[PRX Quantum 4, 010312]

Counterdiabatic Optimised Local Driving
(COLD)

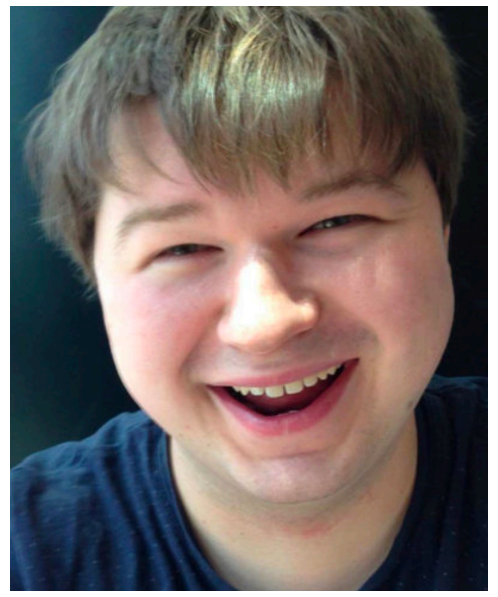
Ieva Čepaitė, Andrew J. Daley, Callum W. Duncan

University of Strathclyde, Glasgow

Anatoli Polkovnikov

Boston University, Boston



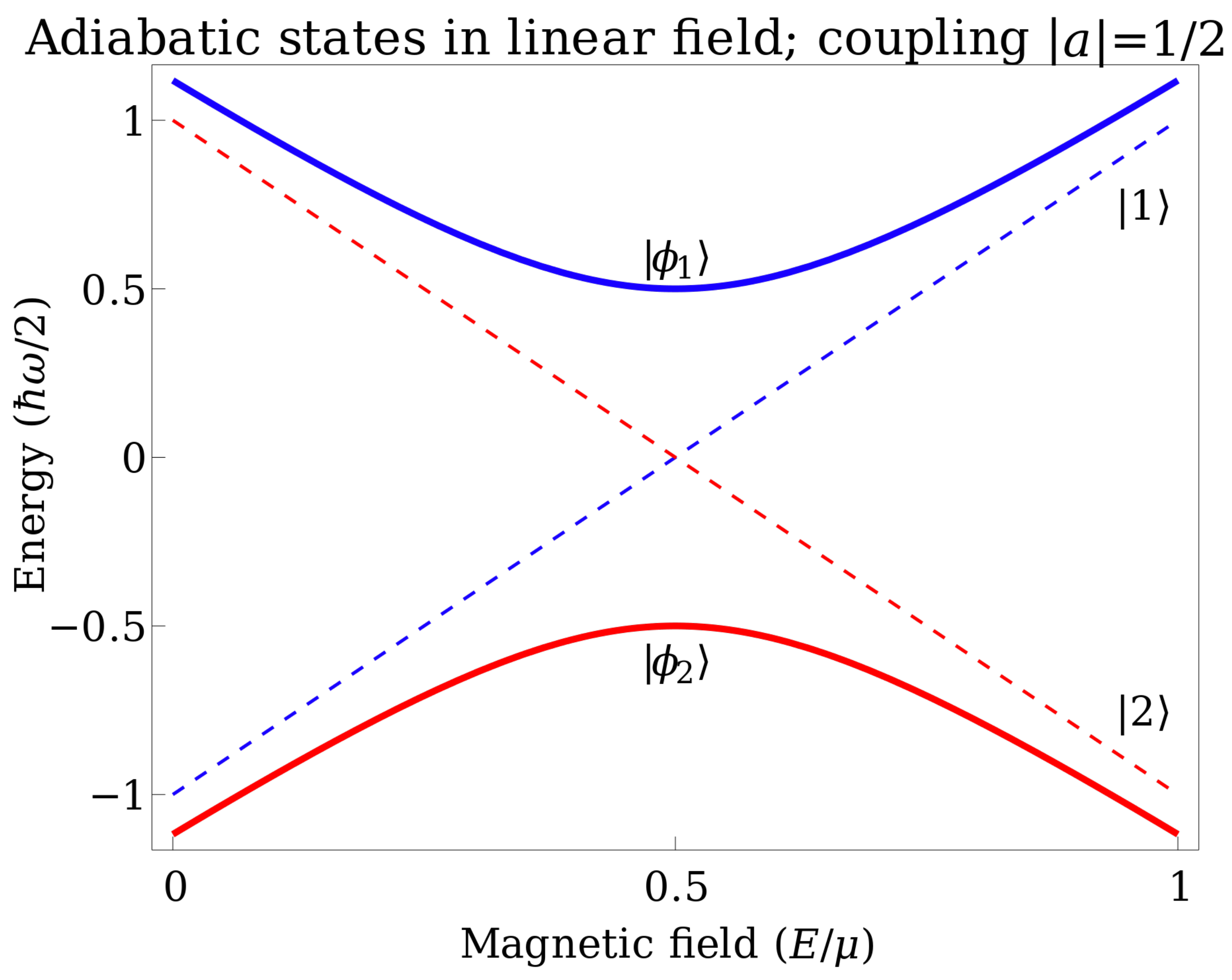



Adiabatic processes

- **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…

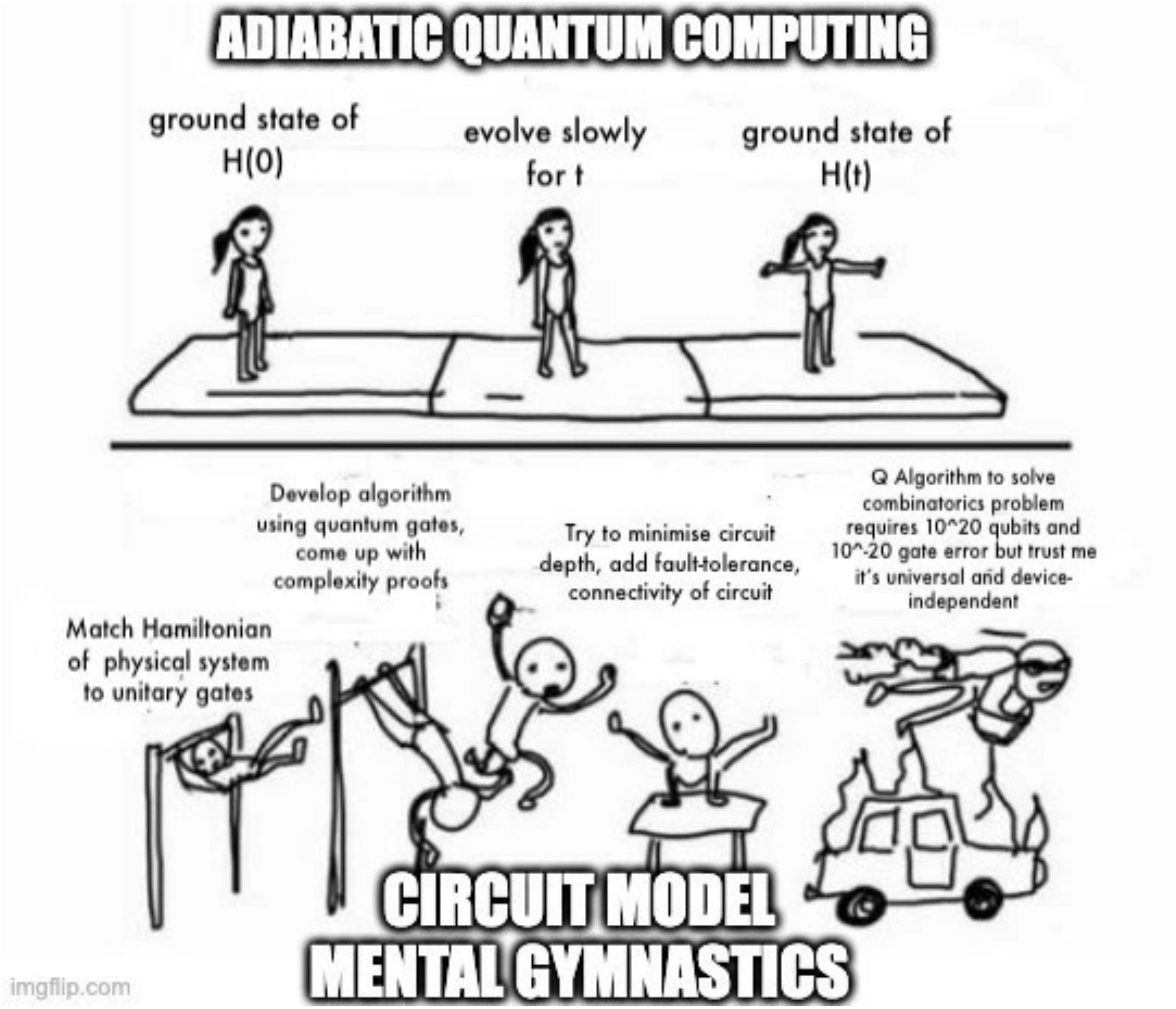


https://en.wikipedia.org/wiki/Adiabatic_theorem

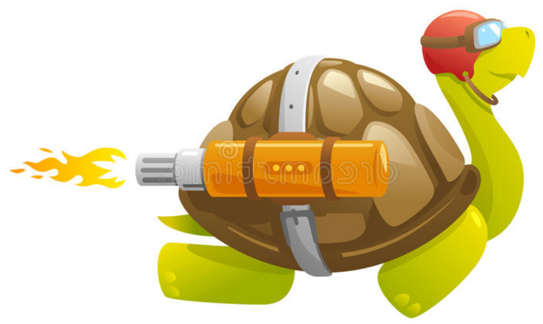
!

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

Adiabatic processes (extra motivation)

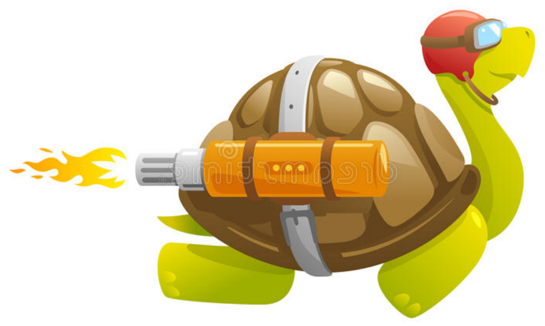


Speeding things up



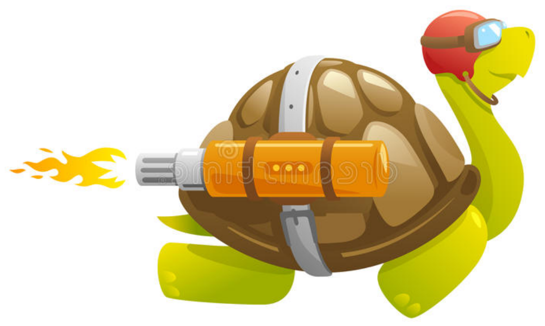
- What do we want?

