

arXiv:2203.01948
accepted in PRX Quantum



Counterdiabatic Optimised Local Driving (COLD)

Ieva Čepaitė, Andrew J. Daley, Callum W. Duncan

University of Strathclyde, Glasgow



University of
Strathclyde
Glasgow

The Boston University logo, featuring a circular seal with Latin text and a crest, next to the words "BOSTON UNIVERSITY" in red capital letters.

BOSTON
UNIVERSITY

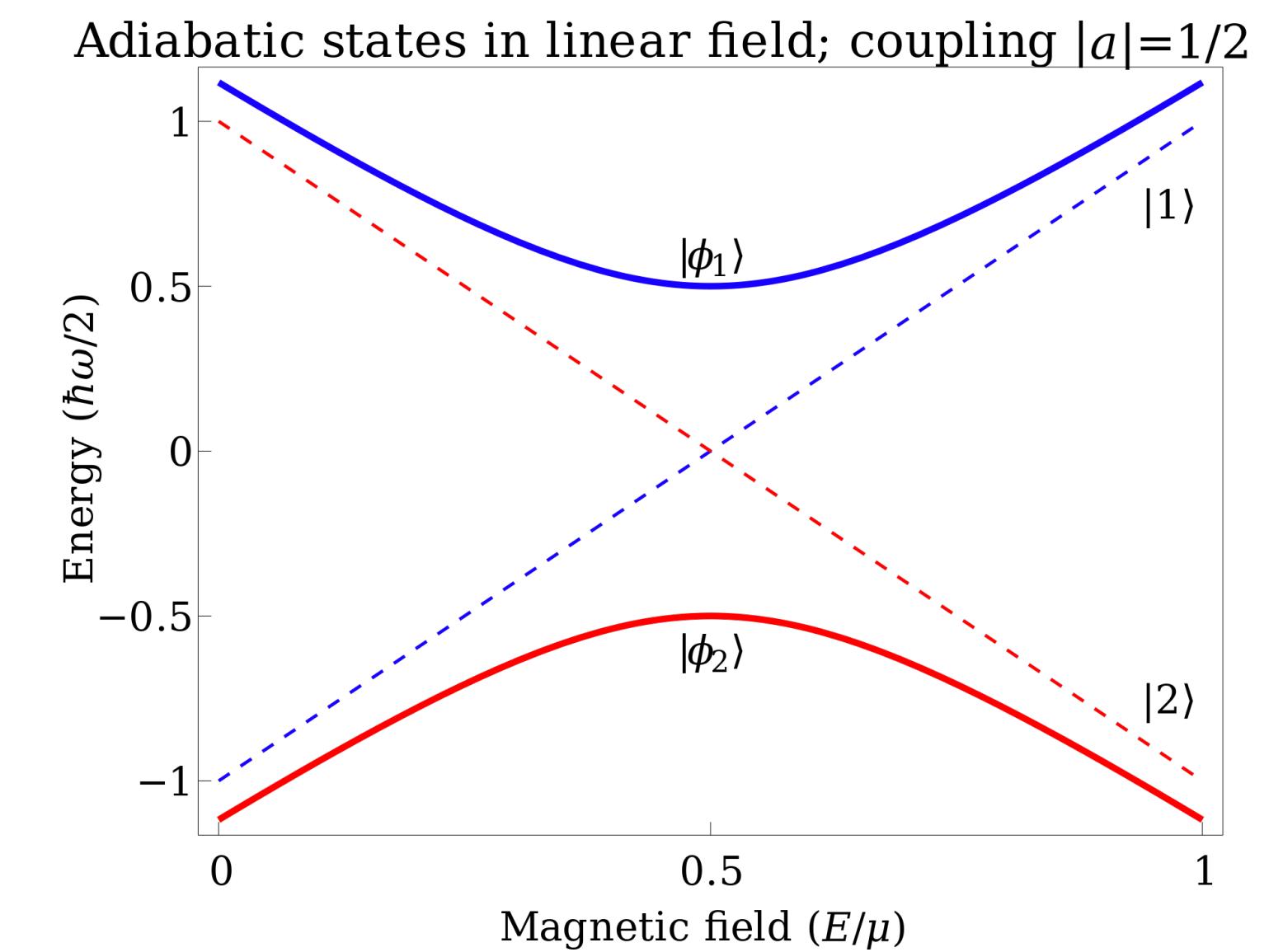




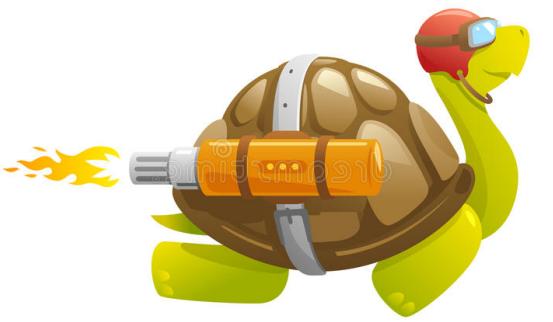
Adiabatic processes

- **Slowly** varying $H(\lambda)$ according to some $\lambda(t)$
 - System in eigenstate of $H(\lambda_0)$ $\xrightarrow{\text{(Adiabatic evolution)}}$ corresponding eigenstate of $H(\lambda_f)$ (**adiabatic theorem**)
- **Useful:** Adiabatic quantum computation, state preparation, quantum annealing...

! Trying to perform an adiabatic evolution **fast** leads to **losses**: transitions out of the required eigenstate

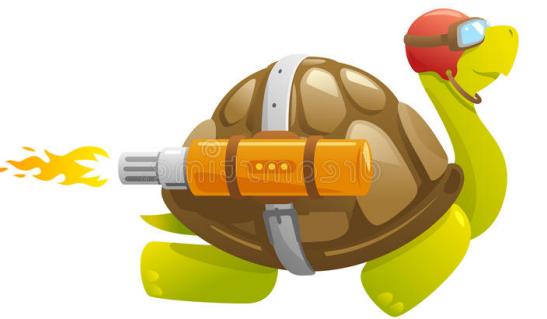


Speeding things up



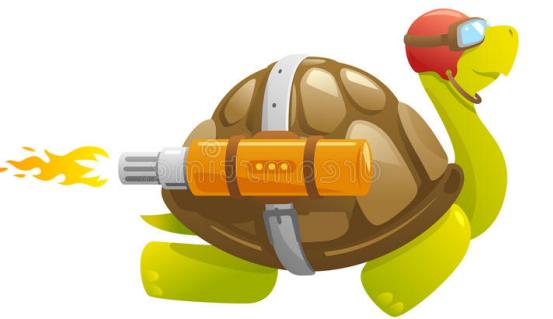
- What do we want?

Speeding things up



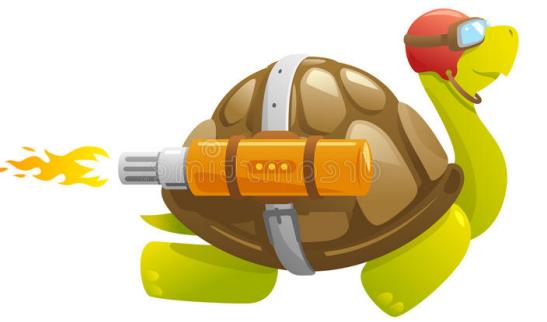
- What do we want?
 - Adiabatic dynamics!

Speeding things up



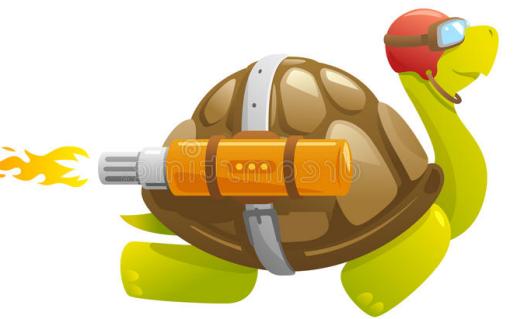
- What do we want?
 - Adiabatic dynamics!
- When do we want them?

Speeding things up

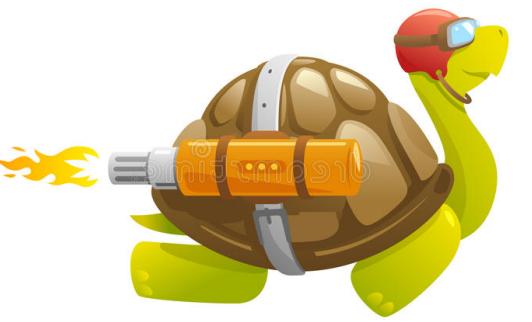


- What do we want?
 - Adiabatic dynamics!
- When do we want them?
 - As quickly as possible!

