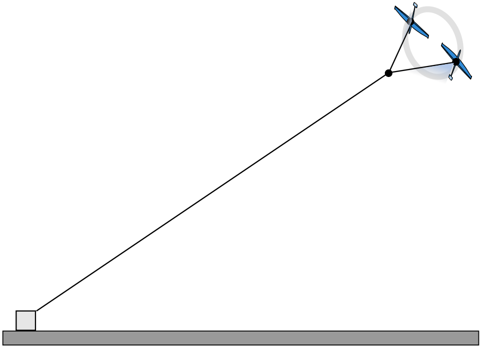


- Optimize a single pumping cycle of a dual-kite AWE system
- Formulate OCP $\rightarrow$ discretize to NLP

$$\begin{aligned} \min_{w} \quad & F(w) \\ \text{s.t.} \quad & 0 = G(w), \\ & 0 \leq H(w) \end{aligned}$$

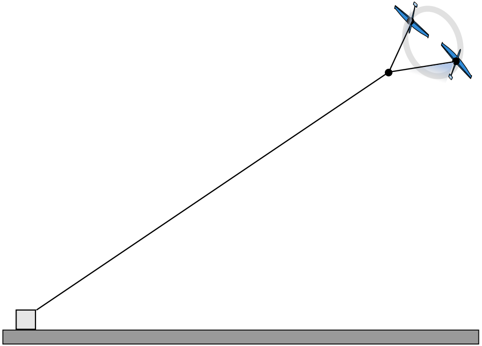
- Very large, complicated nonlinear problem, need good strategy and initialization to solve
- Software Packages such as the AWEBox[2]

Model



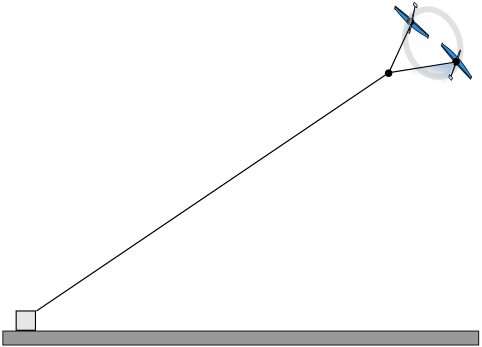
- Dual Kite Setup, two AP2 planes

Model



- Dual Kite Setup, two AP2 planes
- High-Fidelity 6-DOF model for each plane

Model

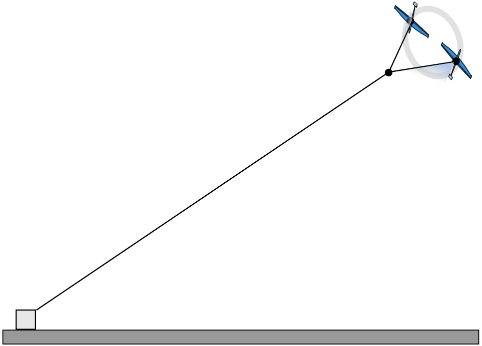


- Dual Kite Setup, two AP2 planes
- High-Fidelity 6-DOF model for each plane
- System Dynamics (Index-1 DEA)

$$f(x, u, z, t) = 0$$

with $x \in \mathbb{R}^{51}$, $u \in \mathbb{R}^{19}$, $z \in \mathbb{R}^{3}$, based on Lagrangian dynamics

Model



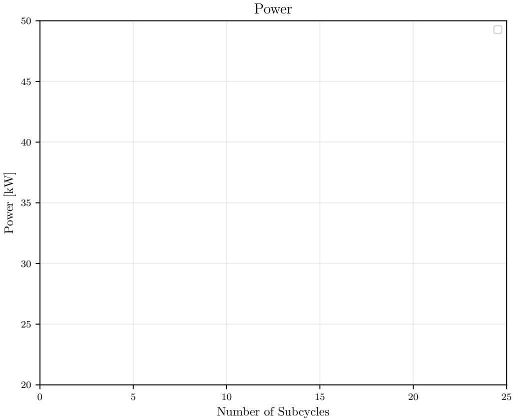
- Dual Kite Setup, two AP2 planes
- High-Fidelity 6-DOF model for each plane
- System Dynamics (Index-1 DEA)

$$f(x, u, z, t) = 0$$

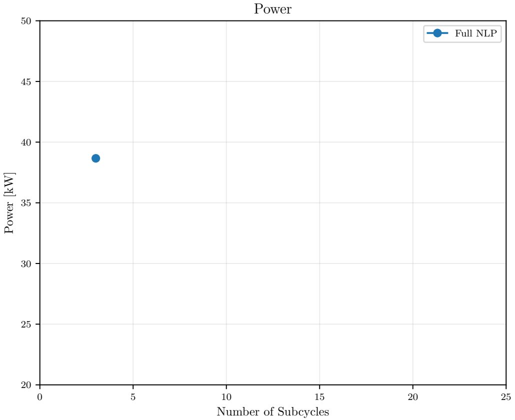
with $x \in \mathbb{R}^{51}$, $u \in \mathbb{R}^{19}$, $z \in \mathbb{R}^{3}$, based on Lagrangian dynamics