Speeding things up



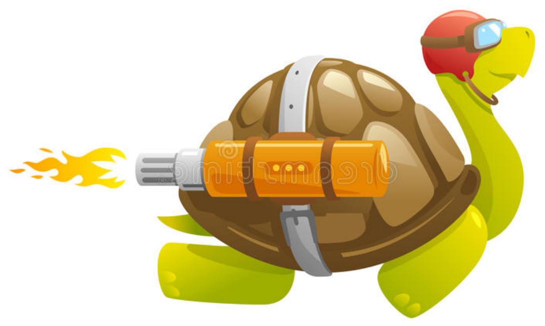
- What do we want?
  - Adiabatic dynamics!

Speeding things up



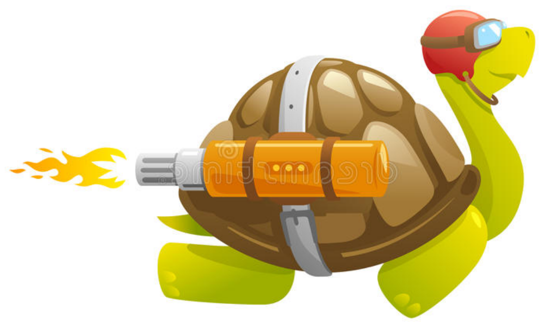
- What do we want?
    - Adiabatic dynamics!
- When do we want them?

Speeding things up



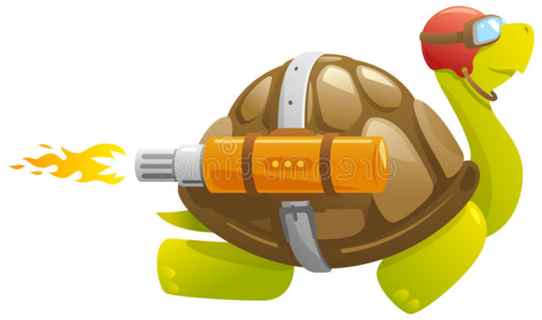
- What do we want?
  - Adiabatic dynamics!
- When do we want them?
  - As quickly as possible!

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

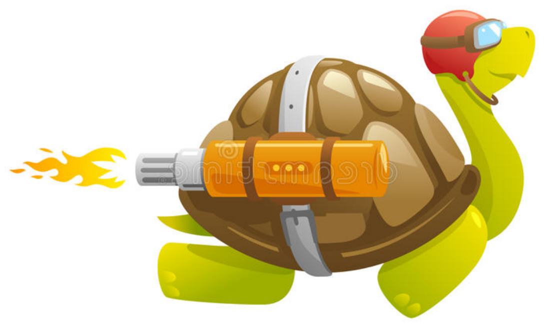
Shortcuts to adiabaticity

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

Optimal control methods

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

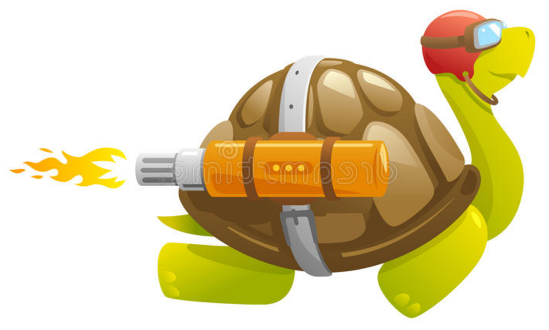
↓

Local counterdiabatic driving

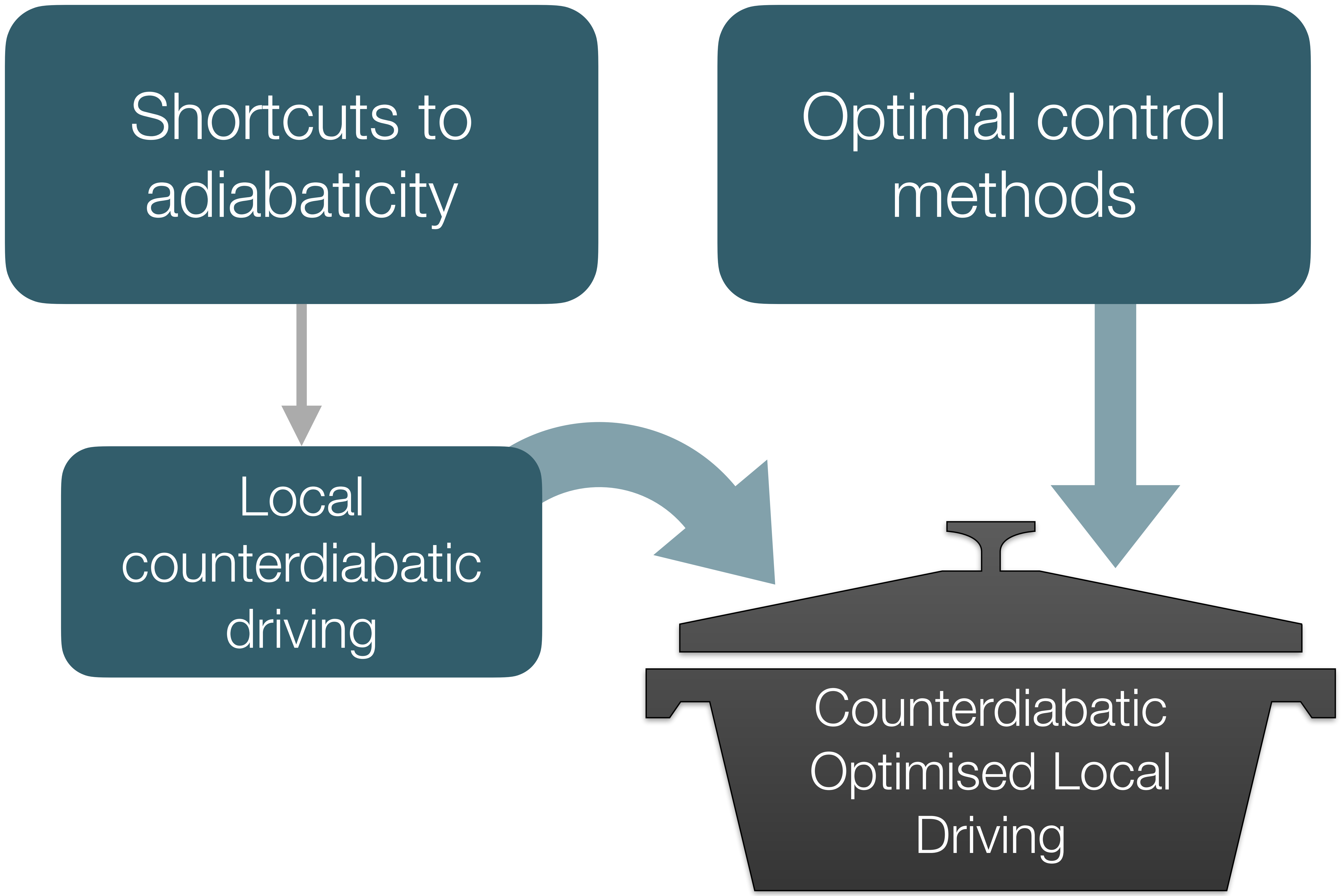
PNAS 114, E3909 (2017)

Optimal control methods

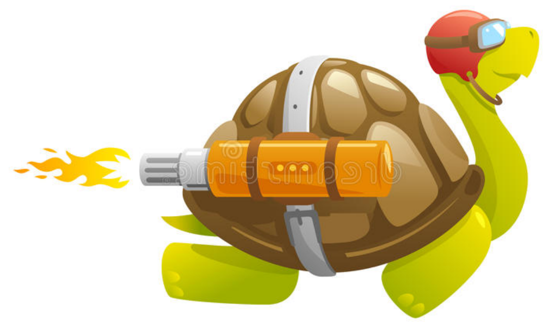
Speeding things up



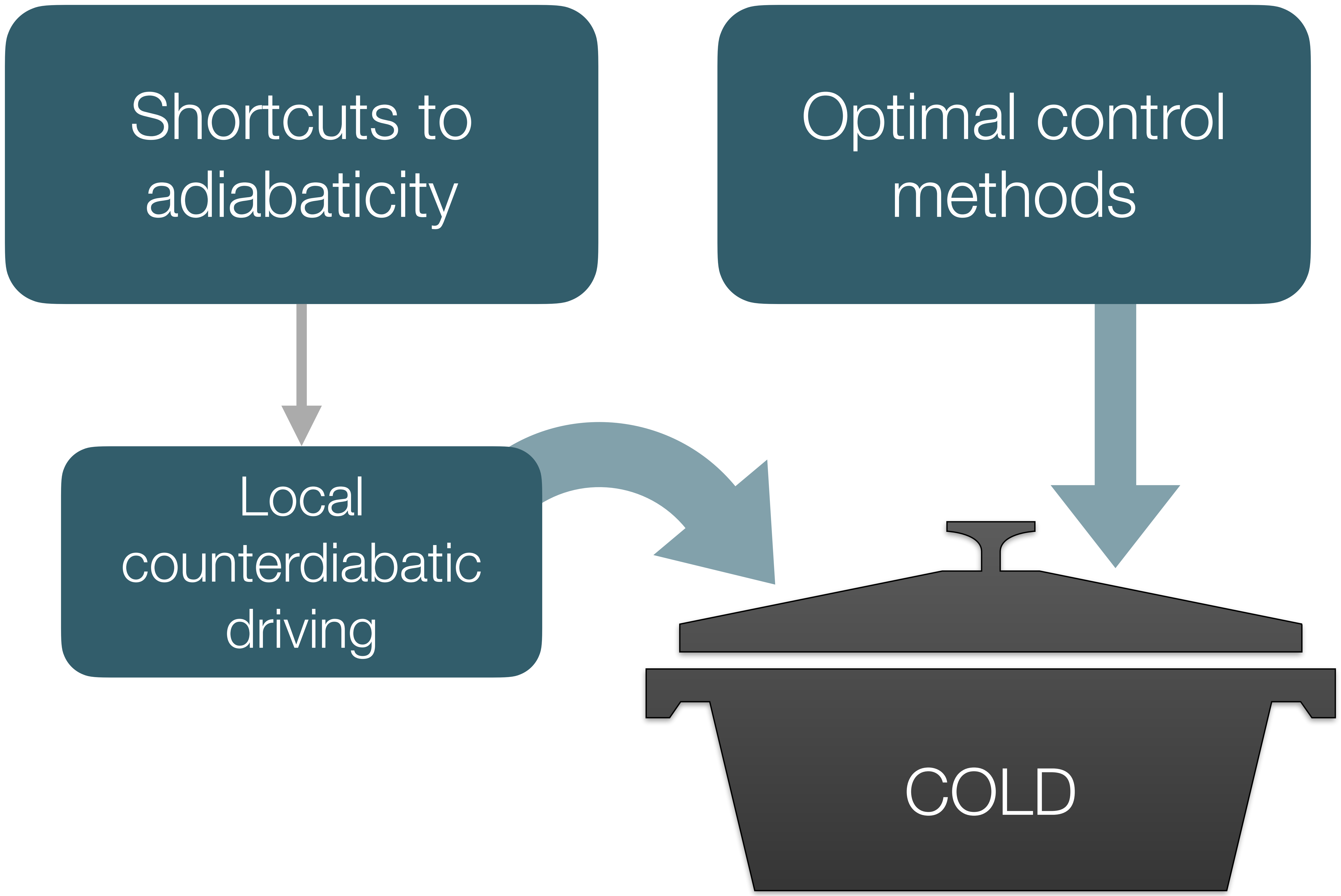
- What do we want?
  - Adiabatic dynamics!
- When do we want them?
  - As quickly as possible!
    (without losses/transitions)