Speeding things up



- What do we want?
 - Adiabatic dynamics!
- When do we want them?
 - As quickly as possible!
(without losses/transitions)



Speeding things up

- What do we want?
 - Adiabatic dynamics!

Shortcuts to
adiabaticity

- When do we want them?

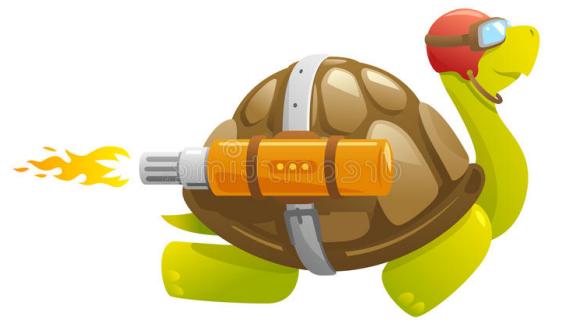
Rev. Mod. Phys. 91, 045001 (2019)

Optimal control
methods

- As quickly as possible!
(without losses/transitions)

Eur. Phys. J. D 69, 1 (2015).

Speeding things up



- What do we want?
 - Adiabatic dynamics!
- When do we want them?
 - As quickly as possible!
(without losses/transitions)

Shortcuts to
adiabaticity

Optimal control
methods

Local
counterdiabatic
driving

Speeding things up



- What do we want?
 - Adiabatic dynamics!
- When do we want them?
 - As quickly as possible!
(without losses/transitions)

Shortcuts to
adiabaticity

Optimal control
methods

Local
counterdiabatic
driving

Counterdiabatic
Optimised Local
Driving

Speeding things up



- What do we want?
 - Adiabatic dynamics!
- When do we want them?
 - As quickly as possible!
(without losses/transitions)

Shortcuts to
adiabaticity

Optimal control
methods

Local
counterdiabatic
driving

COLD

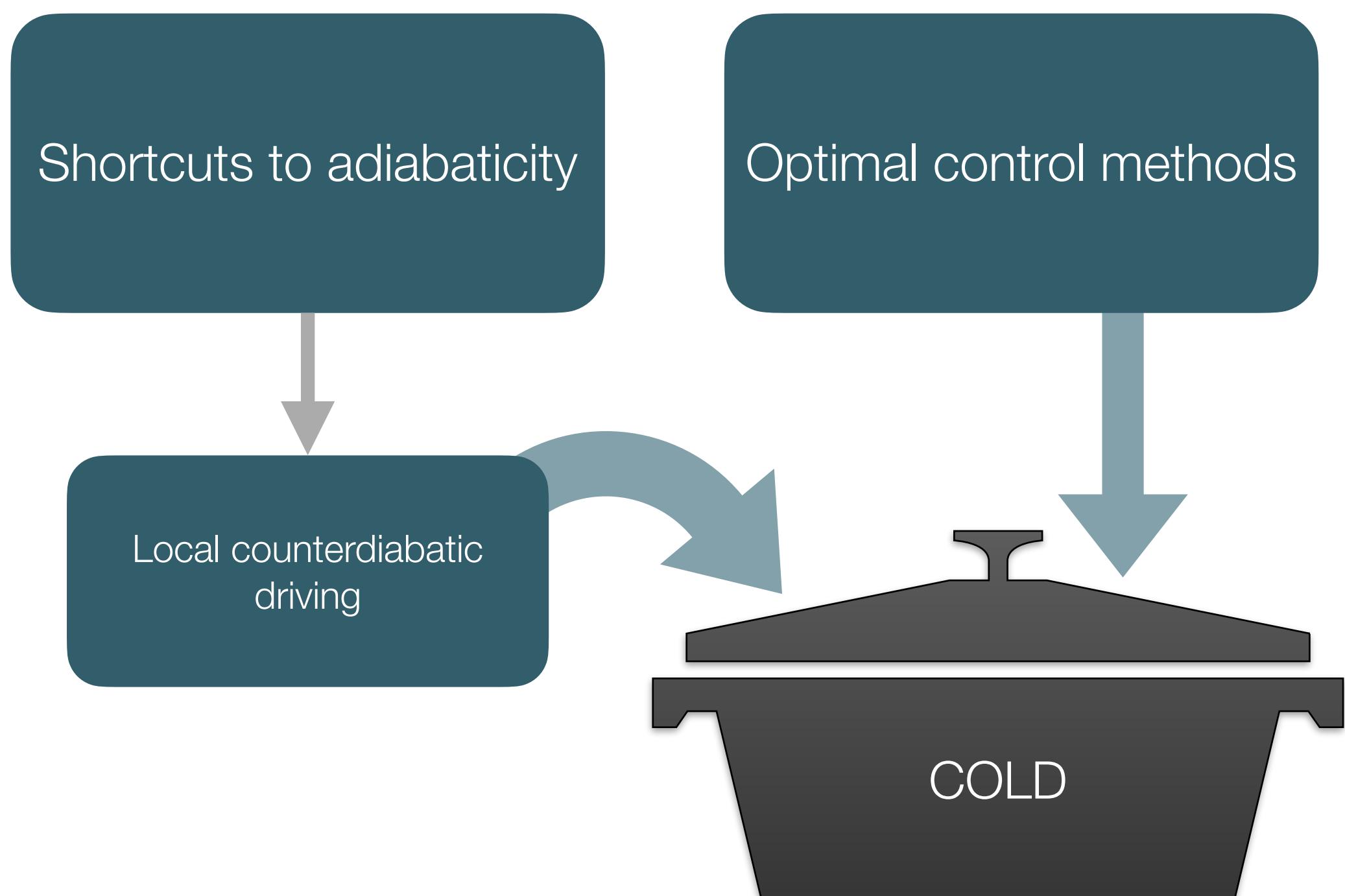
Counterdiabatic optimised local driving (COLD)

1. Start with adiabatic protocol: $H(\lambda)$



Counterdiabatic optimised local driving (COLD)

1. Start with adiabatic protocol: $H(\lambda)$



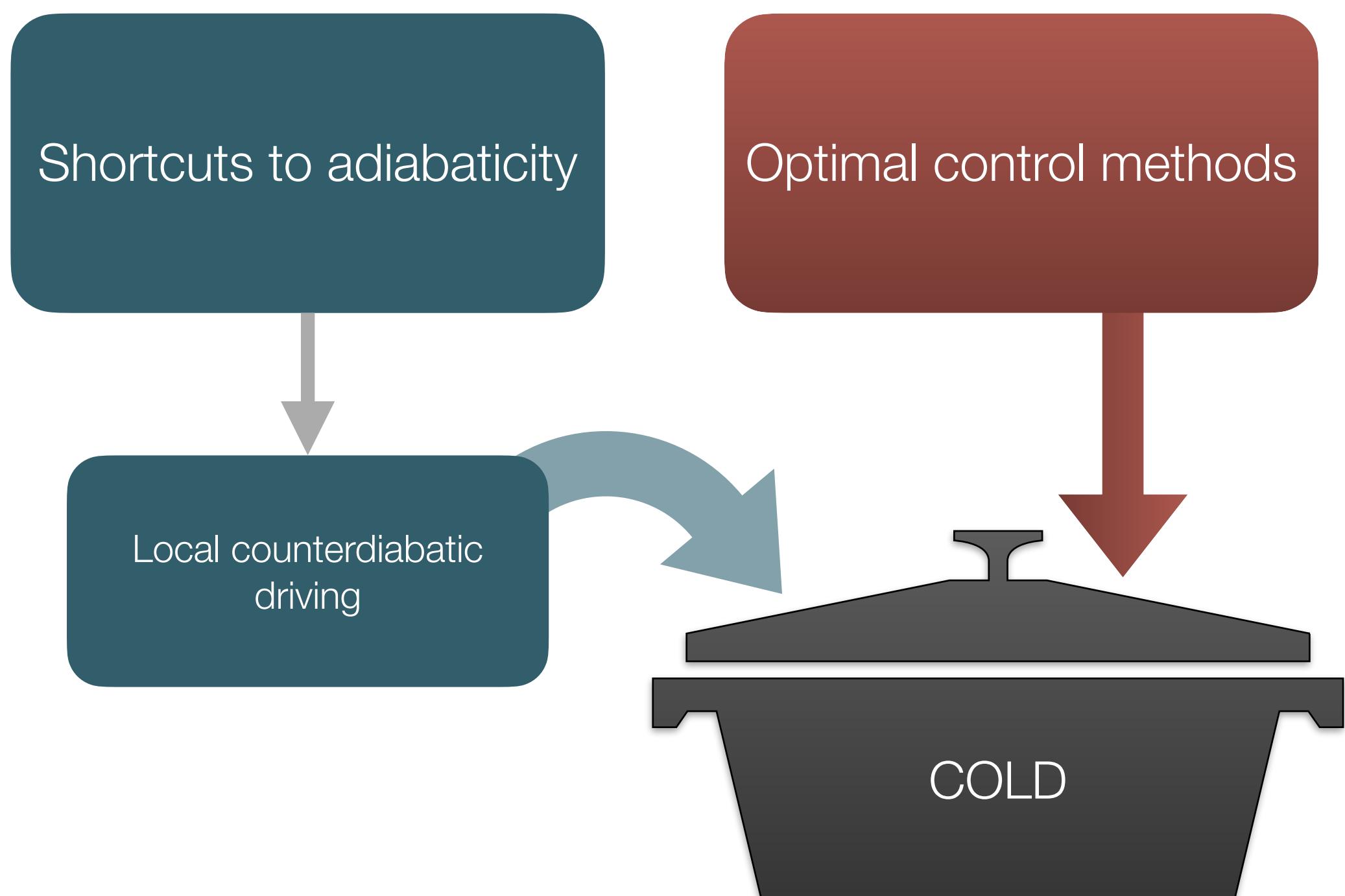
Counterdiabatic optimised local driving (COLD)