- x: 21 states per kite + 6 states connection point + 3 auxillary

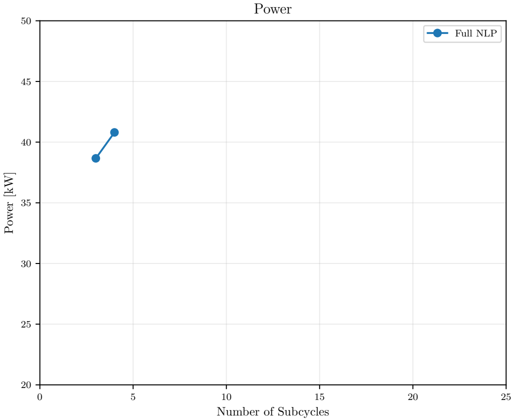
Experiments



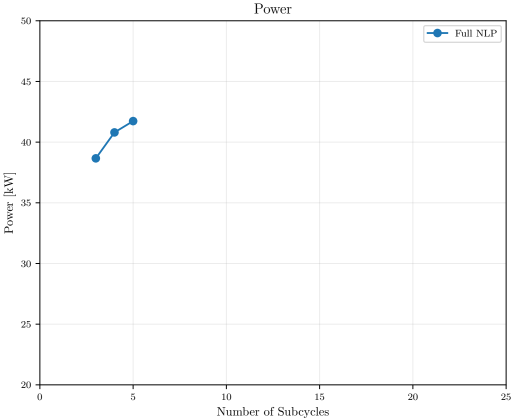
Experiments



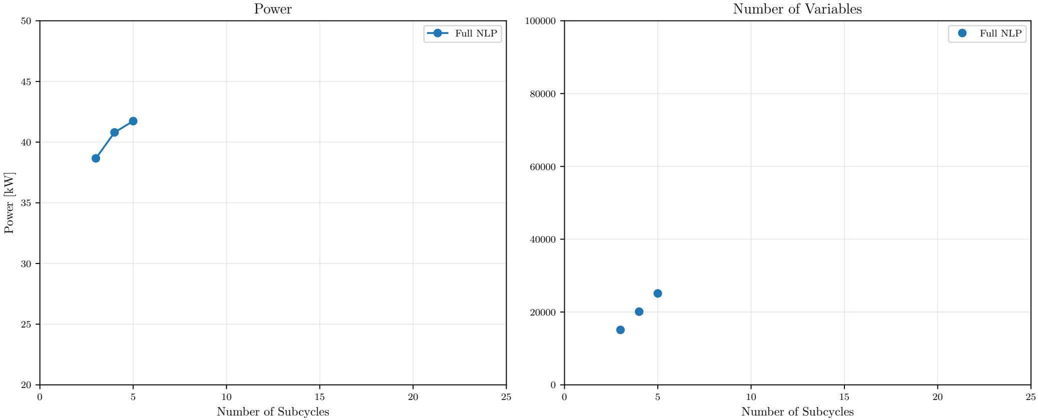
Experiments



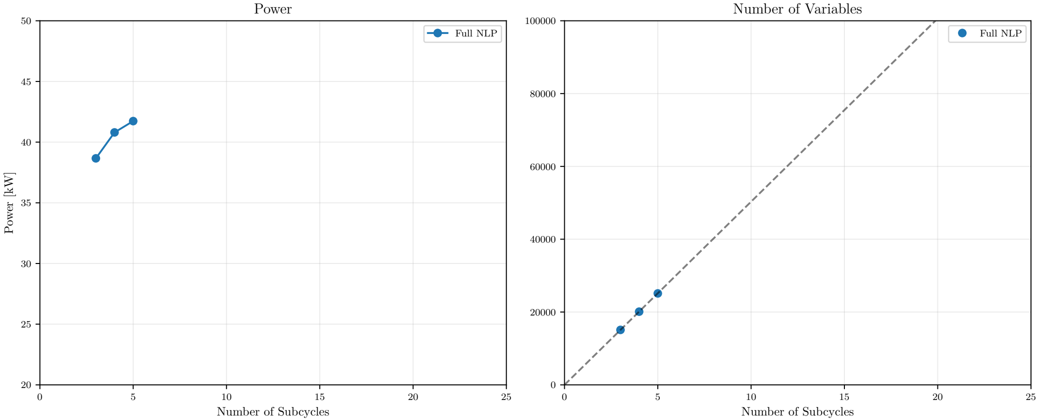
Experiments



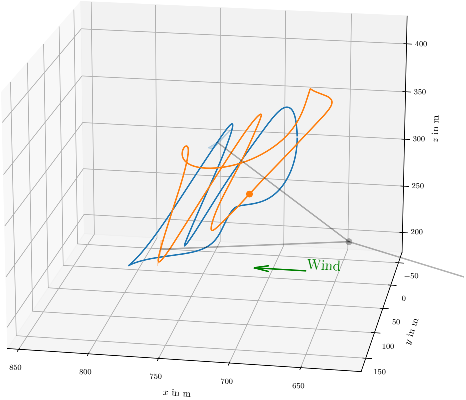
Experiments



Experiments

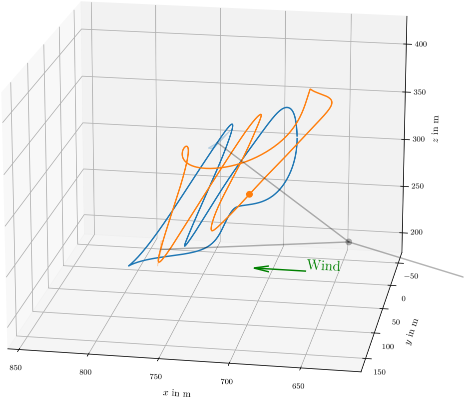


Some Observations

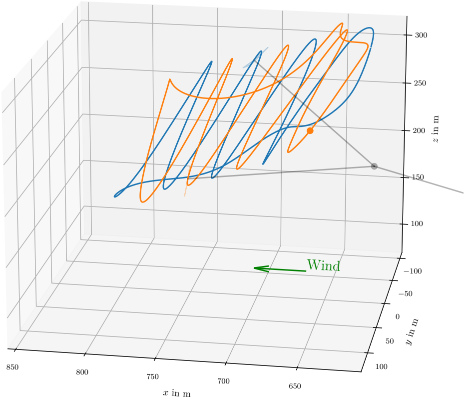


$N = 2$

Some Observations

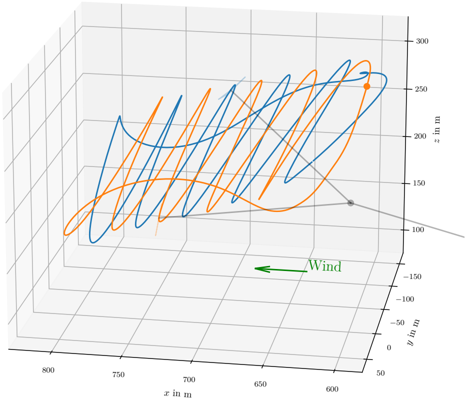


$N = 2$



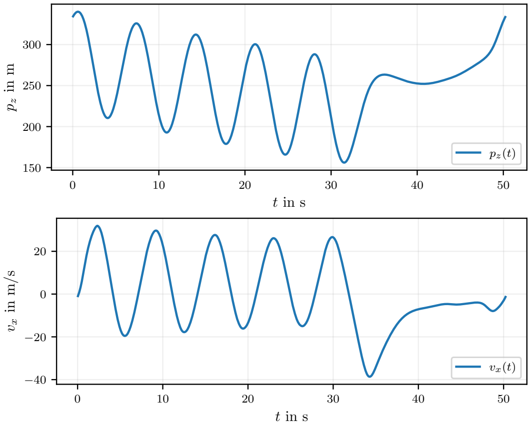
$N = 4$

Some Observations



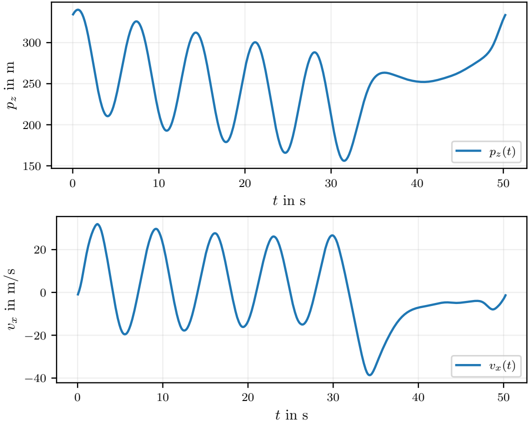
$$N = 5$$

Some Observations



$N = 5$

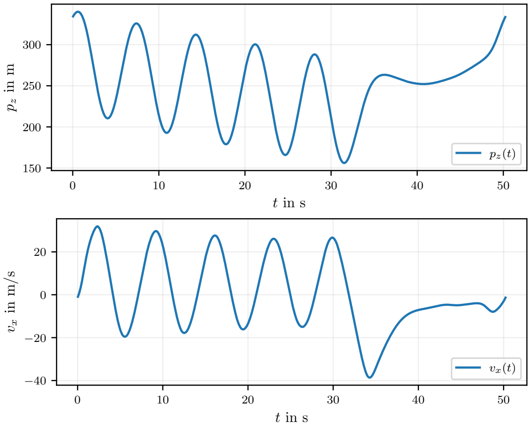
Some Observations



$N = 5$

- In the reel-out phase, the subcycles look similar

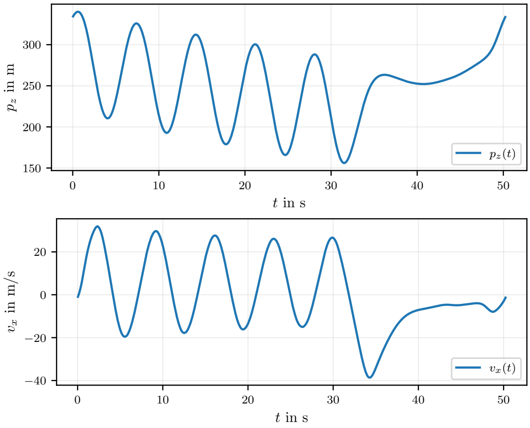
Some Observations



$N = 5$

- In the reel-out phase, the subcycles look similar
- There is some 'slow' or 'average' mode of the trajectory