Speeding things up



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

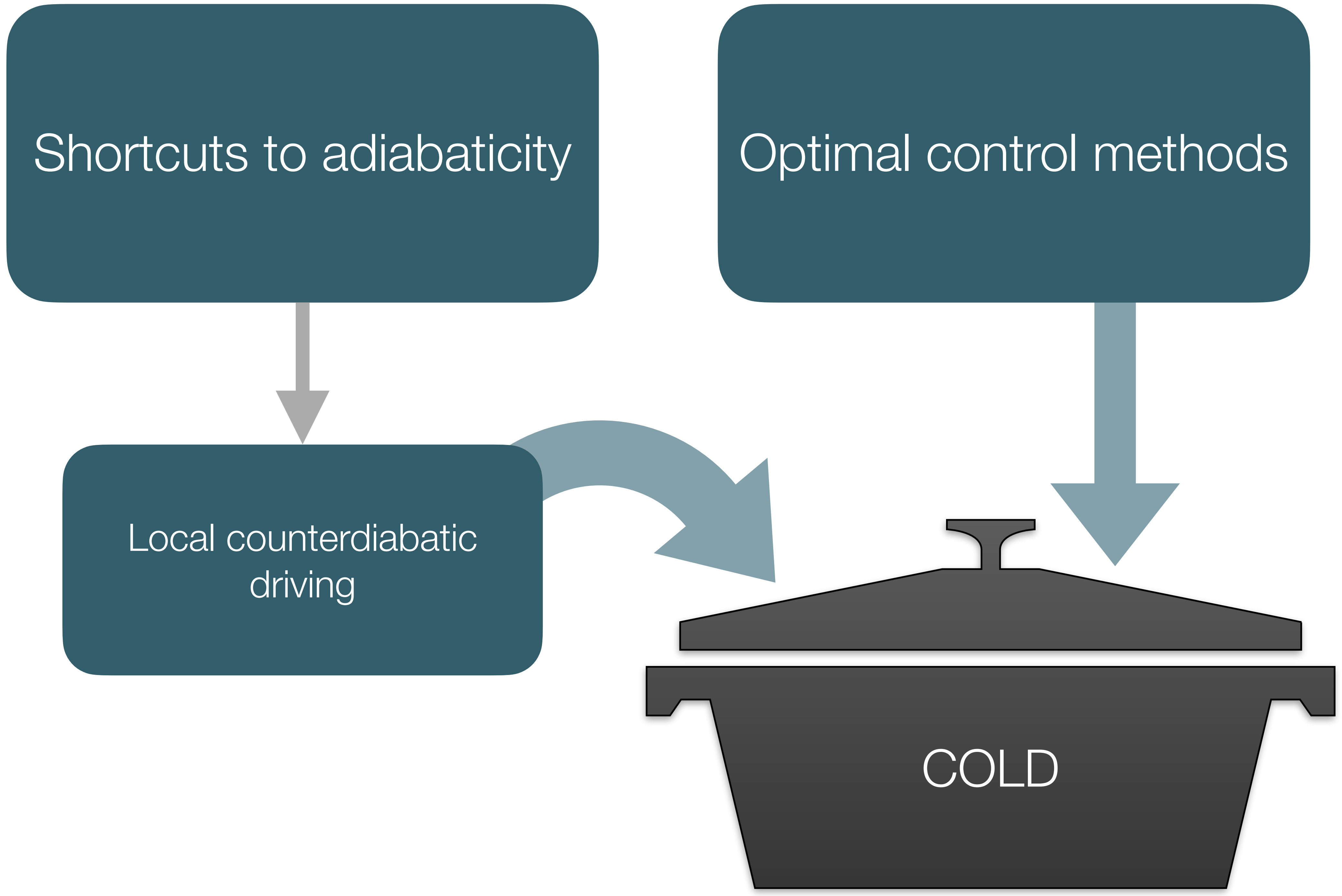


Counterdiabatic optimised local driving (COLD)

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

Counterdiabatic optimised local driving (COLD)

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

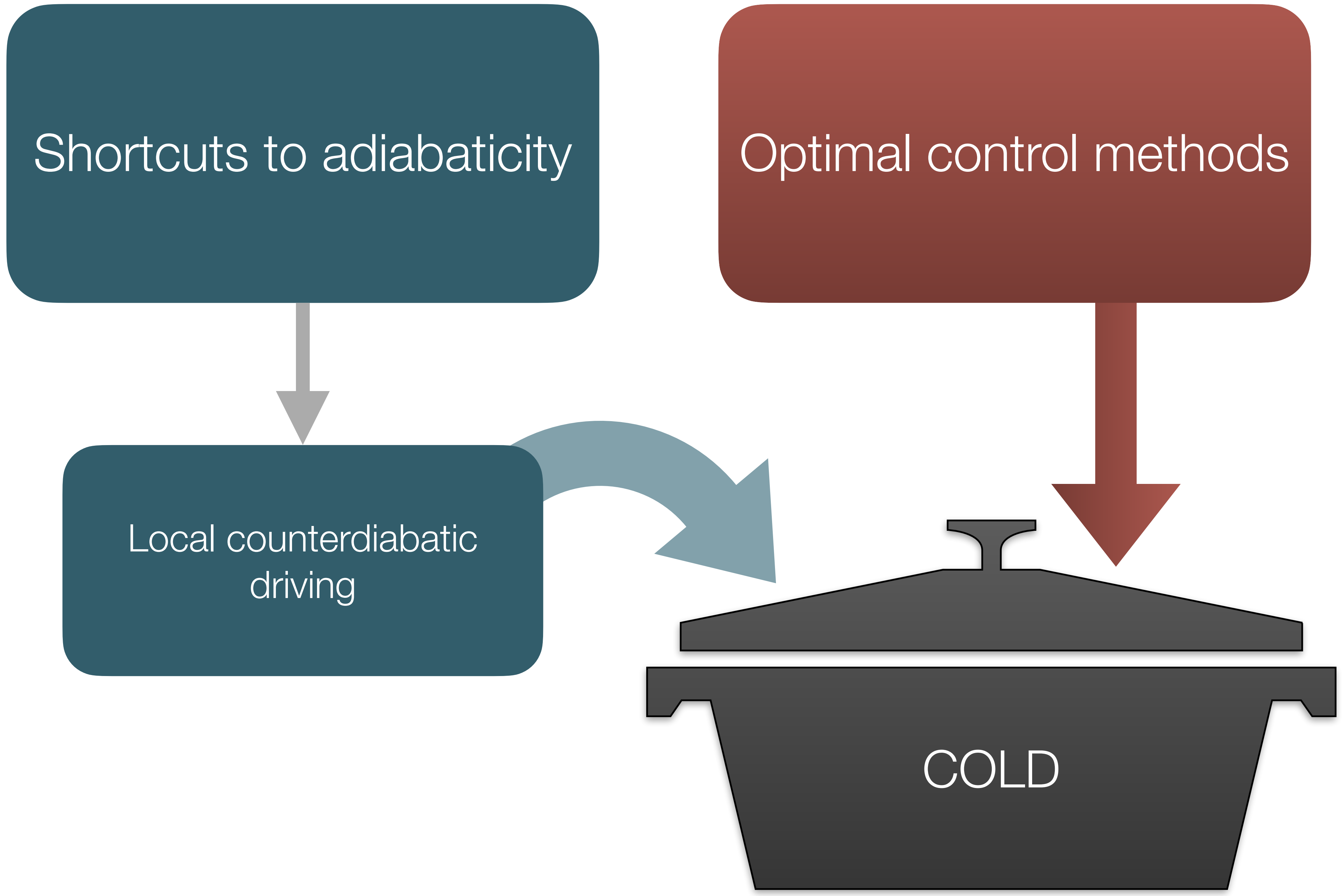


Counterdiabatic optimised local driving (COLD)