1. Start with adiabatic protocol: $H(\lambda)$



2. Add control pulse:

$$H_\beta(\lambda, \beta) = H(\lambda) + \beta \mathcal{O}_{\text{opt}}$$



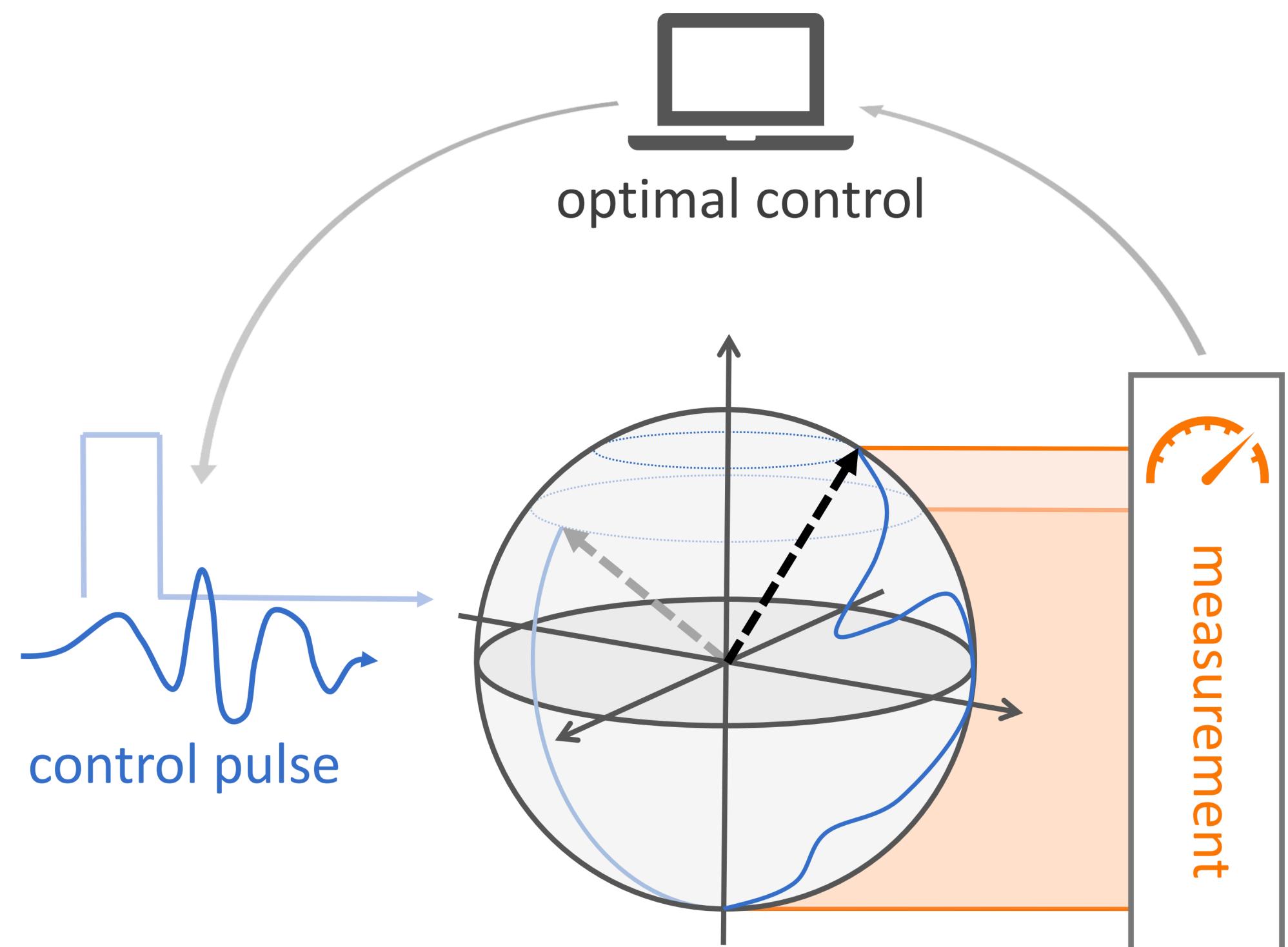
Counterdiabatic optimised local driving (COLD)

1. Start with adiabatic protocol: $H(\lambda)$



2. Add control pulse:

$$H_\beta(\lambda, \beta) = H(\lambda) + \beta \mathcal{O}_{\text{opt}}$$



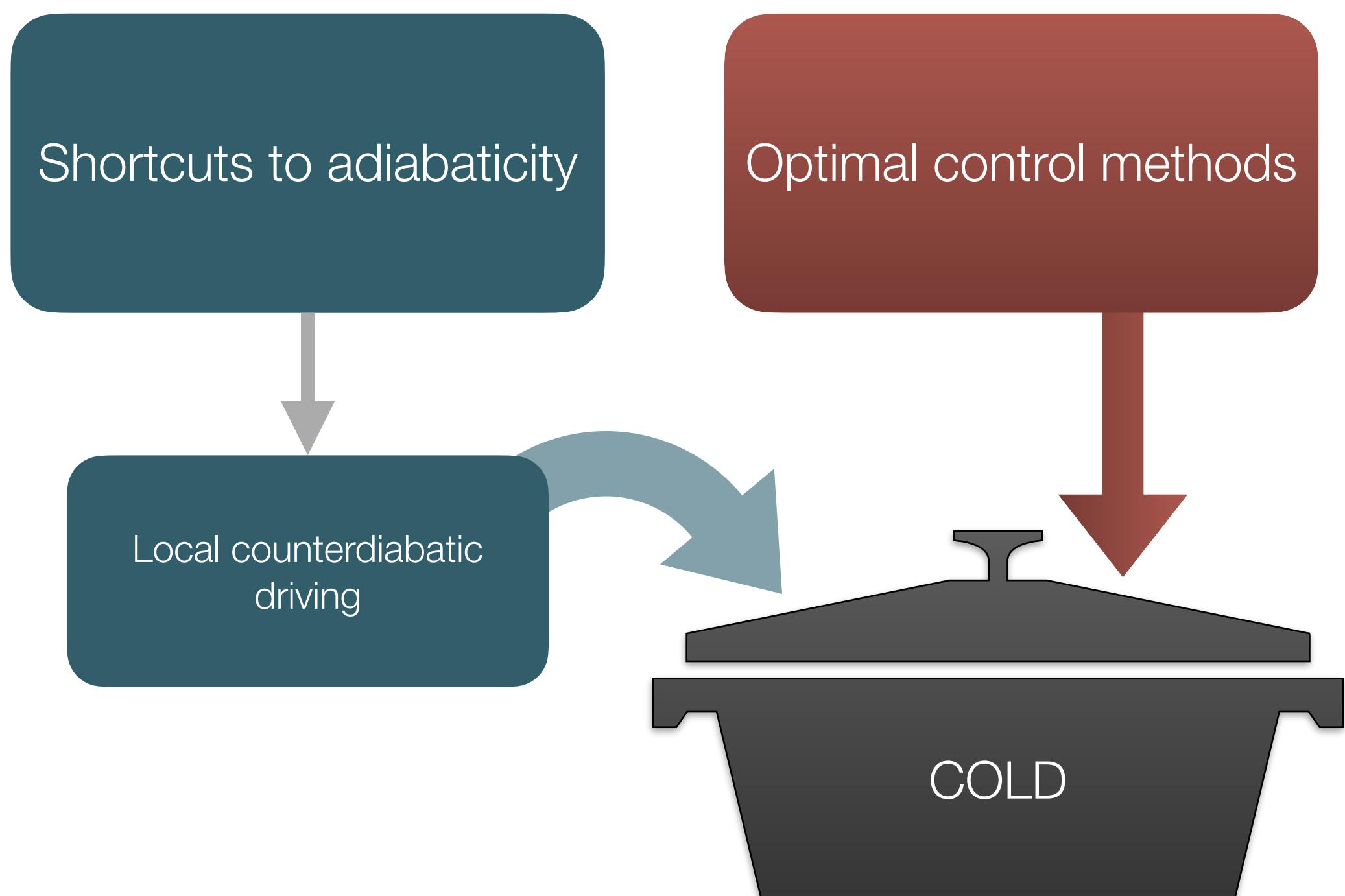
Counterdiabatic optimised local driving (COLD)