Some Observations



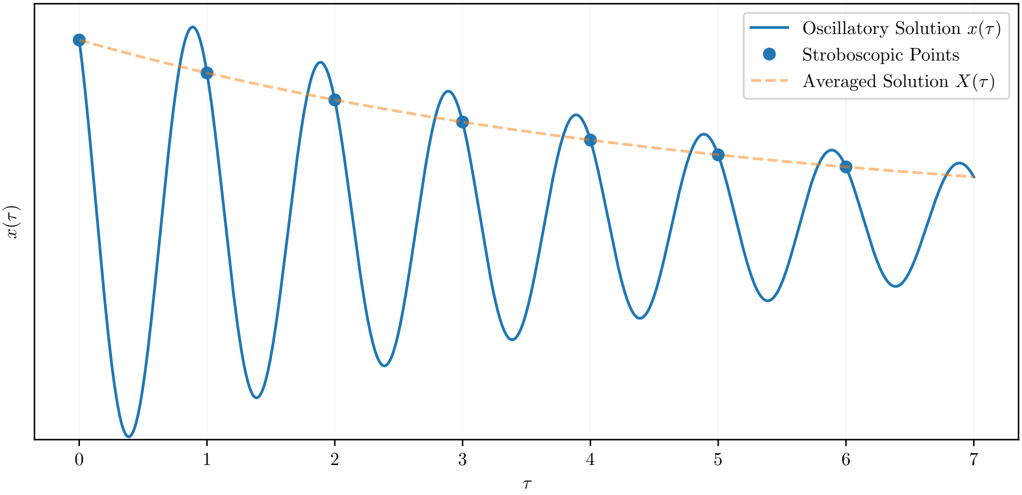
$N = 5$

- In the reel-out phase, the subcycles look similar
- There is some 'slow' or 'average' mode of the trajectory

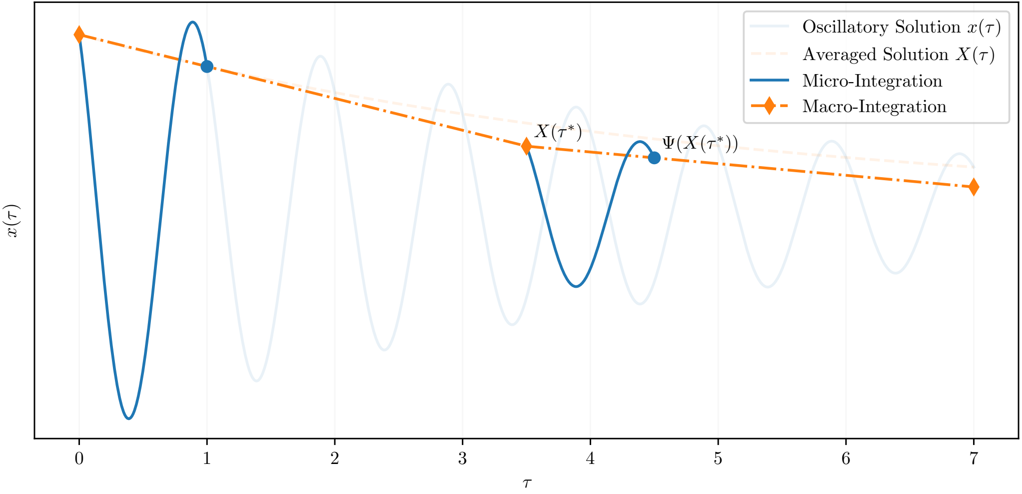
Strong Assumption

In the reel-out phase, the power optimal trajectory $x^*(t)$ and the corresponding control $u^*(t)$ consist of many similar, slowly changing cycles.

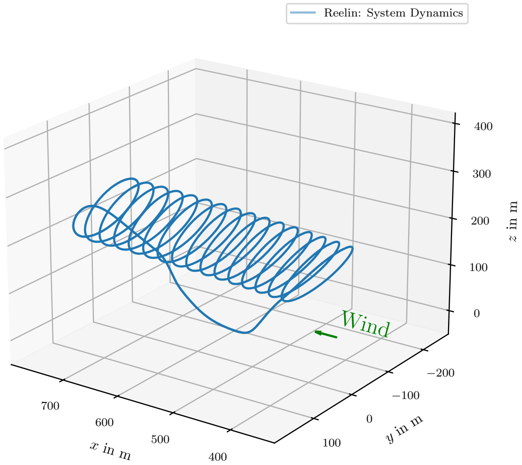
Stroboscopic Averaging Method



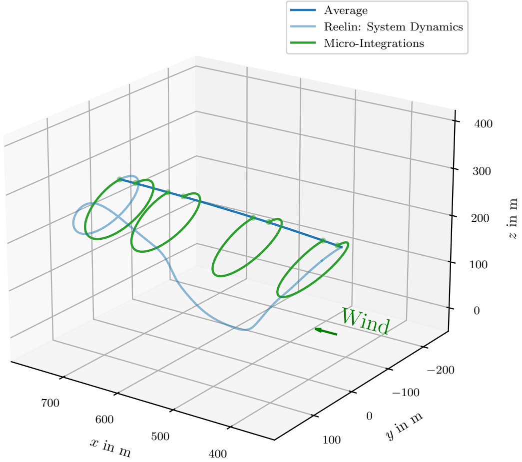
Stroboscopic Averaging Method



... Applied to Dual Kites

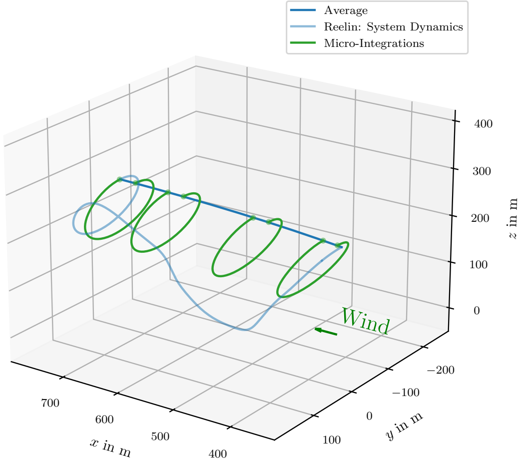


... Applied to Dual Kites

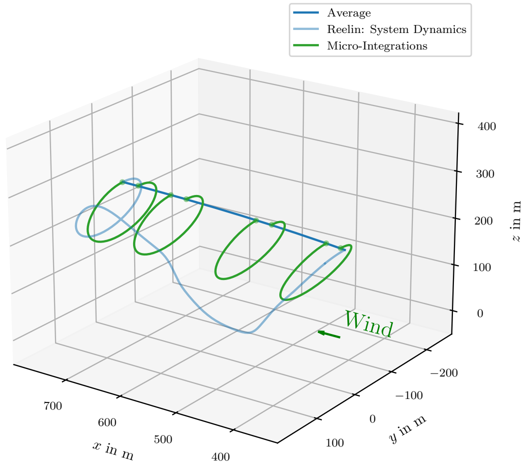


- **Reel-out phase:** Simulate the approximated *average dynamics* using stroboscopic averaging