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

2. Add control pulse:

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

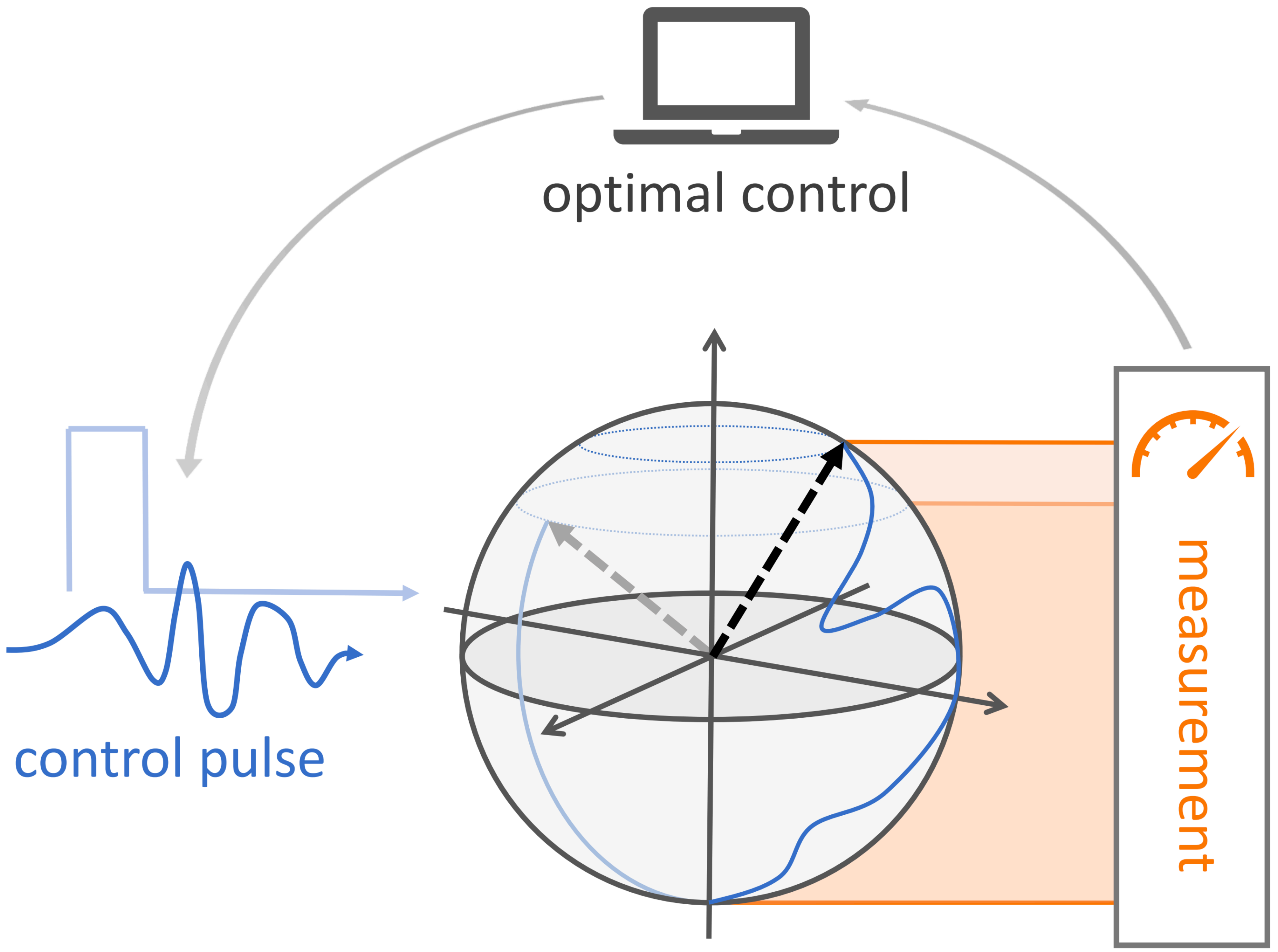


Counterdiabatic optimised local driving (COLD)

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

2. Add control pulse:

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



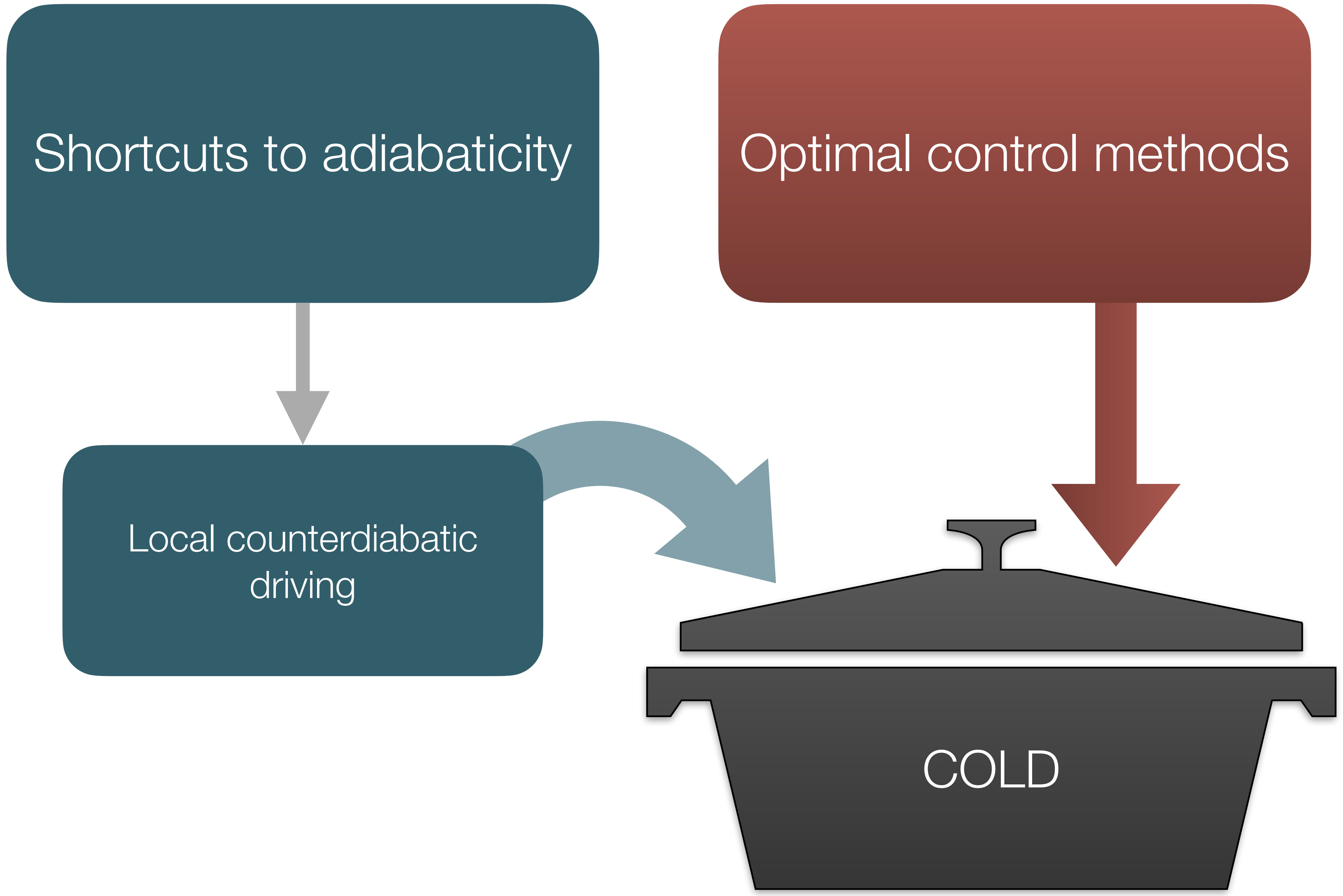
https://qusco-itn.eu/2020/06/19/intro_nvc/

Counterdiabatic optimised local driving (COLD)

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

2. Add control pulse:

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



Counterdiabatic optimised local driving (COLD)

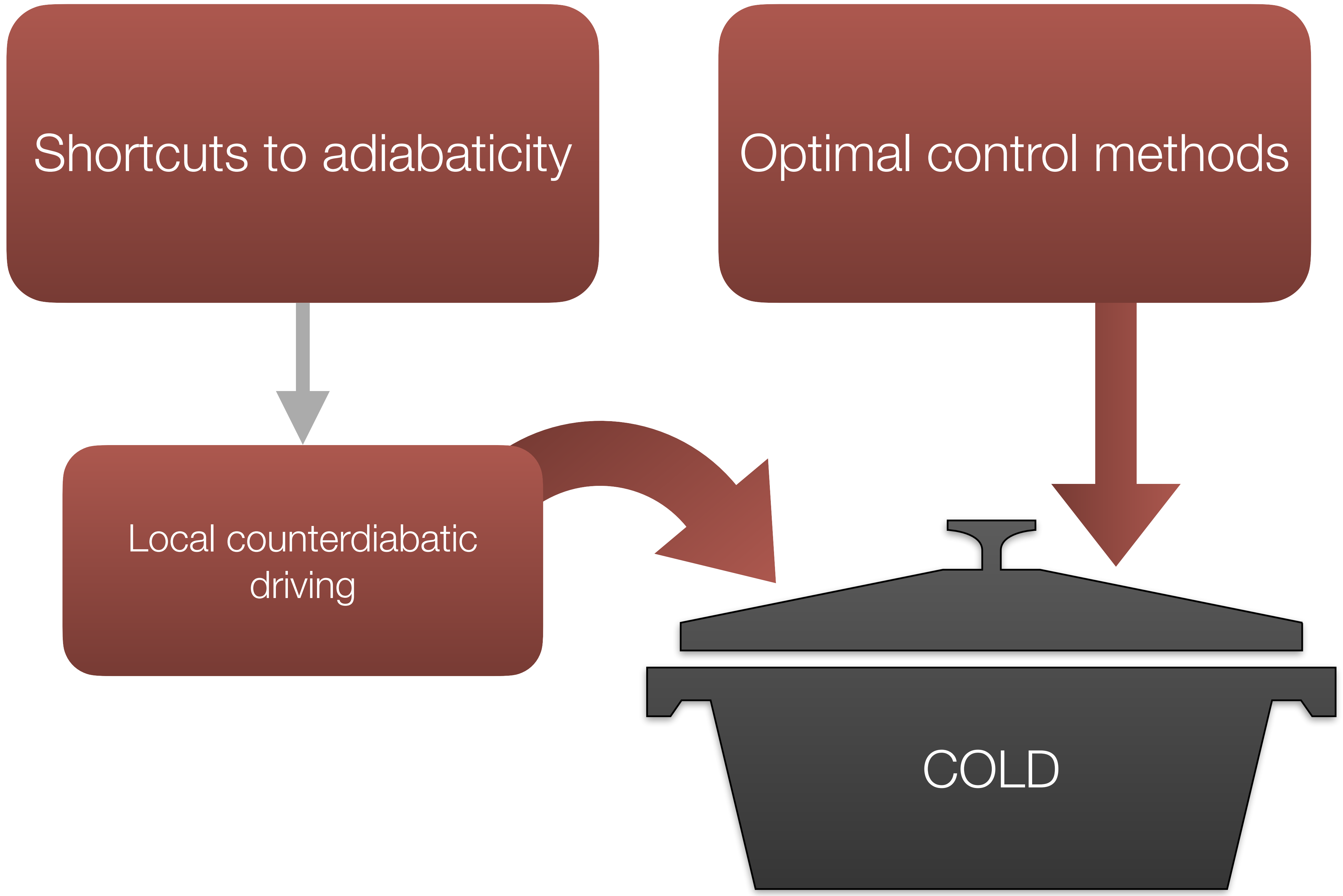
1. Start with adiabatic protocol: $H(\lambda)$

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)