1. Start with adiabatic protocol: $H(\lambda)$



2. Add control pulse:

$$H_{\beta}(\lambda, \beta) = H(\lambda) + \beta \mathcal{O}_{\text{opt}}$$



Counterdiabatic optimised local driving (COLD)

1. Start with adiabatic protocol: $H(\lambda)$

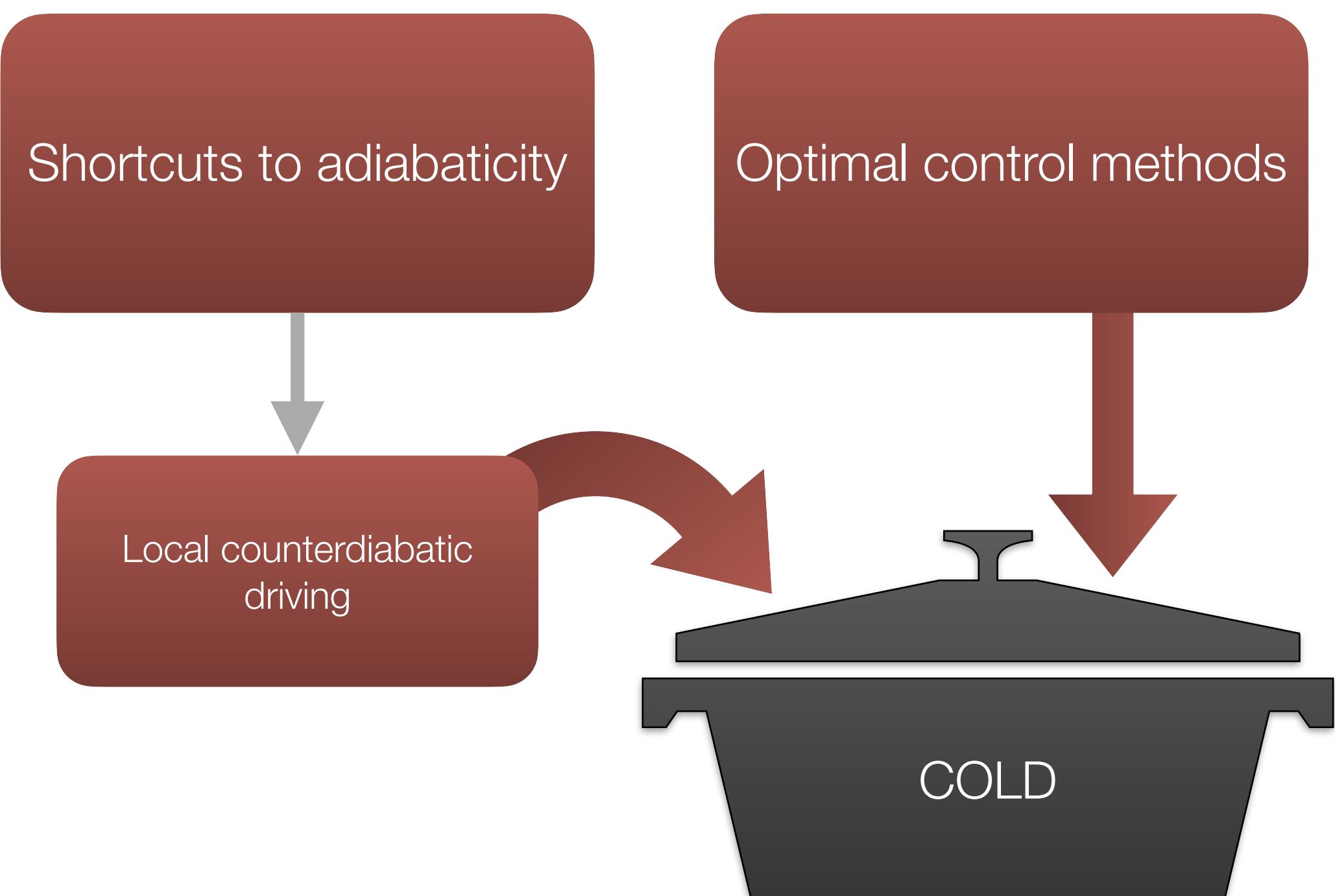


2. Add control pulse:

$$H_\beta(\lambda, \beta) = H(\lambda) + \beta \mathcal{O}_{\text{opt}}$$

3. Add LCD Ansatz:

$$H_{\text{COLD}}(\lambda, \beta) = H_\beta(\lambda, \beta) + \sum_j \alpha(\lambda, \beta) \mathcal{O}_{\text{LCD}}$$



Counterdiabatic optimised local driving (COLD)