... Applied to Dual Kites

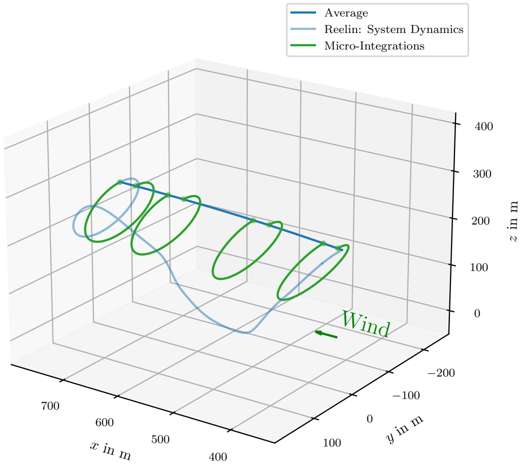


- **Reel-out phase:** Simulate the approximated *average dynamics* using stroboscopic averaging
  - Macro-Integration of the average dynamics with large steps

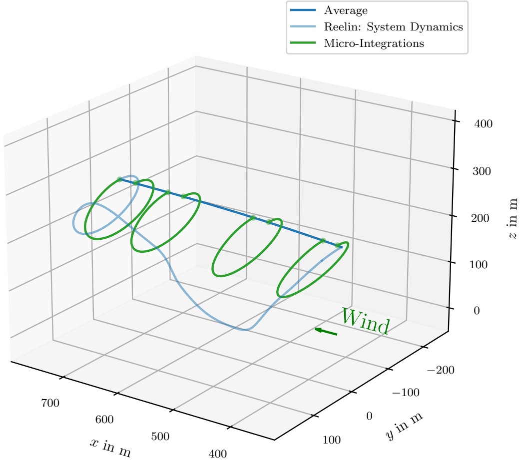


- **Reel-out phase:** Simulate the approximated *average dynamics* using stroboscopic averaging
  - Macro-Integration of the average dynamics with large steps
  - Approximation of the average dynamics using

… Applied to Dual Kites

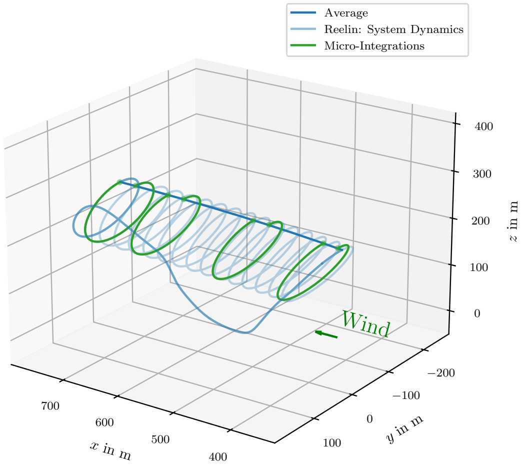


- **Reel-out phase:** Simulate the approximated *average dynamics* using stroboscopic averaging
  - Macro-Integration of the average dynamics with large steps
  - Approximation of the average dynamics using
  - Micro-Integrations the *model dynamics* over one cycle



- **Reel-out phase:** Simulate the approximated *average dynamics* using stroboscopic averaging
  - Macro-Integration of the average dynamics with large steps
  - Approximation of the average dynamics using
  - Micro-Integrations the *model dynamics* over one cycle
- **Reel-in phase:** Simulate the *model dynamics*

... Applied to Dual Kites



- **Reel-out phase:** Simulate the approximated *average dynamics* using stroboscopic averaging
  - Macro-Integration of the average dynamics with large steps
  - Approximation of the average dynamics using
  - Micro-Integrations the *model dynamics* over one cycle
- **Reel-in phase:** Simulate the *model dynamics*

Regularization

Strong Assumption

In the reel-out phase, the power optimal trajectory $x^*(t)$ and the corresponding control $u^*(t)$ consist of many similar, slowly changing cycles.

Picture from freePik.com

Regularization

Strong Assumption

In the reel-out phase, the power optimal trajectory $x^*(t)$ and the corresponding control $u^*(t)$ consist of many similar, slowly changing cycles.

Solver

- similar cycles might be suboptimal

Picture from freePik.com

Regularization

Researcher

- reward similarity in micro-integrations

Strong Assumption

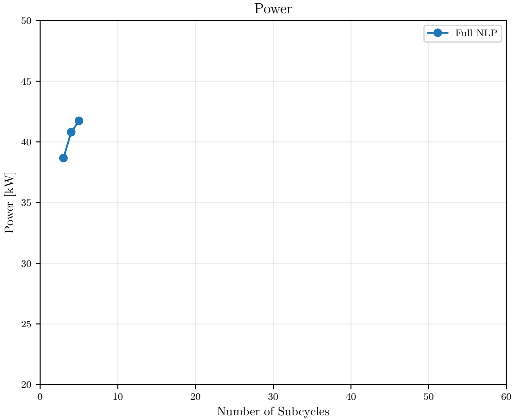
In the reel-out phase, the power optimal trajectory $x^*(t)$ and the corresponding control $u^*(t)$ consist of many similar, slowly changing cycles.

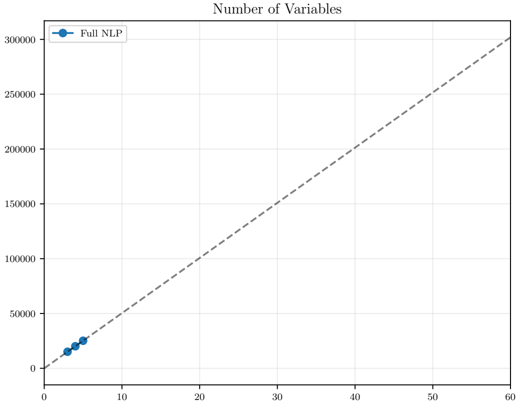
Solver

- similar cycles might be suboptimal

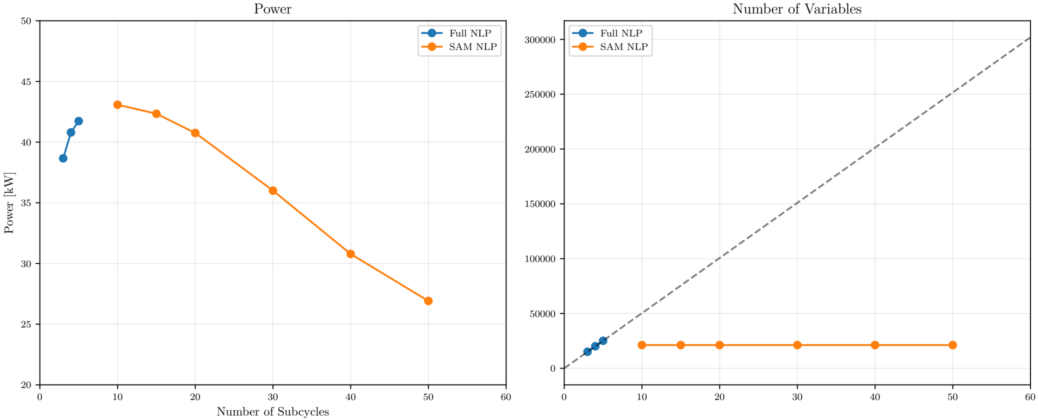
Picture from freePik.com

Experiments (Continued)

