Efficient Combinatorial Optimization via Heat Diffusion

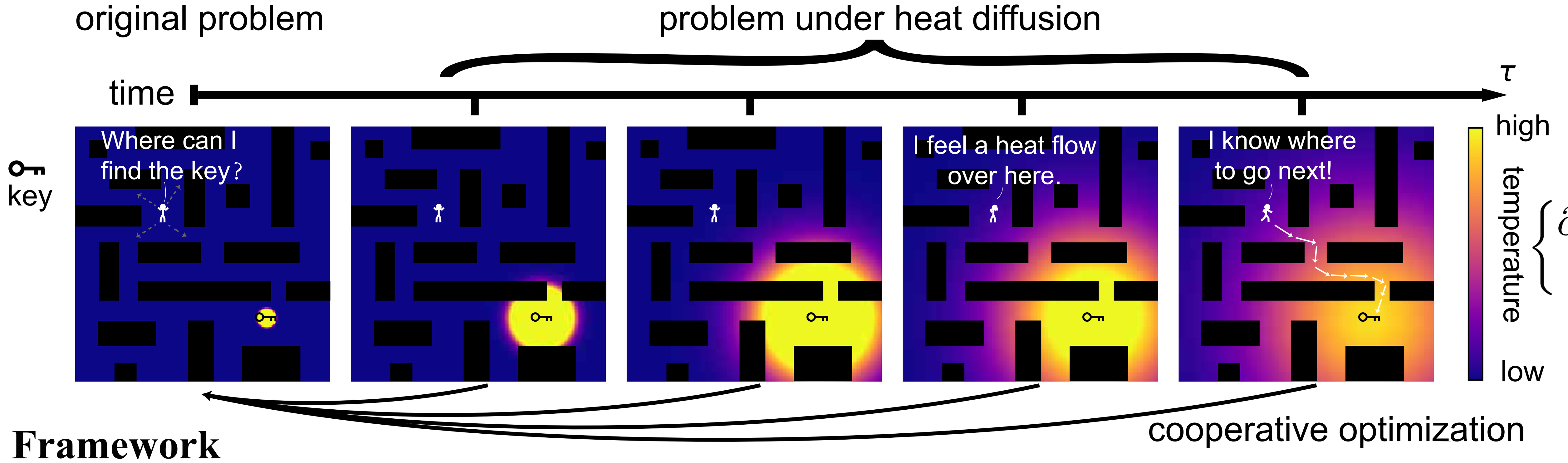
Hengyuan Ma, Wenlian Lu, and Jianfeng Feng

Background

Combinatorial optimization problems are widespread but inherently challenging due to their discrete nature. The primary limitation of existing methods is that they can only access a small fraction of the solution space at each iteration, resulting in limited efficiency .

Insight

Instead of expanding the solver's receptive field to acquire more information from the solution space, we concentrate on propagating information from distant areas of the solution space to the solver via heat diffusion.



heat equation

$$\begin{cases} \partial_\tau u(\tau,\boldsymbol{\theta}) = \Delta_{\boldsymbol{\theta}} u(\tau,\boldsymbol{\theta}), & \tau>0,\ \boldsymbol{\theta}\in\mathbb{R}^n \\ u(\tau,\boldsymbol{\theta}) = h(\boldsymbol{\theta}), & \tau=0,\ \boldsymbol{\theta}\in\mathbb{R}^n \end{cases}$$

Framework

$$\min_{\mathbf{s}\in\{-1,1\}^n} f(\mathbf{s}) \blacktriangleright \min_{\boldsymbol{\theta}\in\mathcal{I}} h(\boldsymbol{\theta}) \begin{cases} p(s_i=\pm1|\boldsymbol{\theta}) = 0.5 \pm (\theta_i - 0.5) \\ h(\boldsymbol{\theta}) = \mathbb{E}_{p(\mathbf{s}|\boldsymbol{\theta})}[f(\mathbf{s})] \\ \mathcal{I} := [0,1]^n \end{cases} \blacktriangleright \boldsymbol{\theta}_{t+1} = \boldsymbol{\theta}_t - \gamma \nabla_{\boldsymbol{\theta}} h(\boldsymbol{\theta}_t) \blacktriangleright \boldsymbol{\theta}_{t+1} = \mathrm{Proj}_{\mathcal{I}}(\boldsymbol{\theta}_t - \gamma \nabla_{\boldsymbol{\theta}} u(\tau_t,\boldsymbol{\theta}_t))$$

probabilistic encoding — allowing gradient descent — improved by heat diffusion

Transforming the target function by solving a heat equation in which the original target function is the initial state, and the states at different time points are considered as the target functions under different transform.