1. Start with adiabatic protocol: $H(\lambda)$

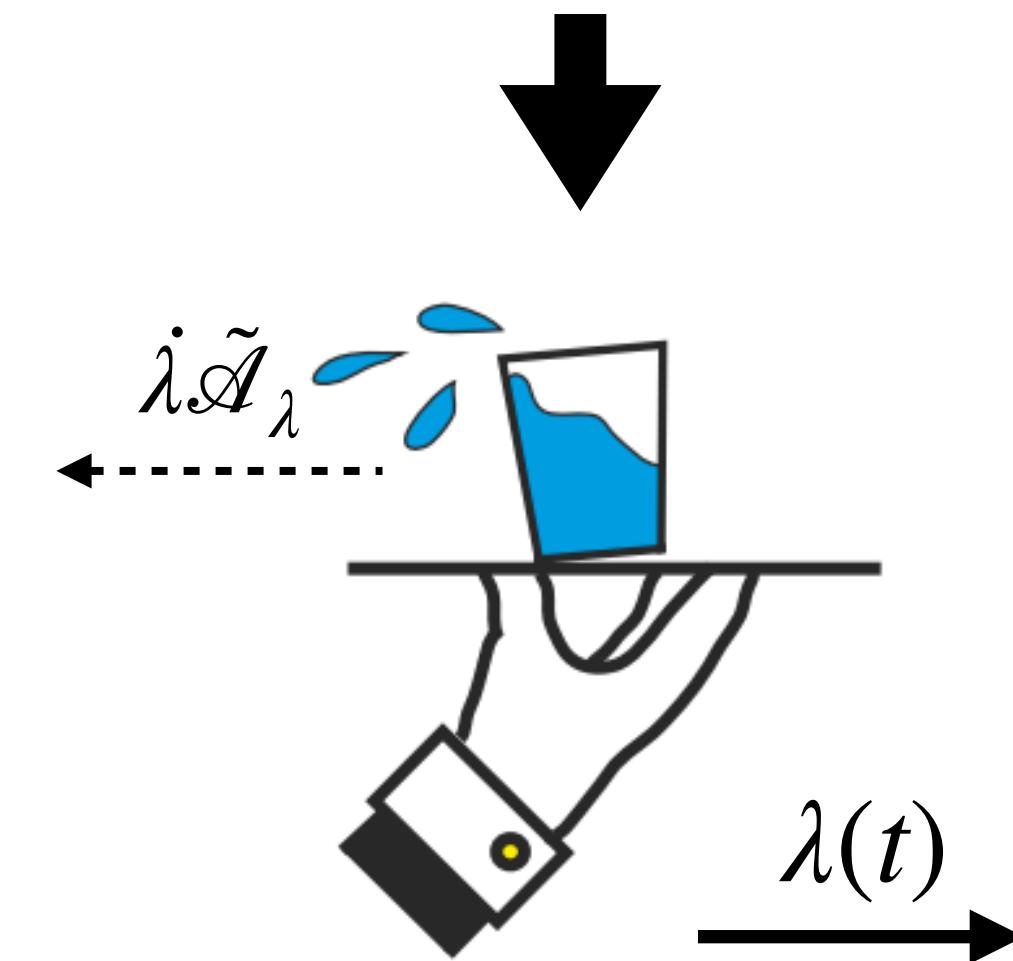
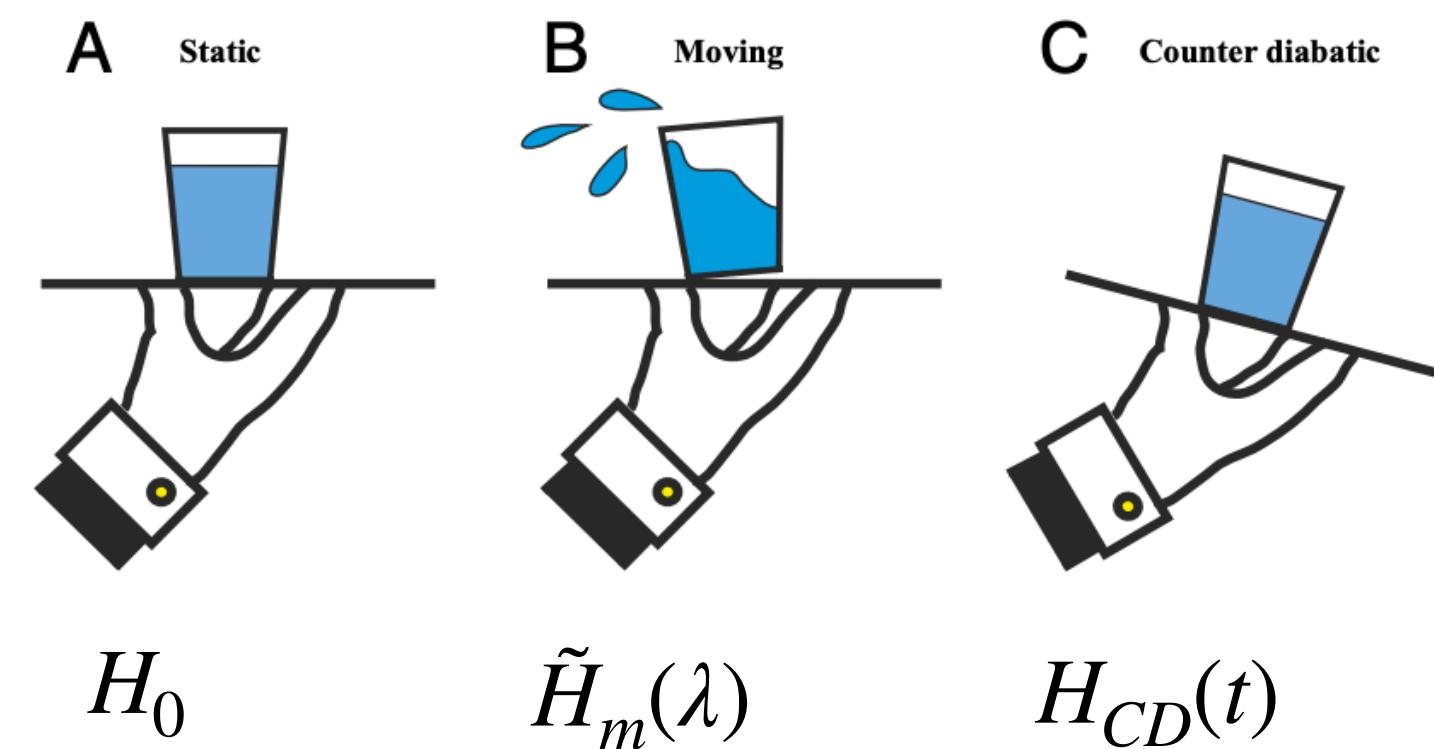


2. Add control pulse:

$$H_\beta(\lambda, \beta) = H(\lambda) + \beta \mathcal{O}_{\text{opt}}$$

3. Add LCD Ansatz:

$$H_{\text{COLD}}(\lambda, \beta) = H_\beta(\lambda, \beta) + \sum_j \alpha(\lambda, \beta) \mathcal{O}_{\text{LCD}}$$



Counterdiabatic optimised local driving (COLD)

1. Start with adiabatic protocol: $H(\lambda)$

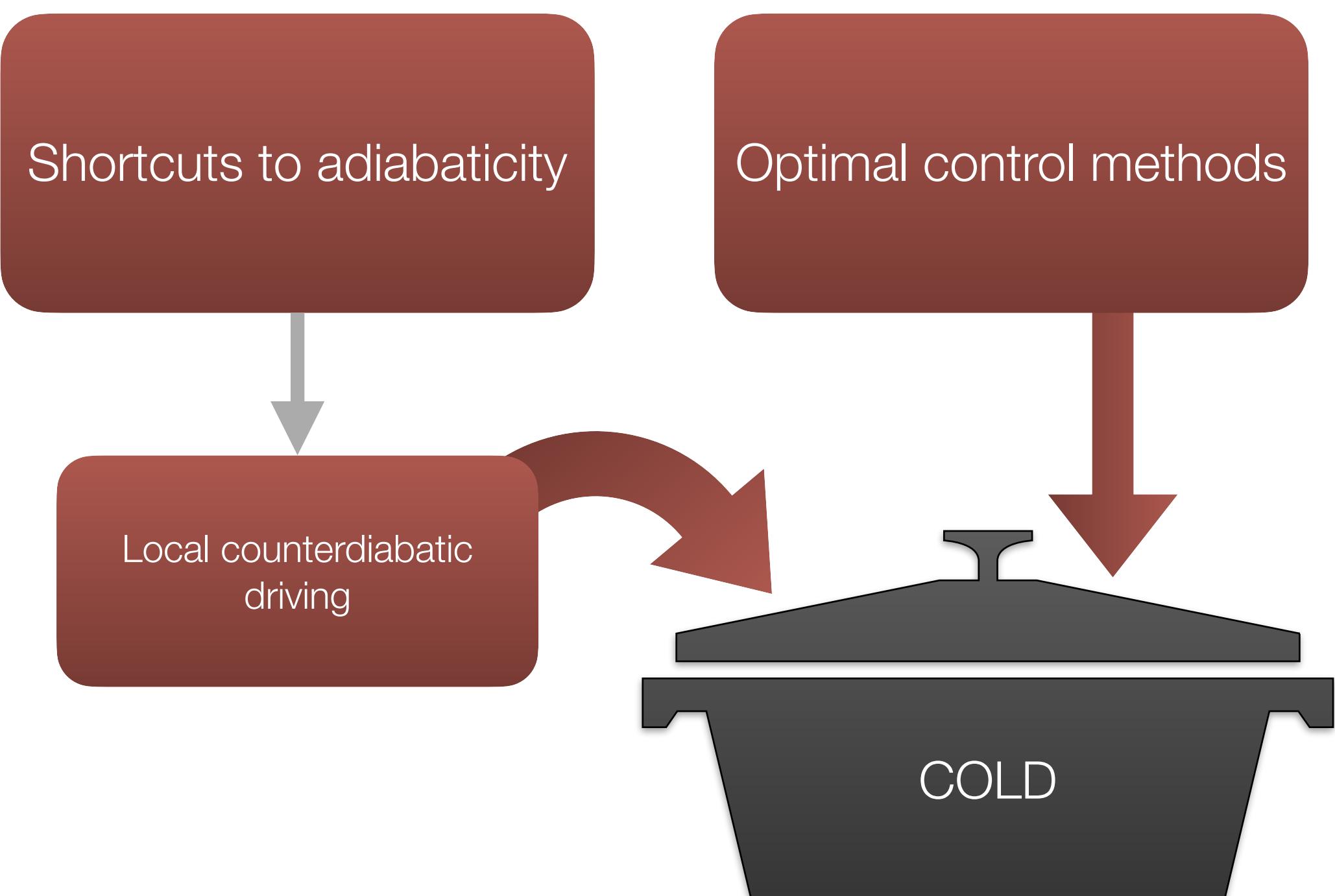


2. Add control pulse:

$$H_\beta(\lambda, \beta) = H(\lambda) + \beta \mathcal{O}_{\text{opt}}$$

3. Add LCD Ansatz:

$$H_{\text{COLD}}(\lambda, \beta) = H_\beta(\lambda, \beta) + \sum_j \alpha(\lambda, \beta) \mathcal{O}_{\text{LCD}}$$