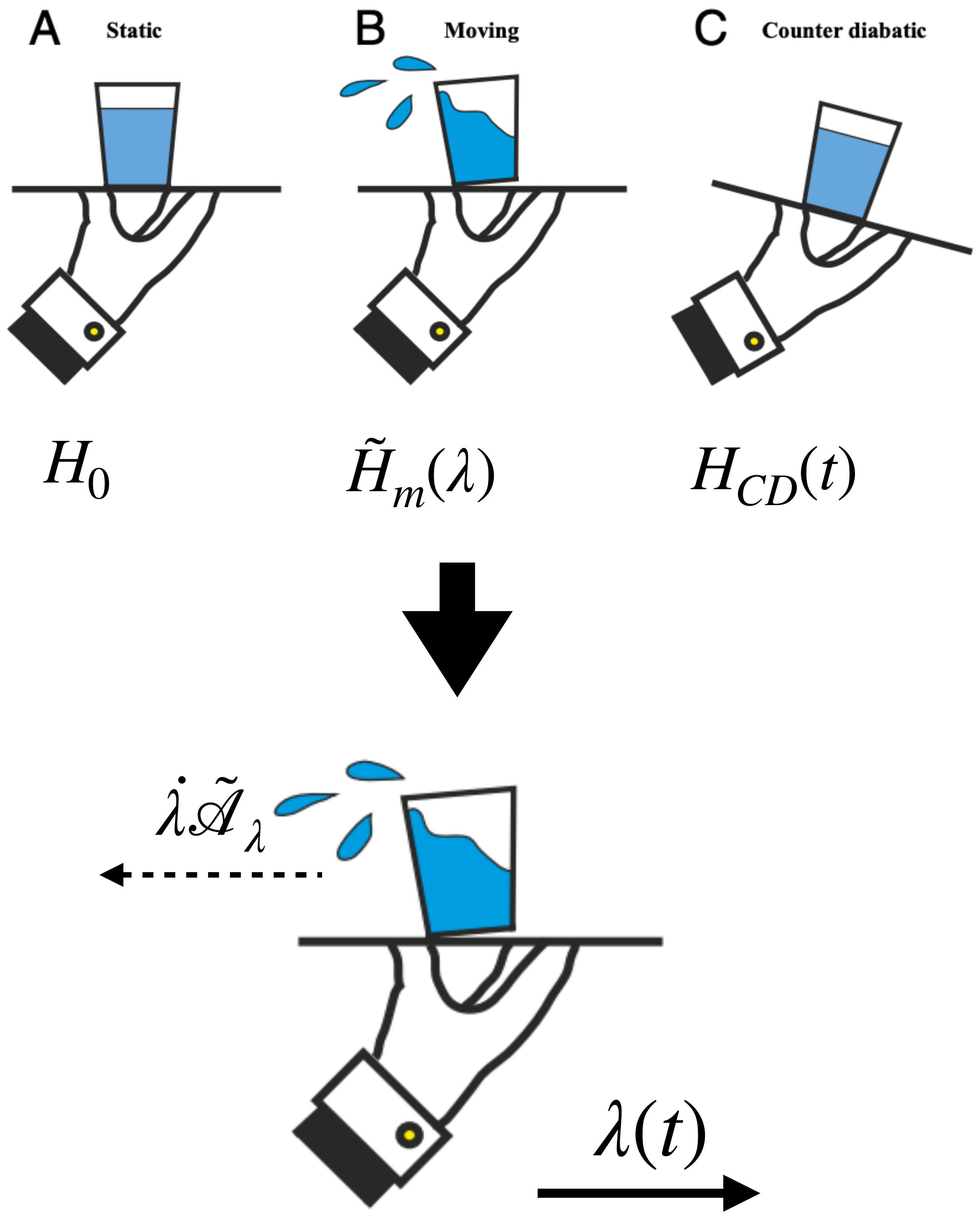
1. Start with adiabatic protocol: $H(\lambda)$

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)

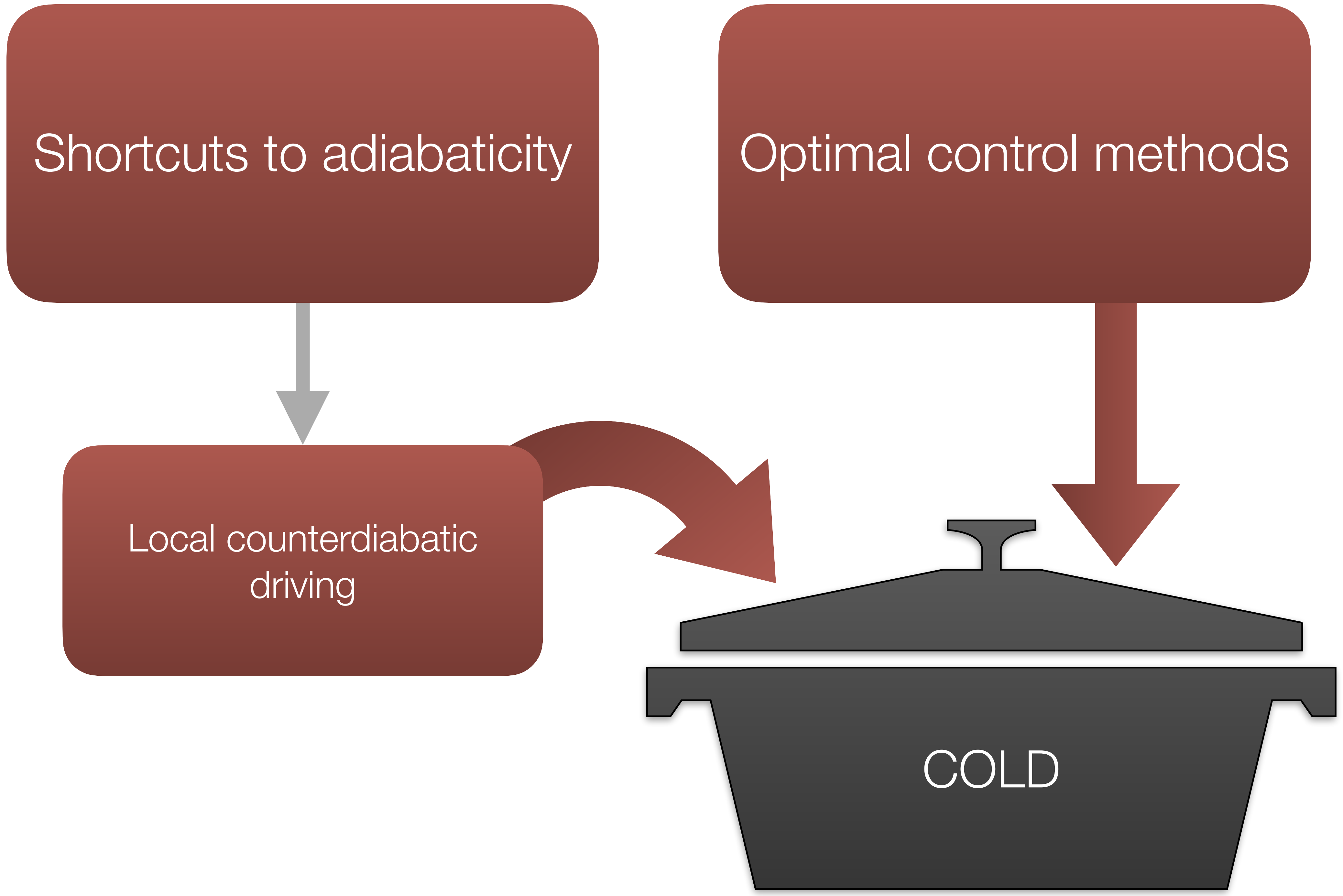
1. Start with adiabatic protocol: $H(\lambda)$

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: $H(\lambda)$

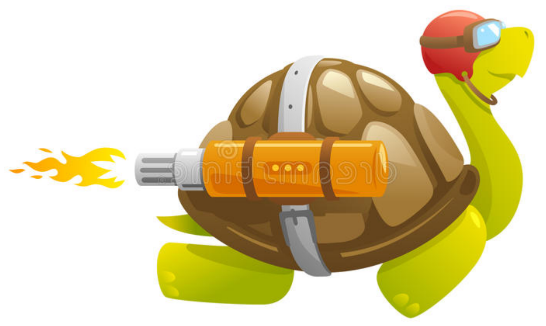
2. Add control pulse:

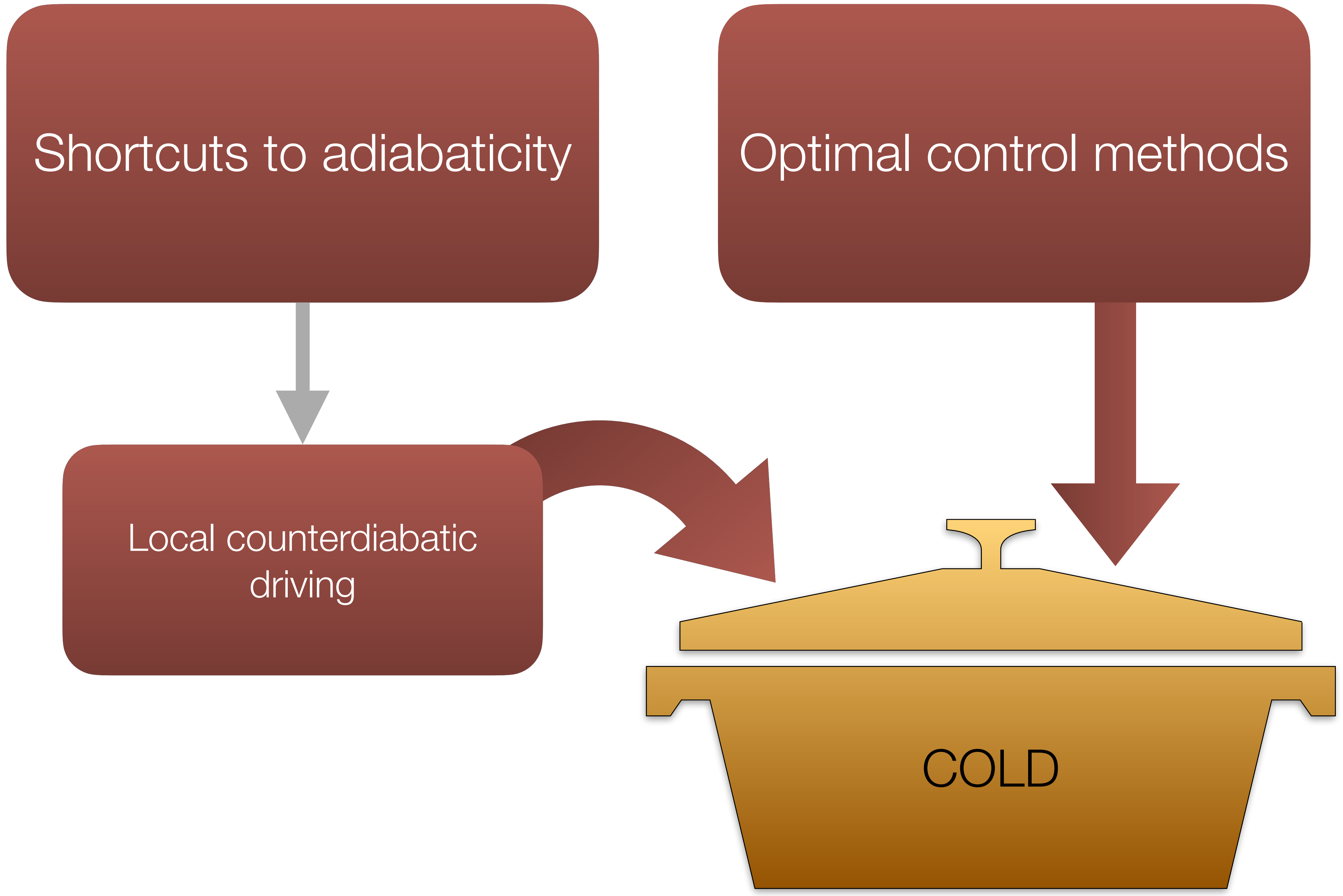
$$H_{\beta}(\lambda, \boldsymbol{\beta}) = H(\lambda) + \boldsymbol{\beta}\mathcal{O}_{\mathrm{opt}}$$