Theoretical results

• Consistency (invariance of global minimal under different heat diffusion)

Theorem 1. *For any $\tau>0$, the function $u(\tau,\boldsymbol{\theta})$ and $h(\boldsymbol{\theta})$ has the same global minima in $\bar{\mathbb{R}}^n$*

$$\arg\min_{\boldsymbol{\theta}\in\bar{\mathbb{R}}^n} u(\tau,\boldsymbol{\theta}) = \arg\min_{\boldsymbol{\theta}\in\bar{\mathbb{R}}^n} h(\boldsymbol{\theta})$$

• Efficiency (Analytic solution to the heat equation)

Theorem 2. *Supposed that $f(\mathbf{s})$ is a multilinear polynomial of $\mathbf{s}$, then the solution to the heat equation is*

$$u(\tau,\boldsymbol{\theta}) = \mathbb{E}_{p(\mathbf{x})}[f(\mathrm{erf}(\frac{\boldsymbol{\theta}-\mathbf{x}}{\sqrt{\tau}}))], \quad \mathbf{x}\in \mathrm{Unif}[0,1]^n,$$

where $\mathrm{erf}(x) = \frac{2}{\sqrt{\pi}}\int_0^x e^{-t^2}dt$ is the error function.

• Cooperative optimization (Error control from transformed optimization loss)

Theorem 3. *Denote $\check{f} = \max_{\mathbf{s}} f(\mathbf{s})$. Given $\tau_2>0$ and $\epsilon>0$, there exists $\tau_1\in(0,\tau_2)$, such that*

$$h(\boldsymbol{\theta}) - h(\boldsymbol{\theta}^*) \le \Big[(\check{f}-f^*)\big(u(\tau_2,\boldsymbol{\theta}^*) - u(\tau_2,\boldsymbol{\theta})\big) + \frac{n}{2}\int_{\tau_1}^{\tau_2} \frac{u(\tau,\boldsymbol{\theta}) - u(\tau,\boldsymbol{\theta}^*)}{\tau} d\tau\Big)\Big]^{1/2} + \epsilon.$$

Proposed algorithm: HeO

Algorithm 1 Heat diffusion optimization (HeO)

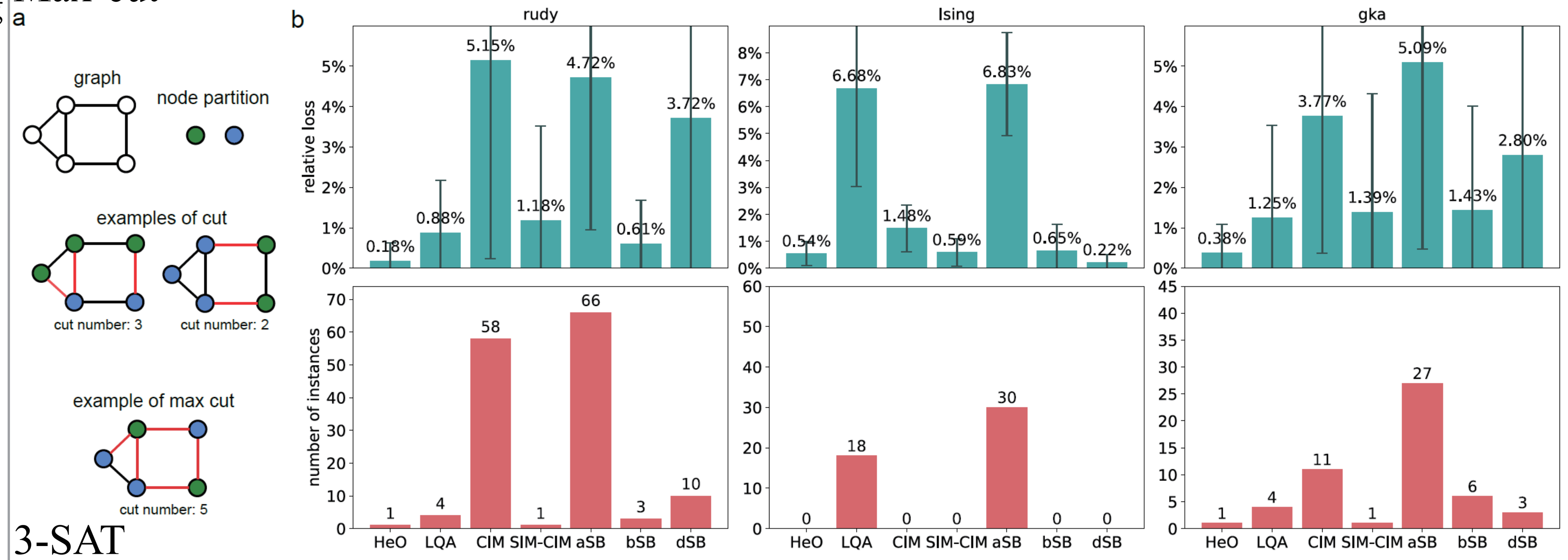
Input: target function $f(\cdot)$, step size γ, σ schedule $\{\sigma_t\}$, iteration number T
initialize elements of $\boldsymbol{\theta}_0$ as 0.5
for $t=0$ **to** $T-1$ **do**
  sample $\mathbf{x}_t \sim \mathrm{Unif}[0,1]^n$
  $\mathbf{g}_t \leftarrow \nabla_{\boldsymbol{\theta}} f(\mathrm{erf}(\frac{\boldsymbol{\theta}_t-\mathbf{x}_t}{\sigma_t}))$
  $\boldsymbol{\theta}_{t+1} \leftarrow \mathrm{Proj}_{\mathcal{I}}(\boldsymbol{\theta}_t - \gamma\mathbf{g}_t)$
end for
Output: binary configuration $\mathbf{s}_T = \mathrm{sgn}(\boldsymbol{\theta}_T - 0.5)$