Counterdiabatic optimised local driving (COLD)

1. Start with adiabatic protocol: $H(\lambda)$



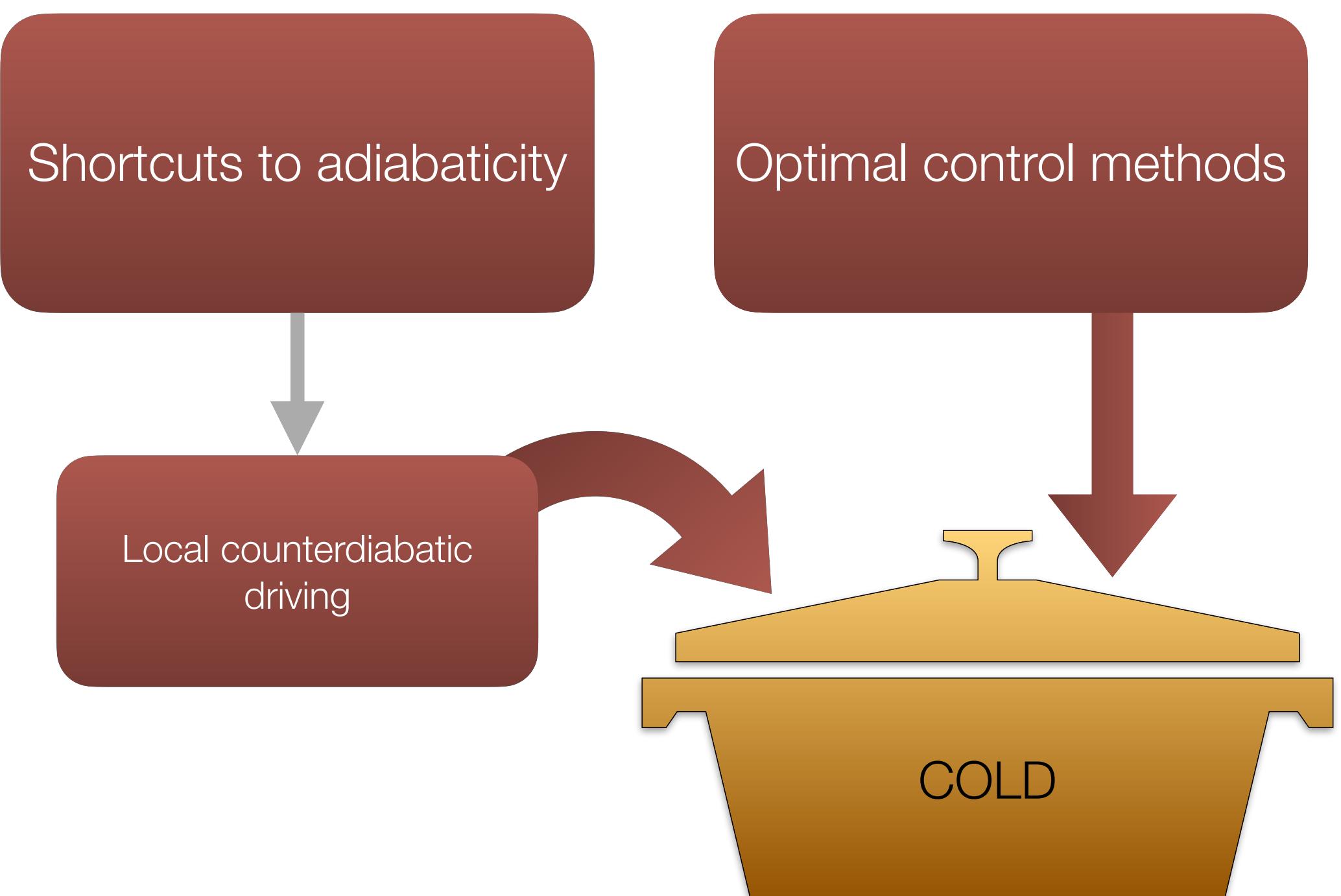
2. Add control pulse:

$$H_{\beta}(\lambda, \beta) = H(\lambda) + \beta \mathcal{O}_{\text{opt}}$$

3. Add LCD Ansatz:

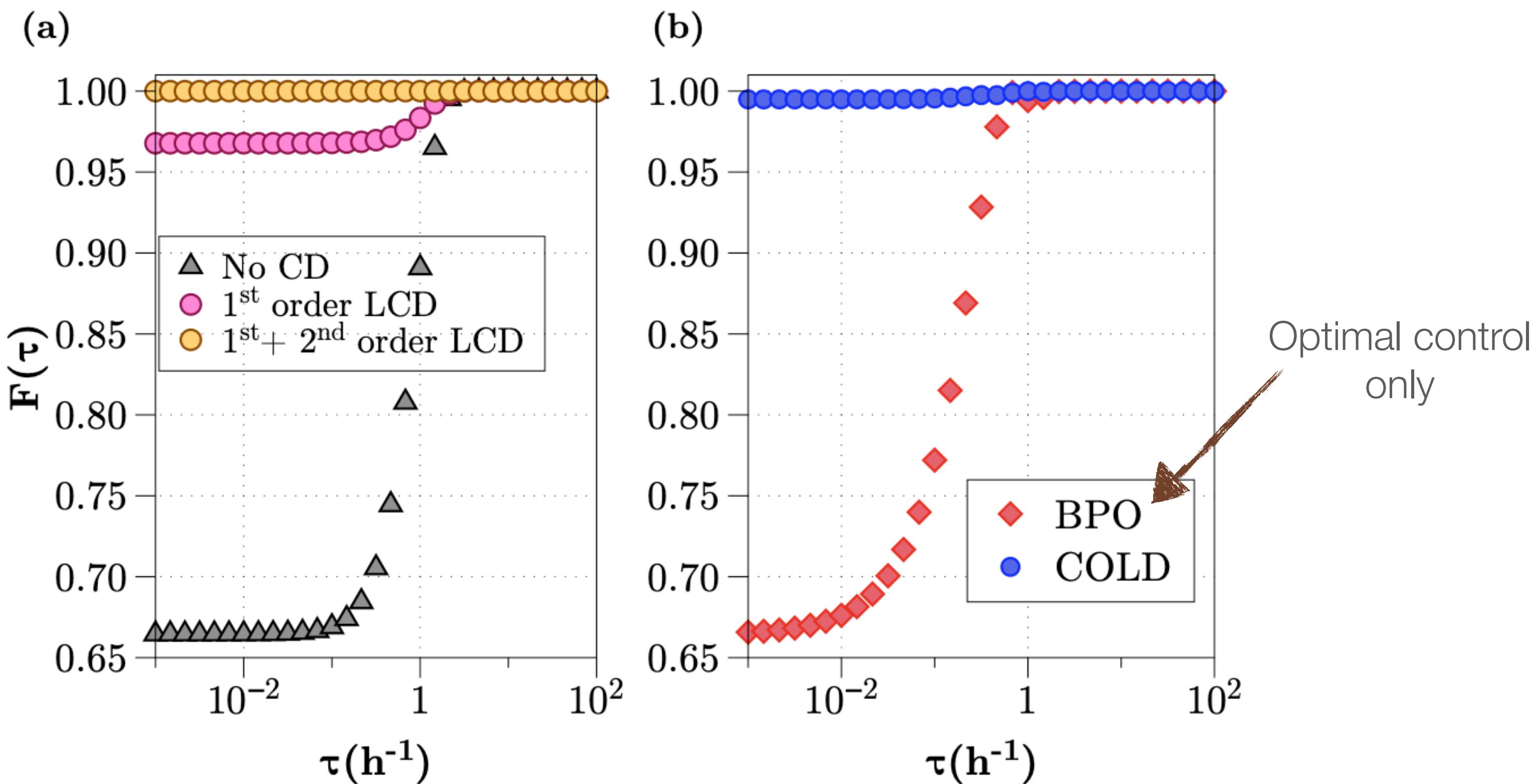
$$H_{\text{COLD}}(\lambda, \beta) = H_{\beta}(\lambda, \beta) + \sum_j \alpha(\lambda, \beta) \mathcal{O}_{\text{LCD}}$$