3. Add LCD Ansatz:

$$H_{\mathrm{COLD}}(\lambda, \boldsymbol{\beta}) = H_{\beta}(\lambda, \boldsymbol{\beta}) + \sum_{j} \alpha(\lambda, \boldsymbol{\beta})\mathcal{O}_{\mathrm{LCD}}$$

4. Optimise parameters $\boldsymbol{\beta}$





Result: 2-spin annealing

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

- 1st order LCD:

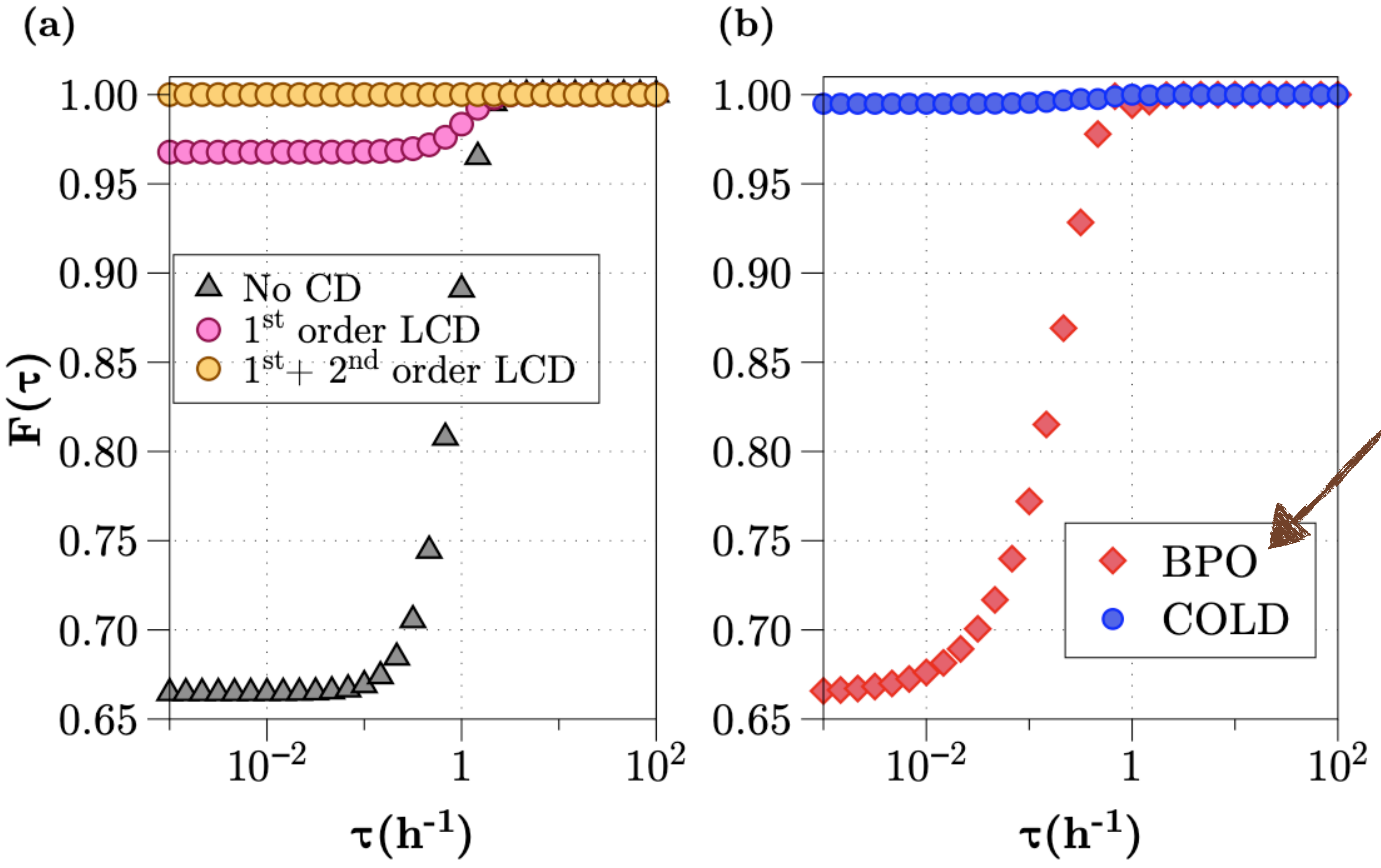
$$\alpha \sum_i \sigma_i^y$$

- 2nd order LCD:

$$\gamma \sum_{i,j} \sigma_{i(j)}^x \sigma_{j(i)}^y + \zeta \sum_{i,j} \sigma_{i(j)}^z \sigma_{j(i)}^y$$

- Control field:

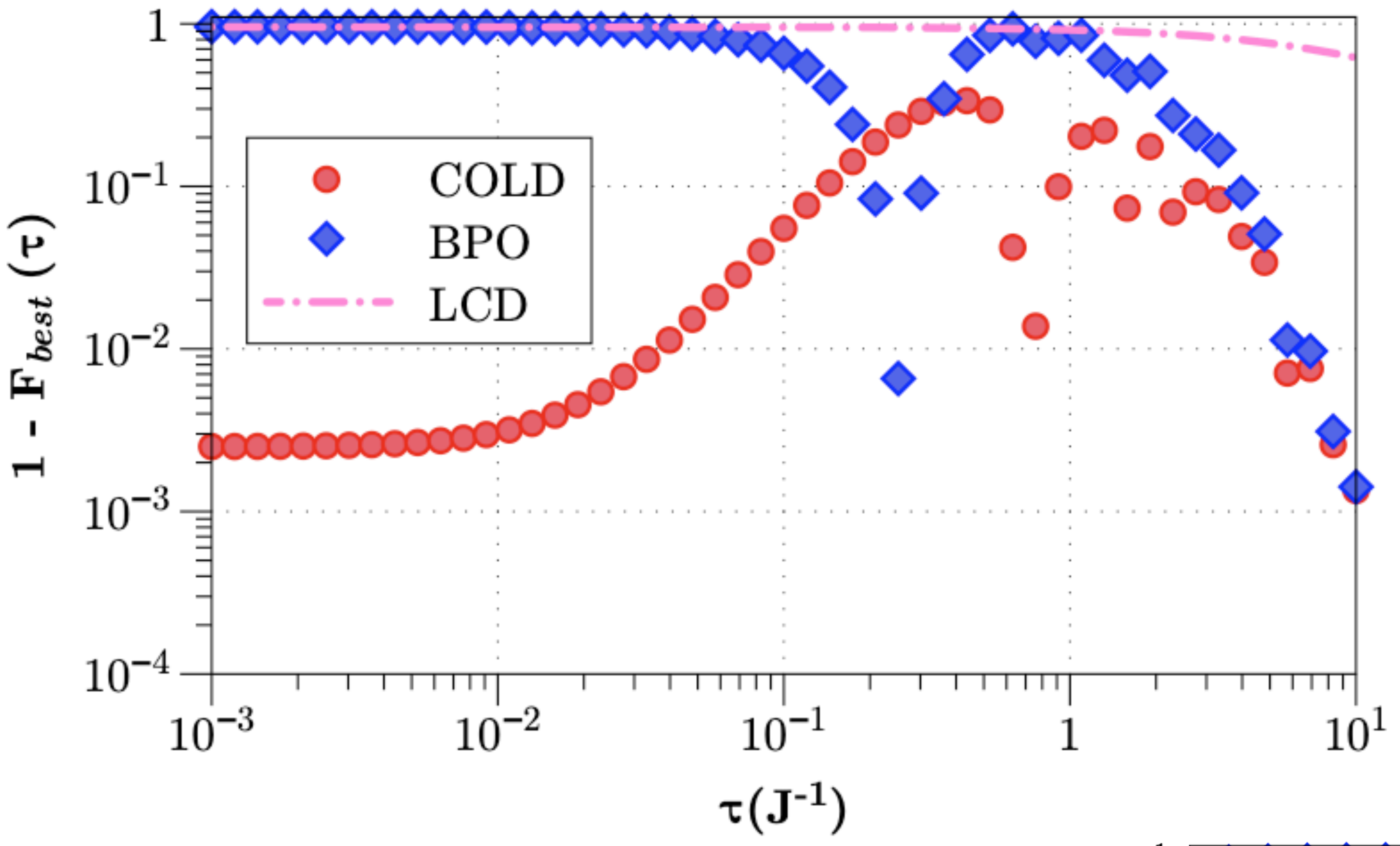
$$f(\lambda, \beta) \sum_i \sigma^z$$



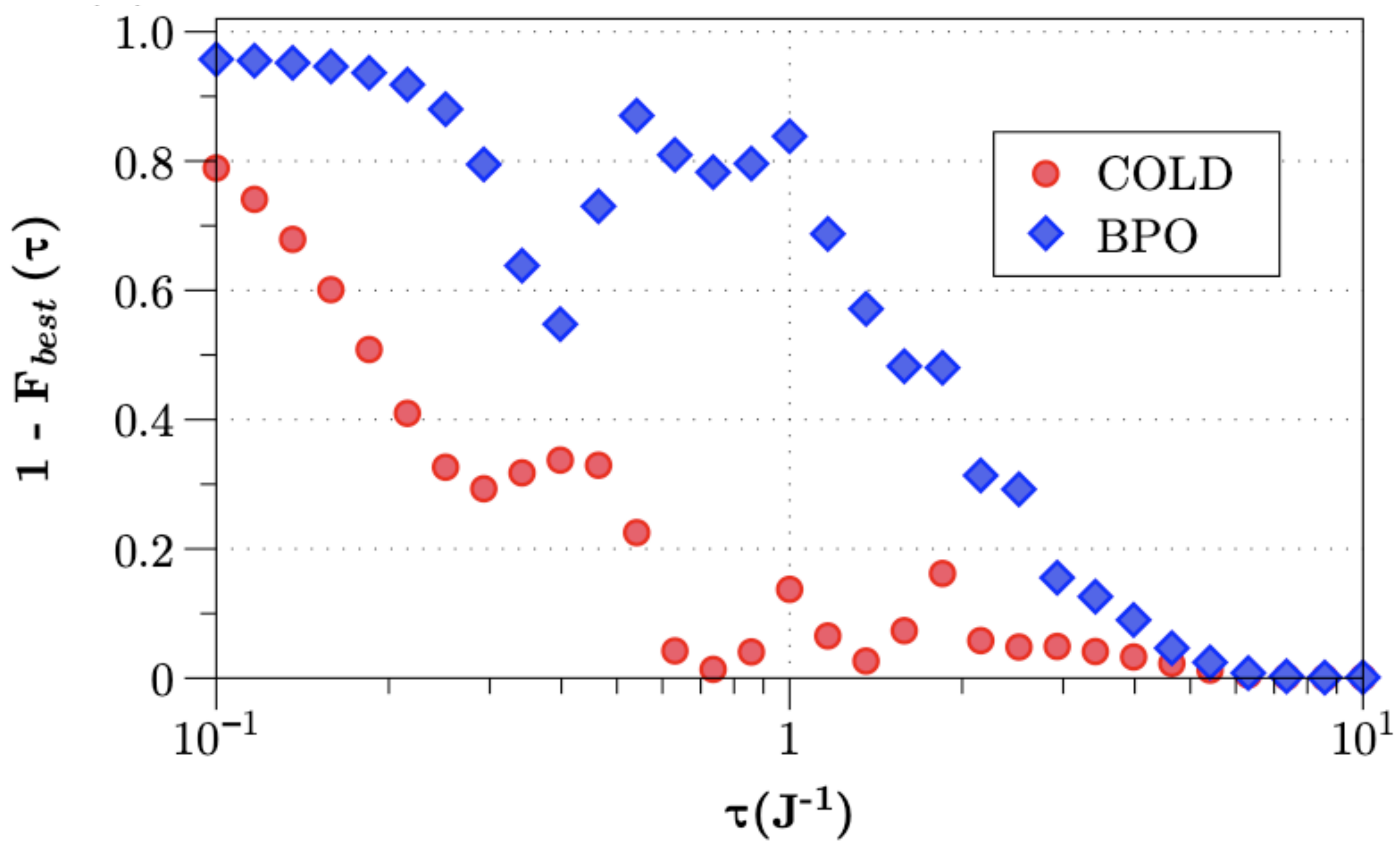
Optimal control only

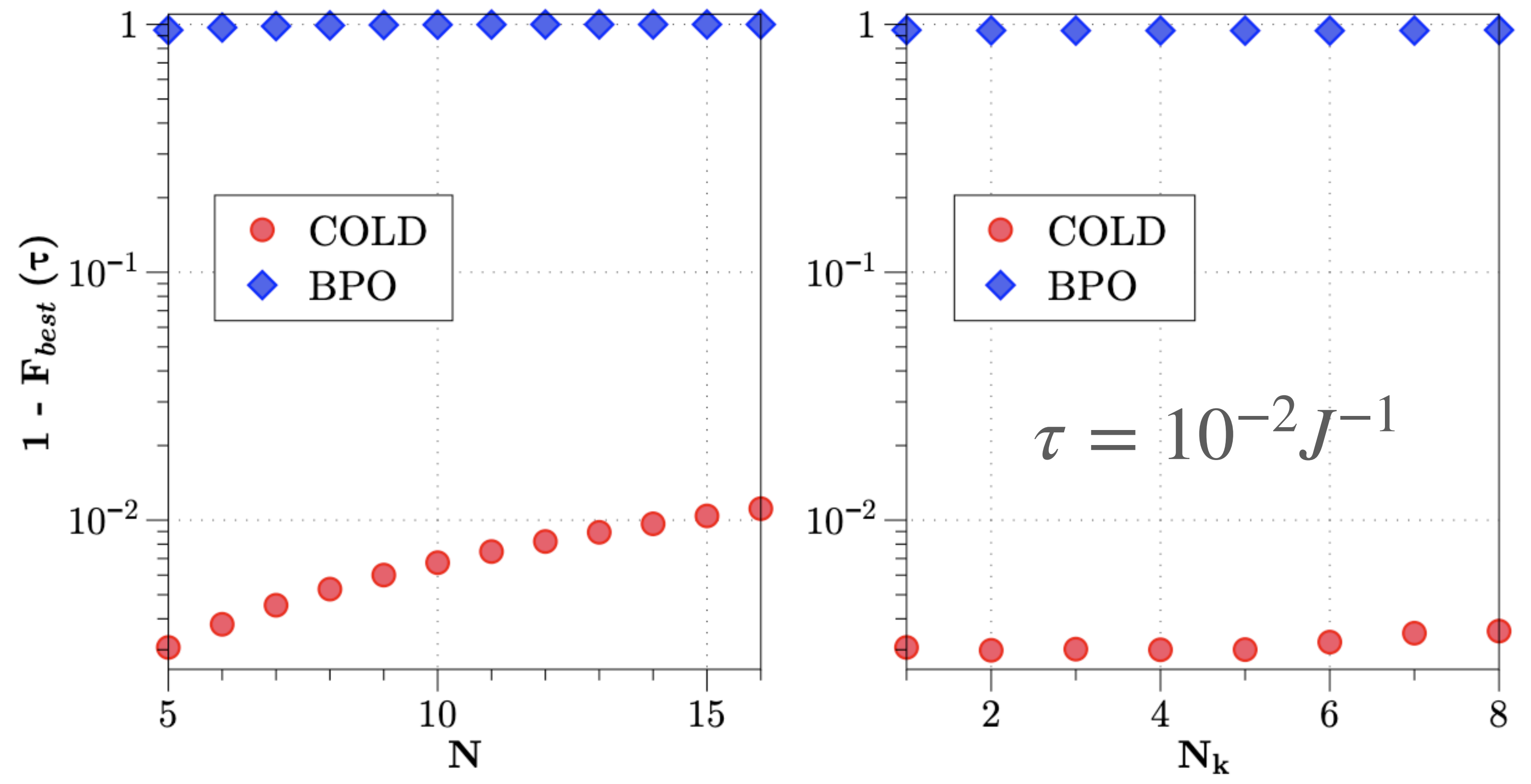
Result: Ising chain phase transition

Unconstrained

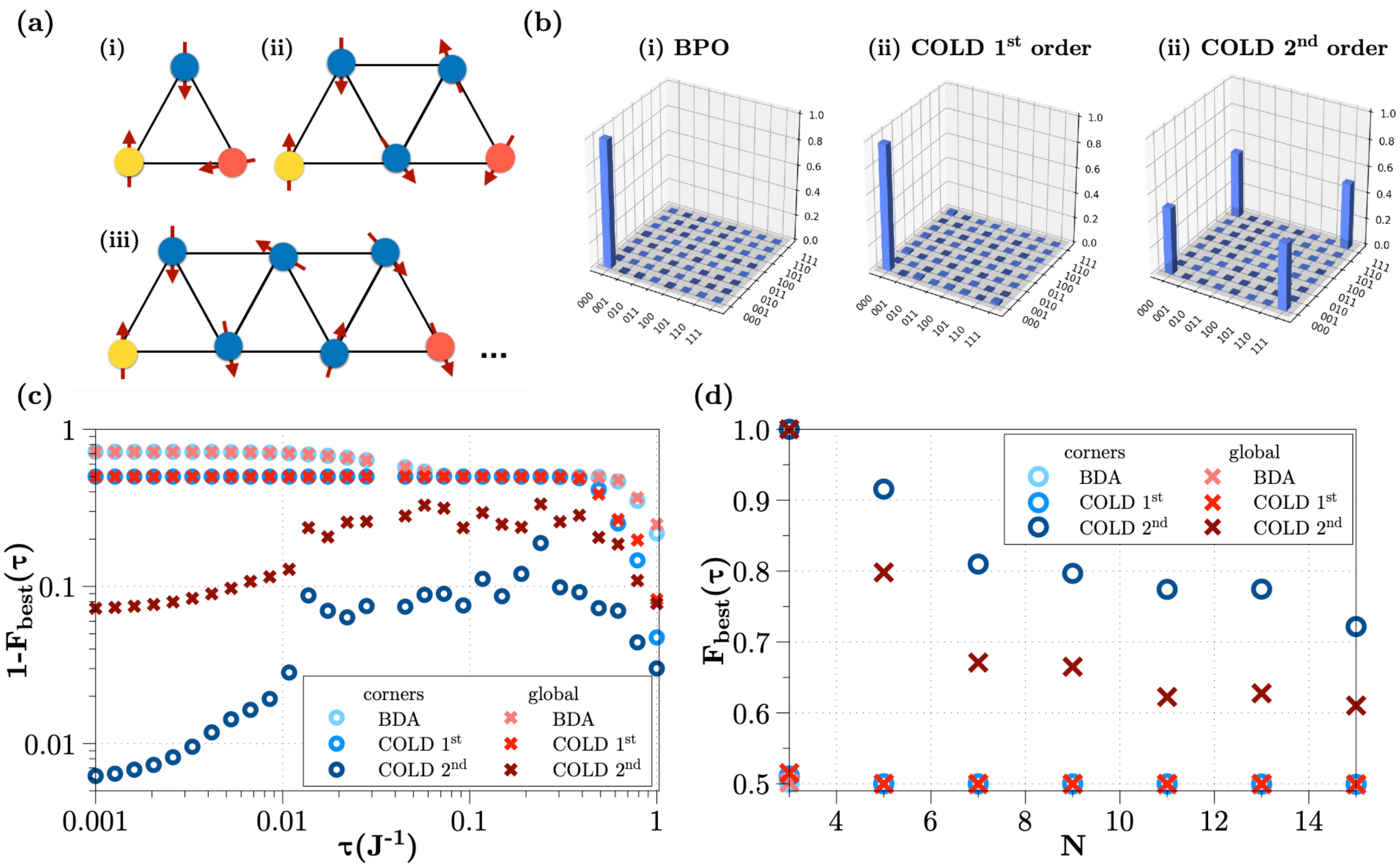


Constrained

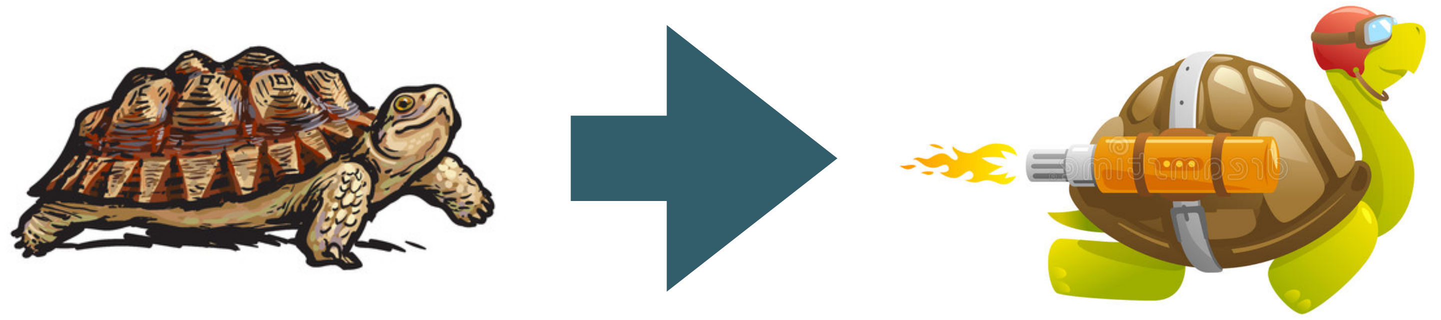




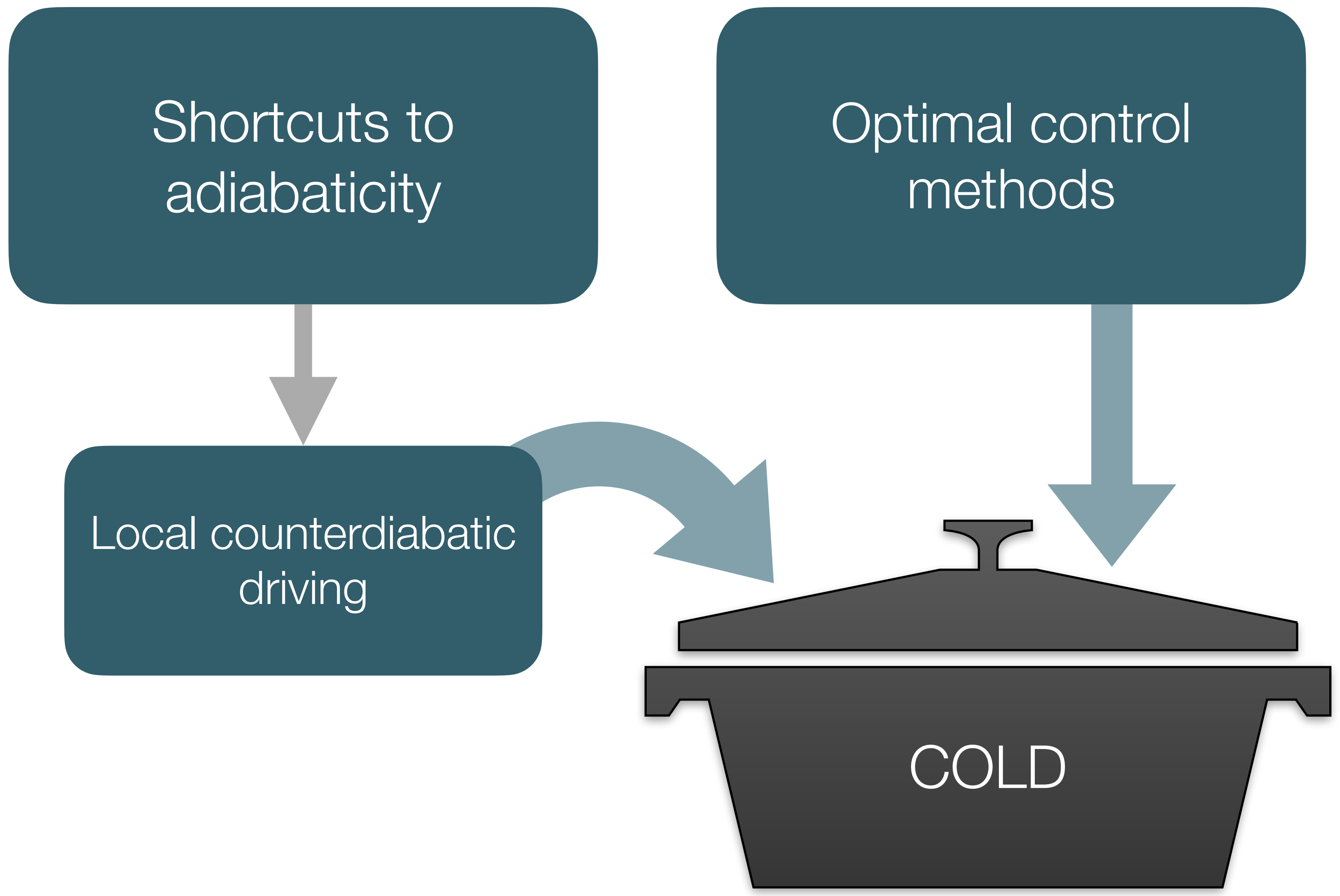
Result: GHZ state preparation in a system of frustrated spins

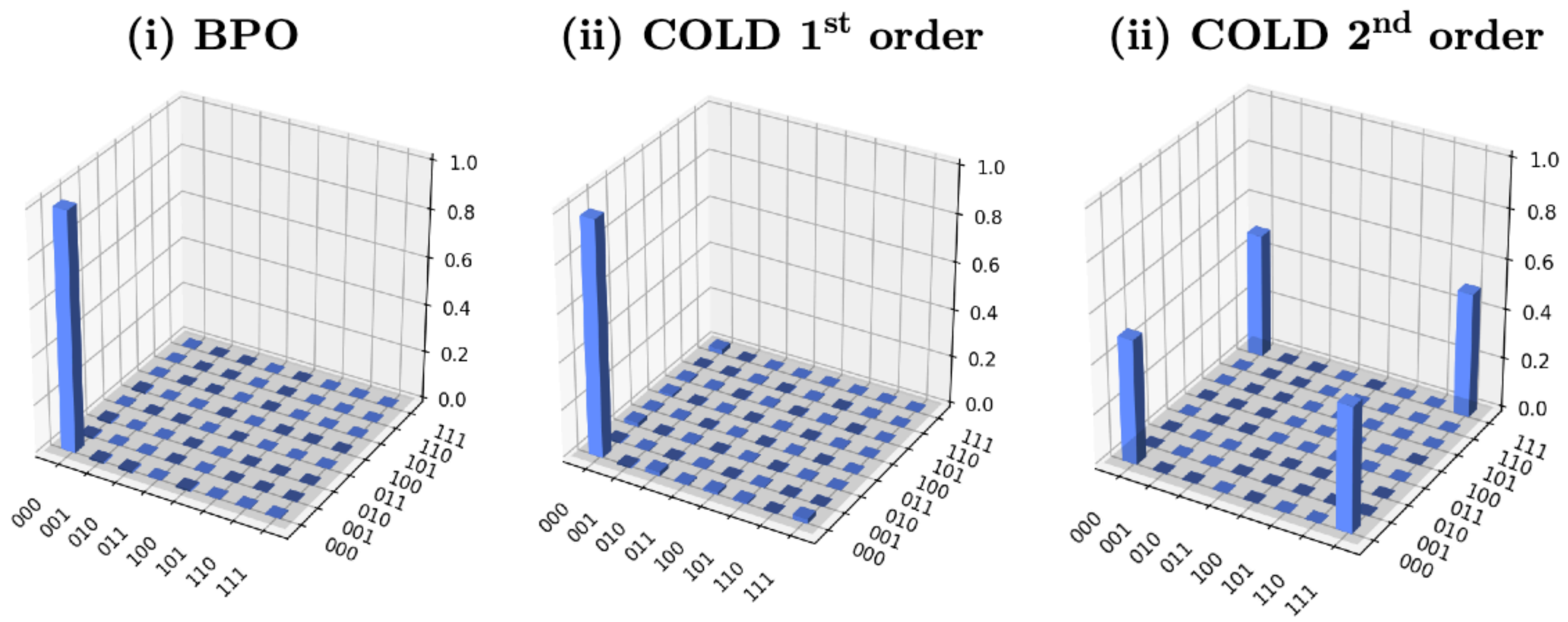


Summary

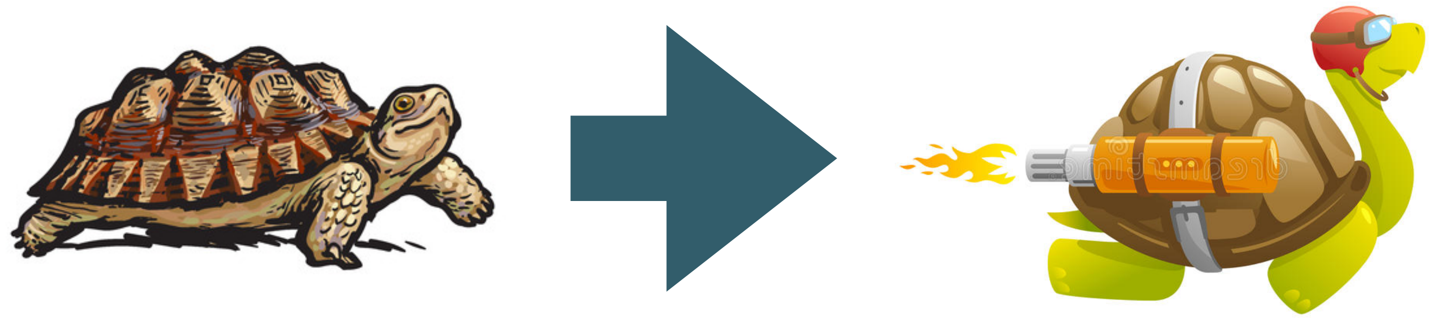


- **COLD**: new method for speeding up adiabatic processes
- Improves upon LCD and optimal control methods
- Outlook
    - Better understanding of AGP to inform/improve optimisation
    - Upcoming paper on new numerical/analytic methods for AGP on random Ising graphs!
      (See myself/**Ewen Lawrence: Y02:11 & N00:230**)





Links



Blogpost

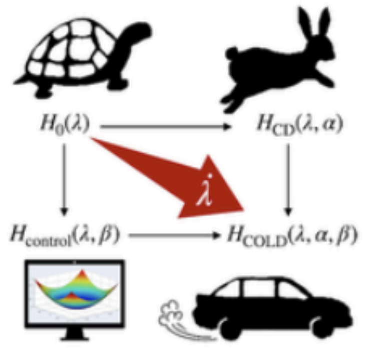
COLD: Counterdiabatic Optimised Local Driving

ievutec · January 31, 2023 · Own work
counterdiabatic driving, Own work, quantum computing · Edit

Paper

Counterdiabatic Optimized Local Driving

Ieva Čepaitė, Anatoli Polkovnikov, Andrew J. Daley, and Callum W. Duncan
PRX Quantum **4**, 010312 (2023) – Published 30 January 2023



A hybrid approach to achieving coherent control, dubbed COLD, promises substantial improvement when applied to annealing protocols, state preparation schemes, entanglement generation, and population transfer on a lattice.

Show Abstract +

@Hyperboleva @hyperboleva@mathstodon.xyz