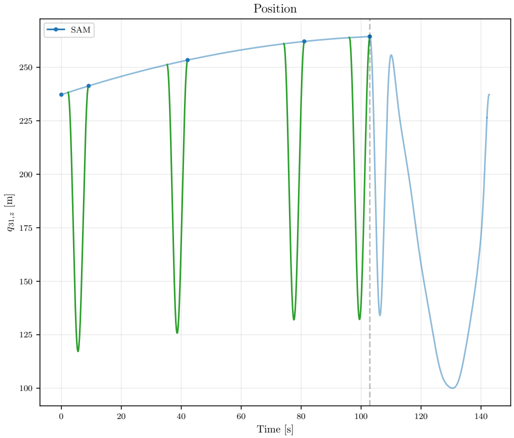




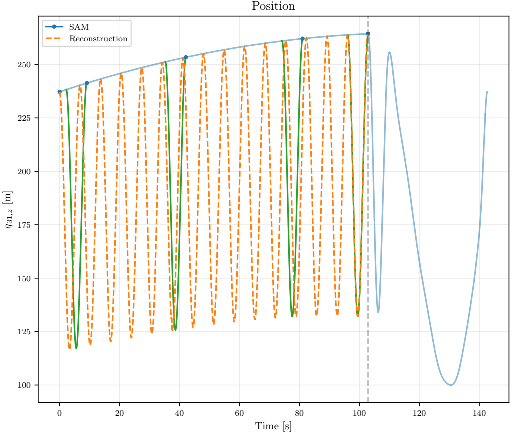
Experiments (Continued)



What about the full trajectory?

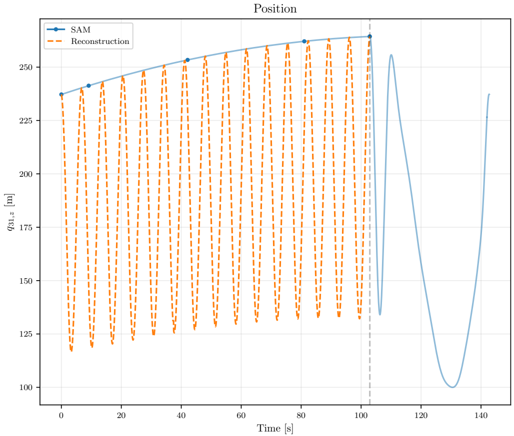


What about the full trajectory?



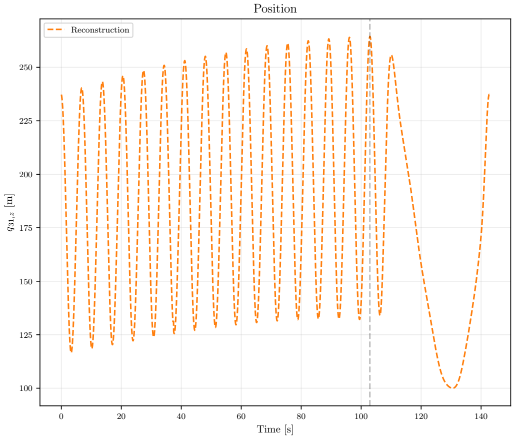
- 'Reconstruct' the reel-out phase by interpolating the micro-integrations

What about the full trajectory?



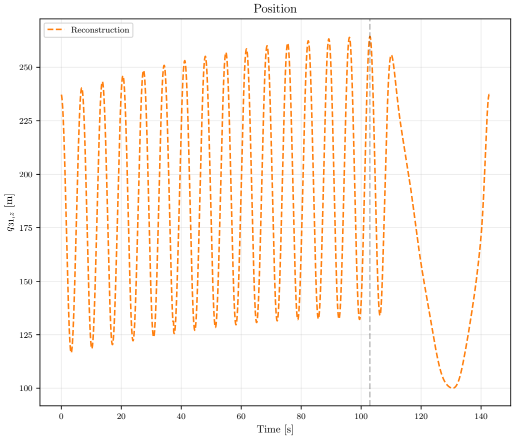
- 'Reconstruct' the reel-out phase by interpolating the micro-integrations
- Not a physical trajectory, only an approximation.

What about the full trajectory?



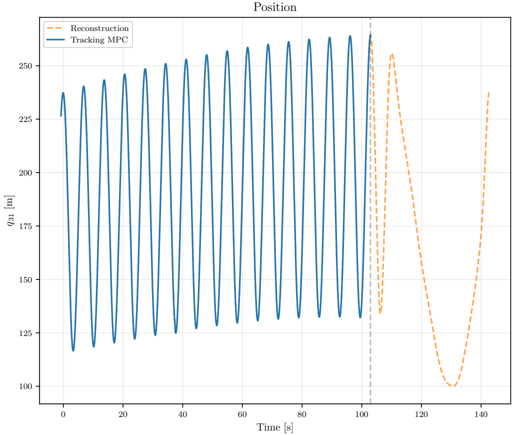
- 'Reconstruct' the reel-out phase by interpolating the micro-integrations
- Not a physical trajectory, only an approximation.

What about the full trajectory?

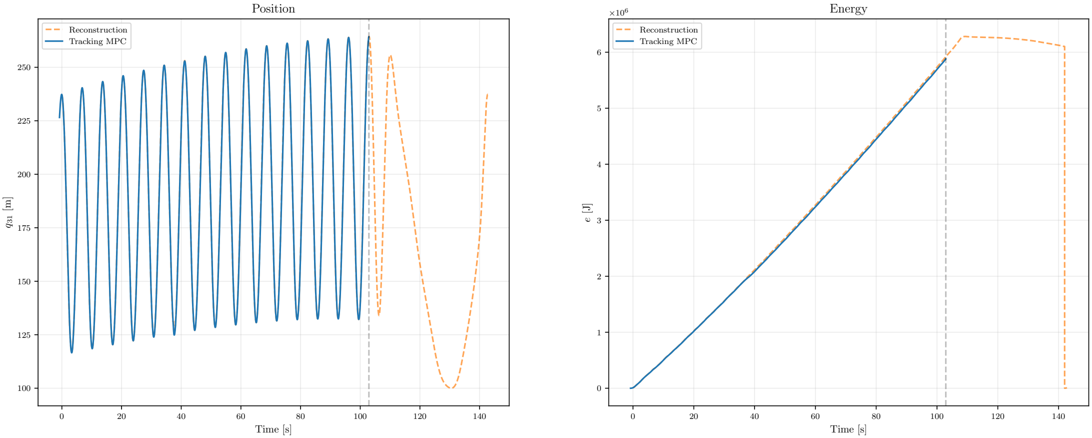


- 'Reconstruct' the reel-out phase by interpolating the micro-integrations
- Not a physical trajectory, only an approximation.
- 'Validate' the trajectory using a tracking MPC (very expensive)