4. Optimise parameters β

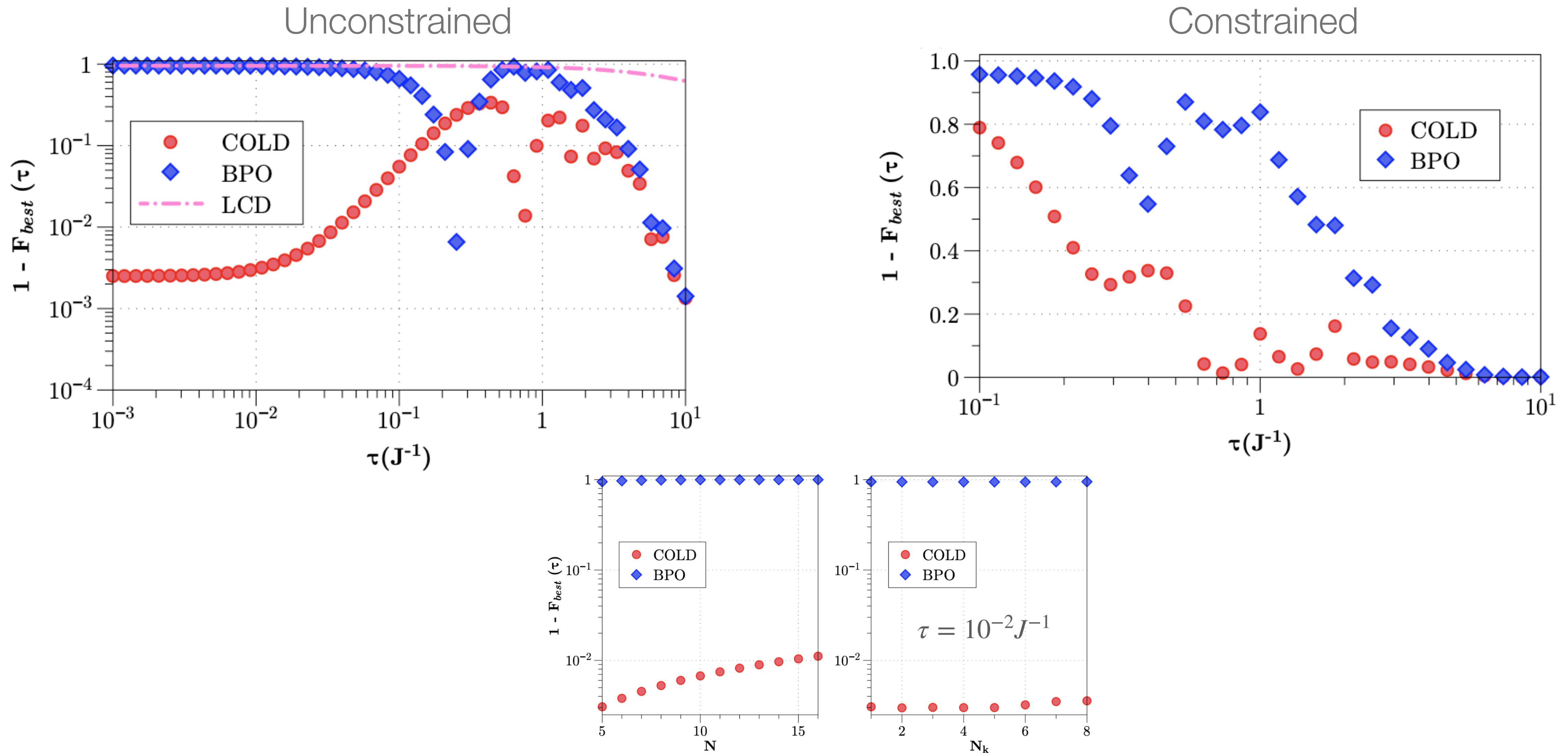


Result: 2-spin annealing

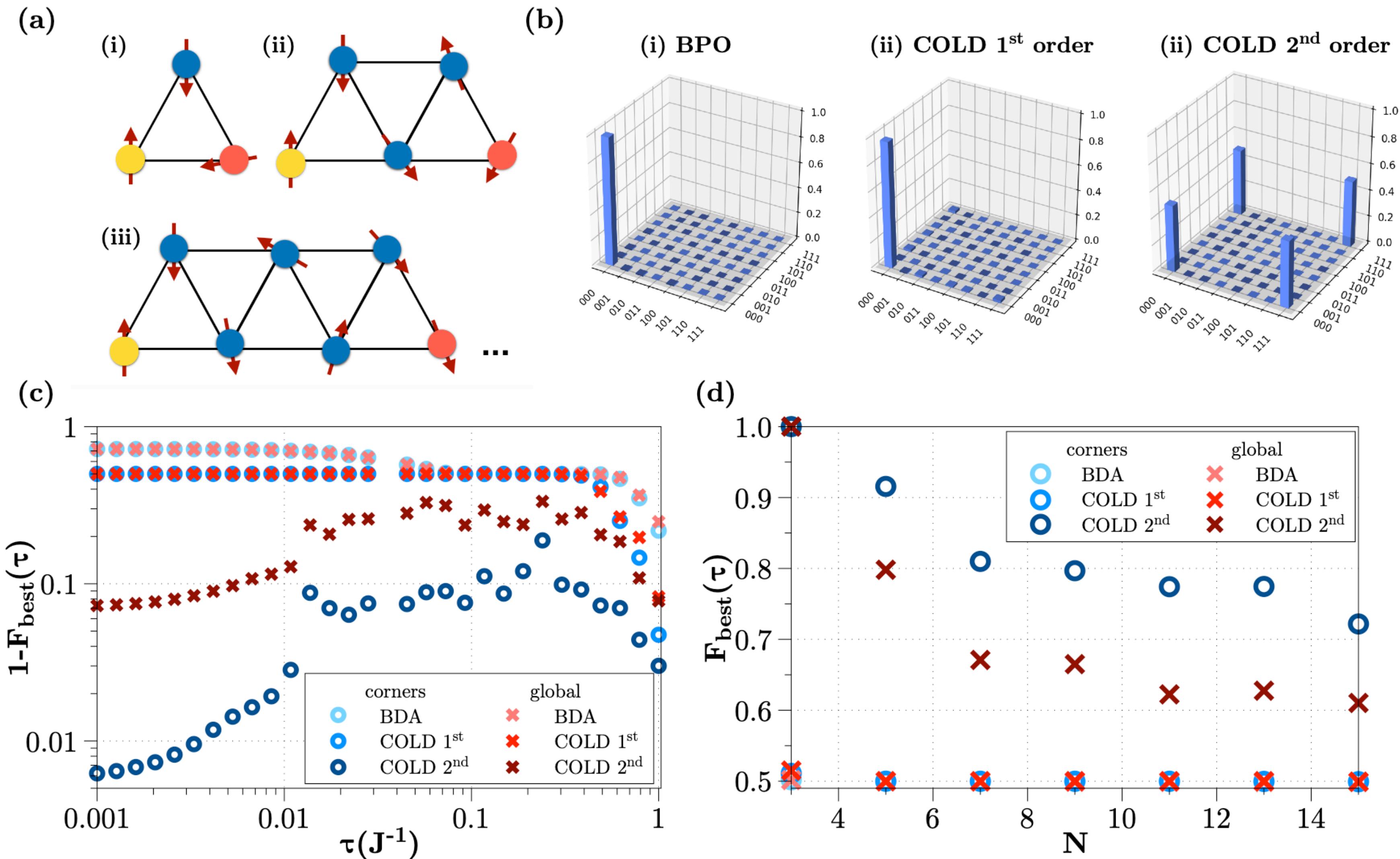
$$H(\lambda) = -2J\sigma_1^z\sigma_2^z - h(\sigma_1^z + \sigma_2^z) + 2h\lambda(t)(\sigma_1^x + \sigma_2^x)$$