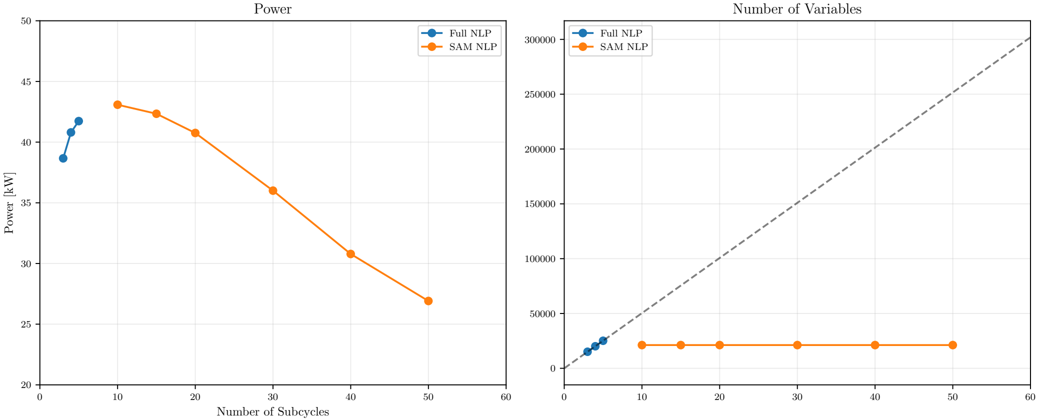
Validation with Tracking MPC



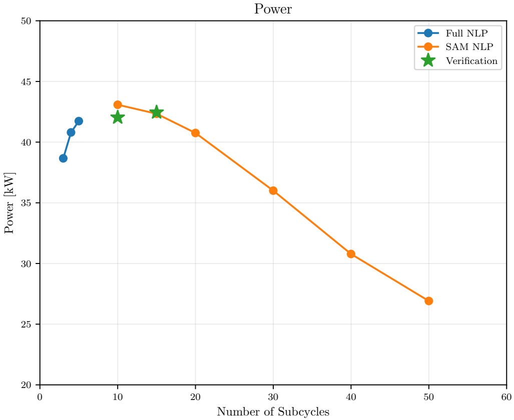
Validation with Tracking MPC

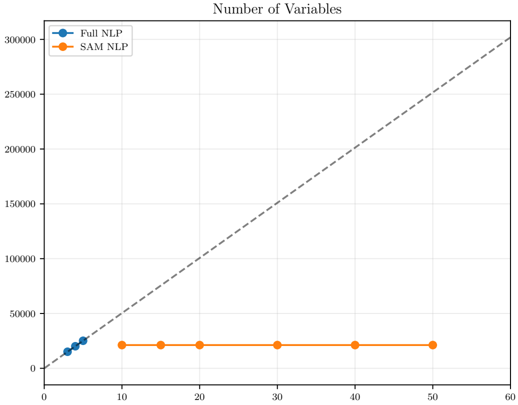


Experiments (Continued)



Experiments (Continued)





Conclusion

- A method to efficiently optimize dual kite trajectories with a large number of subcycles.

Conclusion

- A method to efficiently optimize dual kite trajectories with a large number of subcycles.
- Based on the assumption that the subcycles are only changing slowly

Conclusion

- A method to efficiently optimize dual kite trajectories with a large number of subcycles.
- Based on the assumption that the subcycles are only changing slowly
- "Simulate only 4 cycles instead of 50"

Conclusion

- A method to efficiently optimize dual kite trajectories with a large number of subcycles.
- Based on the assumption that the subcycles are only changing slowly
- "Simulate only 4 cycles instead of 50"
- We can 'reconstruct' the full trajectory and validate it using a tracking MPC

Conclusion

- A method to efficiently optimize dual kite trajectories with a large number of subcycles.
- Based on the assumption that the subcycles are only changing slowly
- "Simulate only 4 cycles instead of 50"
- We can 'reconstruct' the full trajectory and validate it using a tracking MPC

Not covered (yet):

- Induction Model

Conclusion

- A method to efficiently optimize dual kite trajectories with a large number of subcycles.
- Based on the assumption that the subcycles are only changing slowly
- "Simulate only 4 cycles instead of 50"
- We can 'reconstruct' the full trajectory and validate it using a tracking MPC

Not covered (yet):

- Induction Model
- Number of cycles as optimization variable

Conclusion

- A method to efficiently optimize dual kite trajectories with a large number of subcycles.
- Based on the assumption that the subcycles are only changing slowly
- "Simulate only 4 cycles instead of 50"
- We can 'reconstruct' the full trajectory and validate it using a tracking MPC

Not covered (yet):

- Induction Model
- Number of cycles as optimization variable
- ...

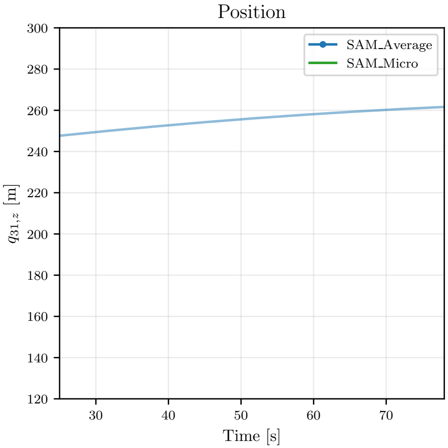
Thank you for your attention!

Useful Sources

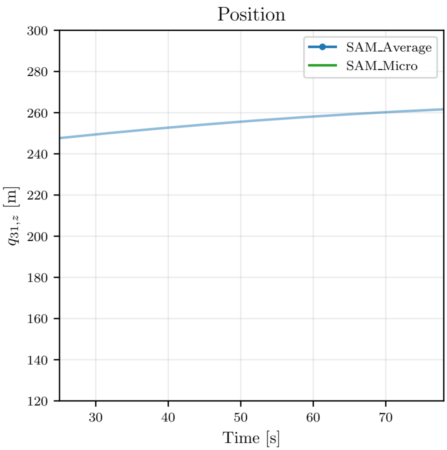
Mari Paz Calvo, Philippe Chartier, Ander Murua, and Jesús María Sanz-Serna.
A stroboscopic numerical method for highly oscillatory problems.
In Björn Engquist, Olof Runborg, and Yen-Hsi R. Tsai, editors, Numerical Analysis of Multiscale Computations, pages 71–85, Berlin, Heidelberg, 2012. Springer Berlin Heidelberg.

Jochem De Schutter, Rachel Leuthold, Thilo Bronnenmeyer, Elena Malz, Sebastien Gros, and Moritz Diehl.
Awebox: An optimal control framework for single- and multi-aircraft airborne wind energy systems.
Energies, 16(4), 2023.

Micro-Integration



Micro-Integration

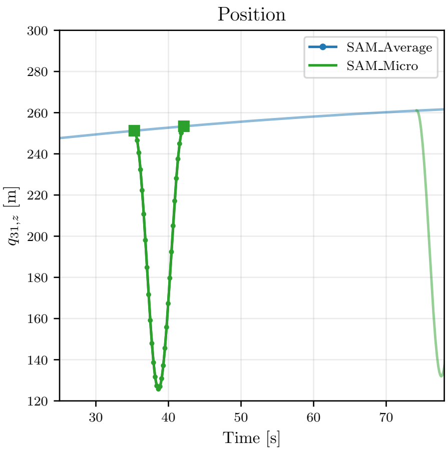


- Approximate the 'average dynamics' with DAE:

$$\dot{X} = F(X, Z) \tag{1a}$$

$$0 = G(X, Z, T) \tag{1b}$$

Micro-Integration

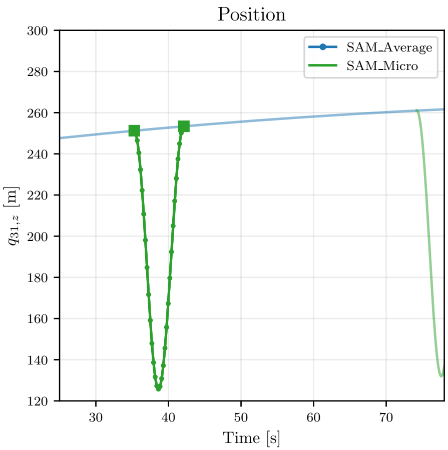


- Approximate the 'average dynamics' with DAE:

$$\dot{X} = F(X, Z) \tag{1a}$$

$$0 = G(X, Z, T) \tag{1b}$$

- G: equations to simulate the cycle



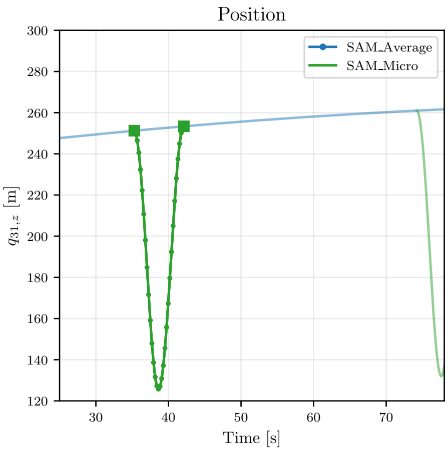
- Approximate the 'average dynamics' with DAE:

$$\dot{X} = F(X, Z) \quad (1a)$$

$$0 = G(X, Z, T) \quad (1b)$$

- G: equations to simulate the cycle
- Algebraic cycle variables Z :

Micro-Integration



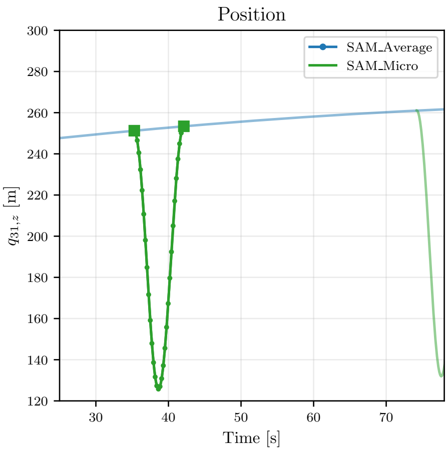
- Approximate the 'average dynamics' with DAE:

$$\dot{X} = F(X, Z) \quad (1a)$$

$$0 = G(X, Z, T) \quad (1b)$$

- G: equations to simulate the cycle
- Algebraic cycle variables Z :
  - Startpoint x^-, Endpoint x^+

Micro-Integration



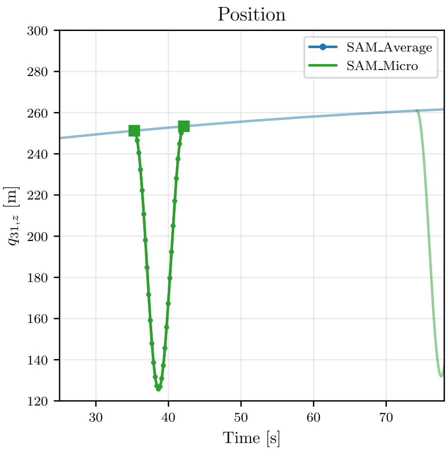
- Approximate the 'average dynamics' with DAE:

$$\dot{X} = F(X, Z) \tag{1a}$$

$$0 = G(X, Z, T) \tag{1b}$$

- G: equations to simulate the cycle
- Algebraic cycle variables Z :
  - Startpoint x^-, Endpoint x^+
  - Shooting nodes of the micro-integration
  - Variable Duration T that scales the dynamics

Micro-Integration



- Approximate the 'average dynamics' with DAE:

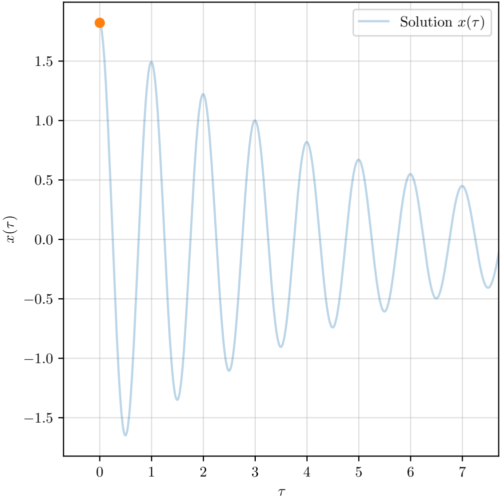
$$\dot{X} = F(X, Z) \qquad (1a)$$

$$0 = G(X, Z, T) \qquad (1b)$$

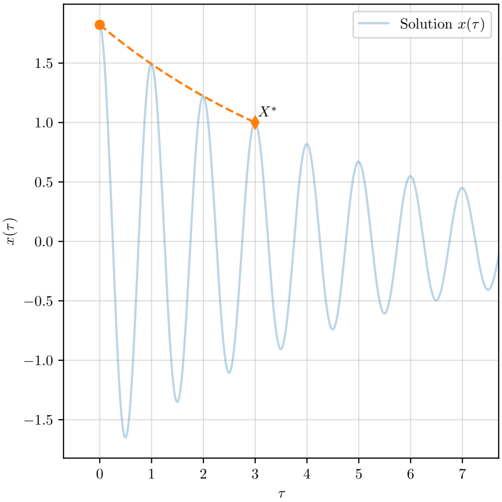
- G: equations to simulate the cycle
- Algebraic cycle variables Z :
  - Startpoint x^-, Endpoint x^+
  - Shooting nodes of the micro-integration
  - Variable Duration T that scales the dynamics
- Regularization to keep the start and endpoint 'close' to each other

$$J_{\text{reg}} = \cdots + \|x^+ - x^-\|_W^2 + \ldots \qquad (2)$$

Numerical Method for Efficient Simulation [1]

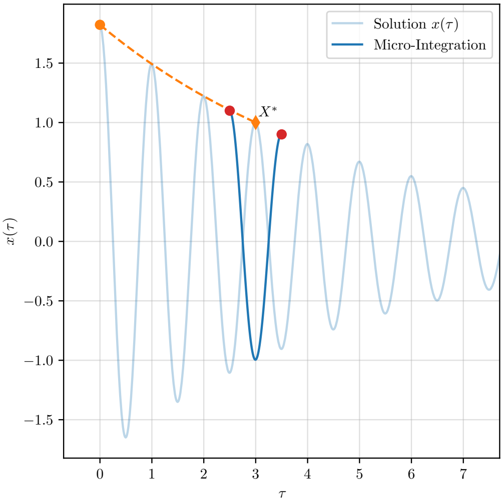


Numerical Method for Efficient Simulation [1]



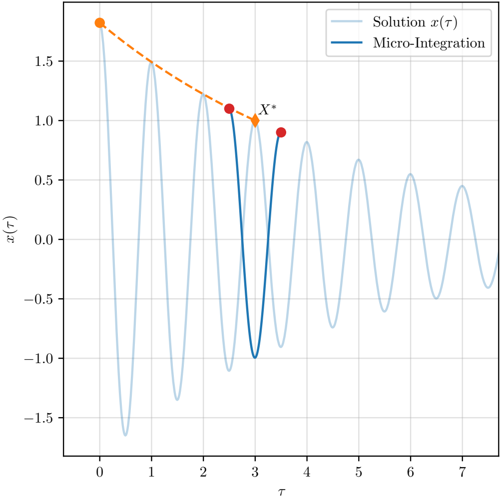
- At some point (τ^*, X^*):

Numerical Method for Efficient Simulation [1]



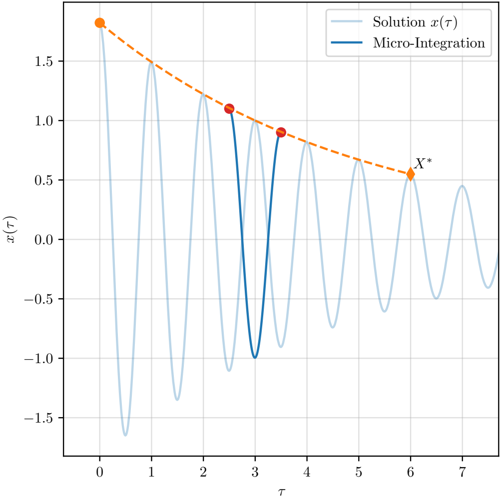
- At some point (τ^*, X^*):
  (a) perform one or more micro-integrations to evaluate the one-cycle map

Numerical Method for Efficient Simulation [1]



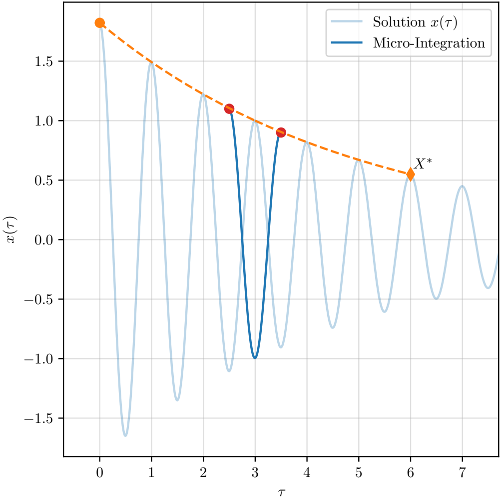
- At some point (τ^*, X^*):
  (a) perform one or more micro-integrations to evaluate the one-cycle map
  (b) approximate the averaged dynamics

Numerical Method for Efficient Simulation [1]



- At some point (τ^*, X^*):
  - (a) perform one or more micro-integrations to evaluate the one-cycle map
  - (b) approximate the averaged dynamics
- Macro-integrate the averaged dynamics

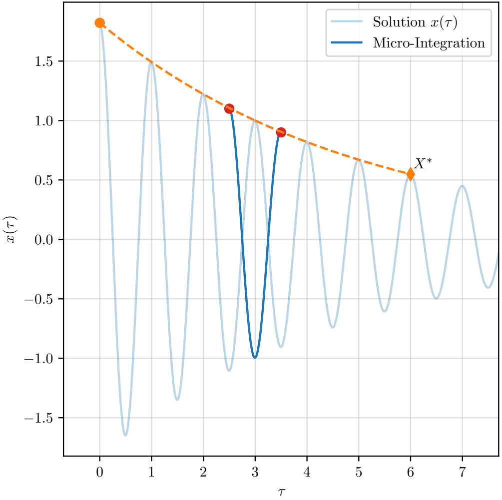
Numerical Method for Efficient Simulation [1]



- At some point (τ^*, X^*):
  - (a) perform one or more micro-integrations to evaluate the one-cycle map
  - (b) approximate the averaged dynamics
- Macro-integrate the averaged dynamics
- Integration horizon of integer size N cycles since

$$x(N) = X(N)$$

Numerical Method for Efficient Simulation [1]

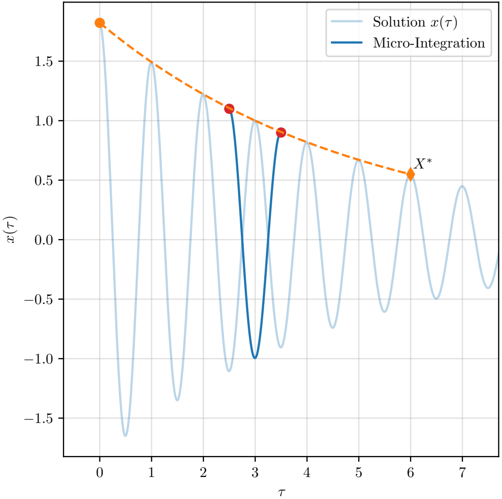


- At some point (τ^*, X^*):
  - (a) perform one or more micro-integrations to evaluate the one-cycle map
  - (b) approximate the averaged dynamics
- Macro-integrate the averaged dynamics
- Integration horizon of integer size N cycles since

$$x(N) = X(N)$$

- Three sources of error:

Numerical Method for Efficient Simulation [1]

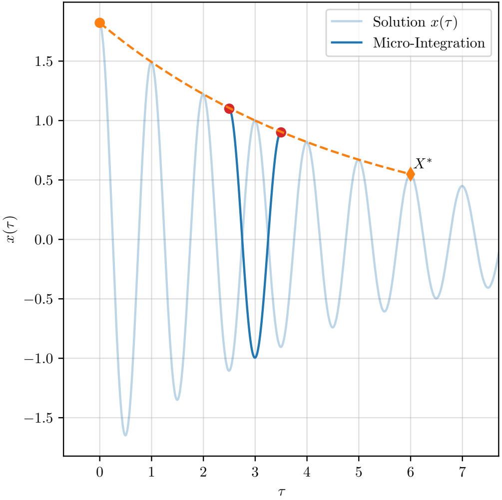


- At some point (τ^*, X^*):
  - (a) perform one or more micro-integrations to evaluate the one-cycle map
  - (b) approximate the averaged dynamics
- Macro-integrate the averaged dynamics
- Integration horizon of integer size N cycles since

$$x(N) = X(N)$$

- Three sources of error:
  - (a) errors in the micro-integration

Numerical Method for Efficient Simulation [1]

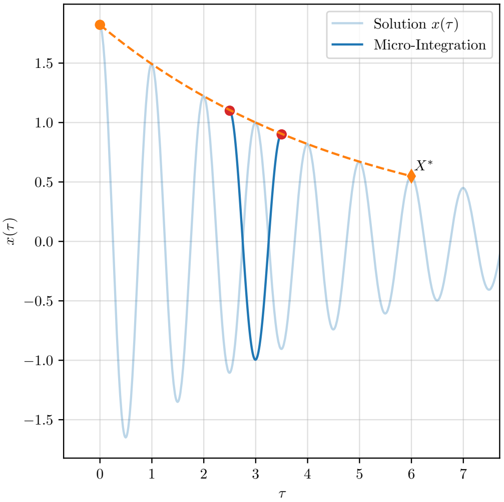


- At some point (τ^*, X^*):
  (a) perform one or more micro-integrations to evaluate the one-cycle map
  (b) approximate the averaged dynamics
- Macro-integrate the averaged dynamics
- Integration horizon of integer size N cycles since

$$x(N) = X(N)$$

- Three sources of error:
  (a) errors in the micro-integration
  (b) errors in the approximation of the dynamics

Numerical Method for Efficient Simulation [1]



- At some point (τ^*, X^*):
  - (a) perform one or more micro-integrations to evaluate the one-cycle map
  - (b) approximate the averaged dynamics
- Macro-integrate the averaged dynamics
- Integration horizon of integer size N cycles since

$$x(N) = X(N)$$

- Three sources of error:
  - (a) errors in the micro-integration
  - (b) errors in the approximation of the dynamics
  - (c) errors in the macro-integration