Result: Ising chain phase transition

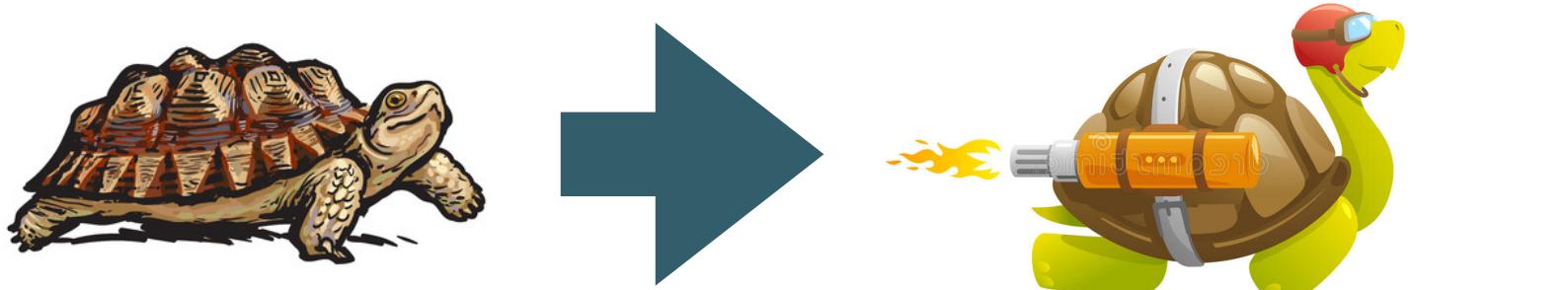


Result: GHZ state preparation in a system of frustrated spins



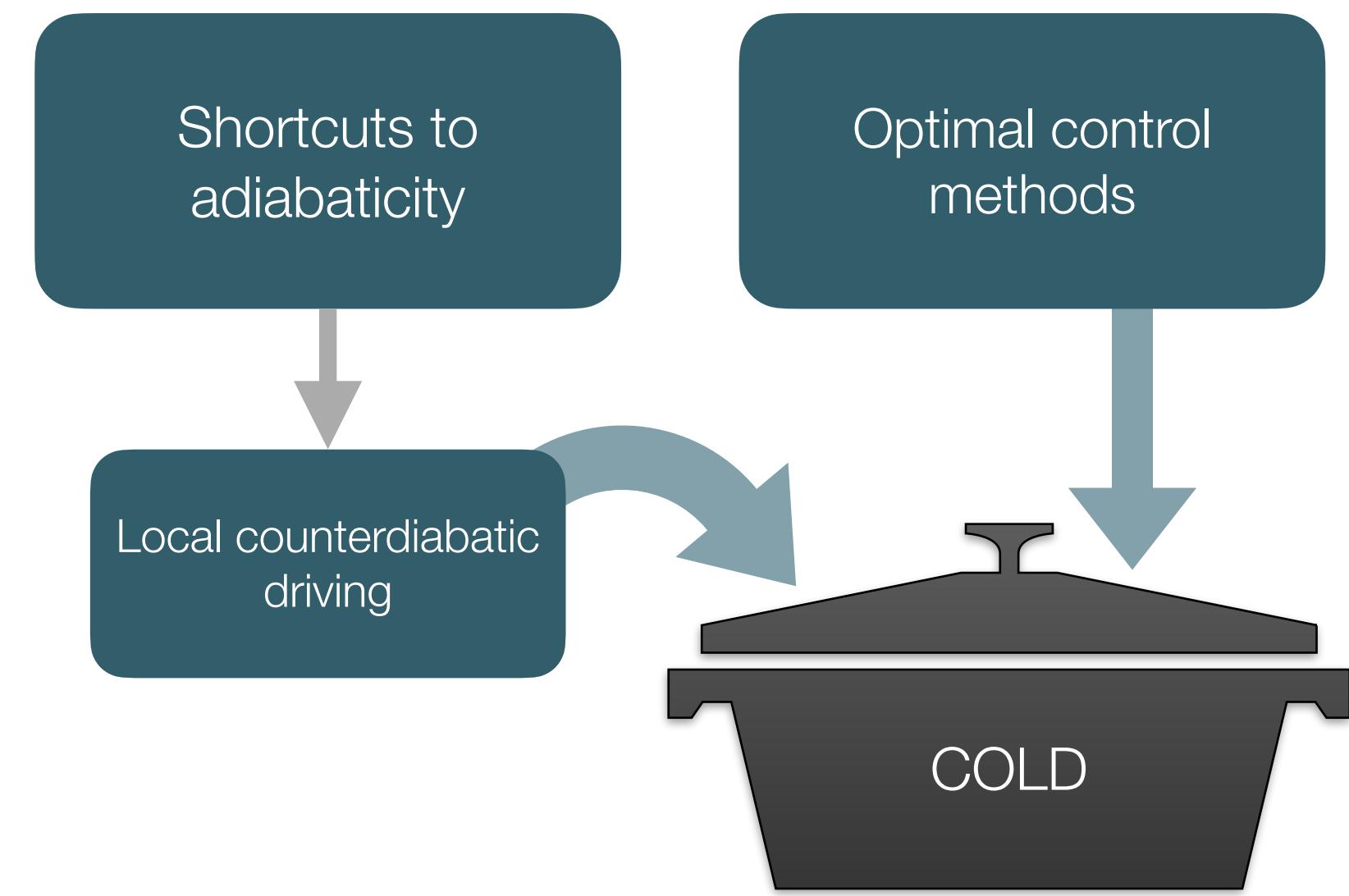
Summary

- **COLD**: new method for speeding up adiabatic processes
- Improves upon LCD and optimal control methods
- Outlook
 - Better understanding of AGP to inform/improve optimisation
 - Working on implementing with cold atoms for optimisation problems!

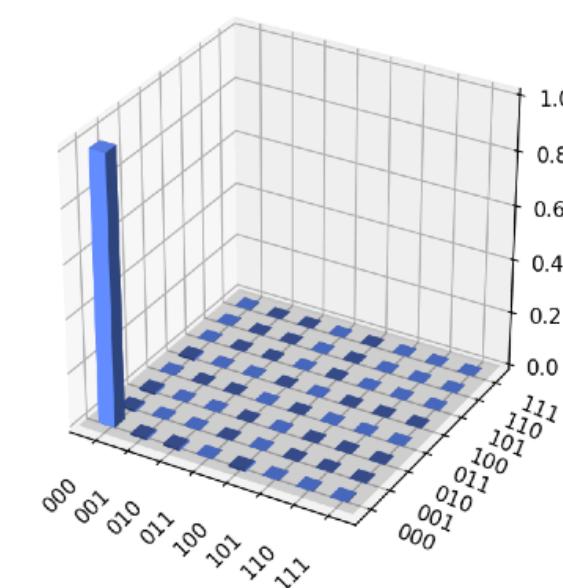


arXiv:2203.01948

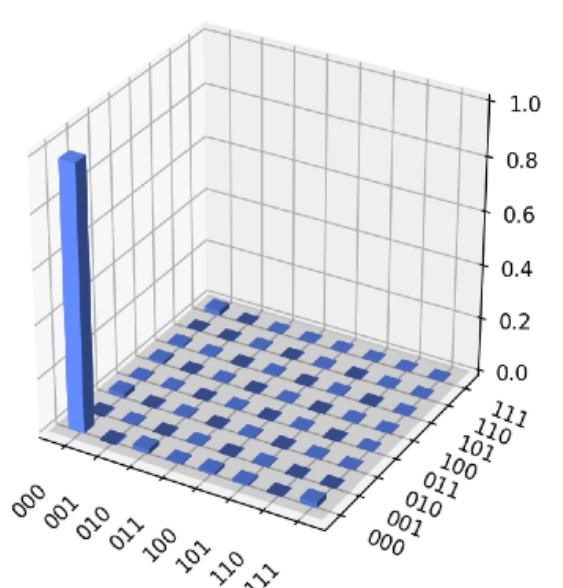
accepted in PRX Quantum



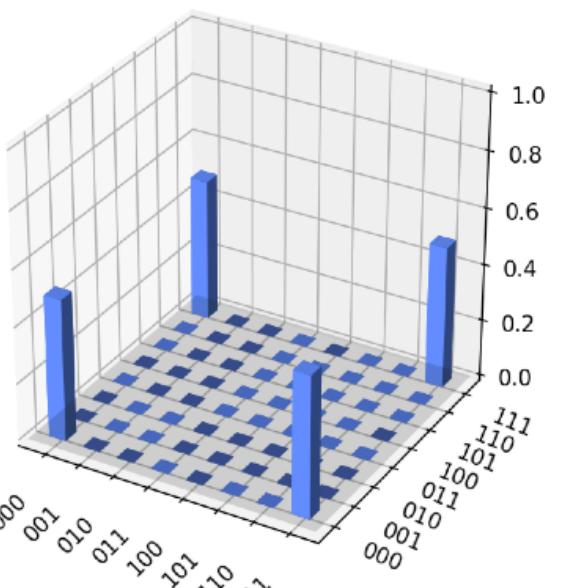
(i) BPO



(ii) COLD 1st order



(iii) COLD 2nd order



@Hyperboleva



@hyperboleva@mathstodon.xyz