Algorithm 3 Heat diffusion optimization (HeO) with momentum

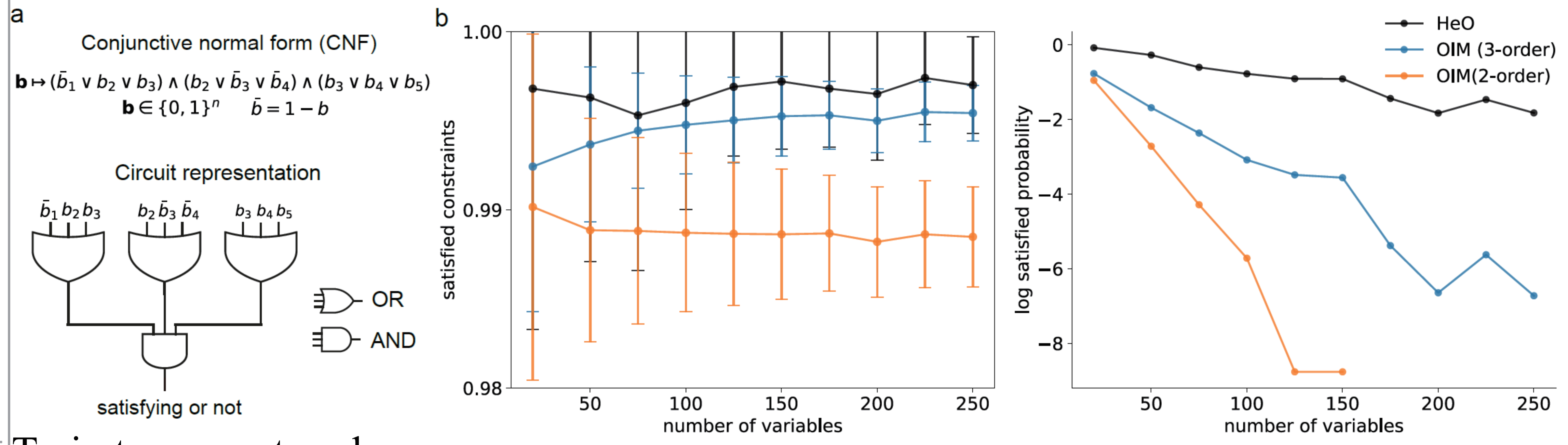
Input: target function $f(\cdot)$, step size γ, momentum κ, σ schedule $\{\sigma_t\}$, iteration number T, set $\mathbf{g}_{-1} = 0$
initialize elements of $\boldsymbol{\theta}_0$ as 0.5
for $t=0$ **to** $T-1$ **do**
  sample $\mathbf{x}_t \sim \mathrm{Unif}[0,1]^n$
  $\mathbf{w}_t \leftarrow \nabla_{\boldsymbol{\theta}_t} f(\mathrm{erf}(\frac{\boldsymbol{\theta}_t-\mathbf{x}_t}{\sigma_t}))$
  $\mathbf{g}_t \leftarrow \kappa\mathbf{g}_{t-1} + \gamma\mathbf{w}_t$
  $\boldsymbol{\theta}_{t+1} \leftarrow \mathrm{Proj}_{\mathcal{I}}(\boldsymbol{\theta}_t + \mathbf{g}_t)$
end for
$\mathbf{s}_T \leftarrow \mathrm{sgn}(\boldsymbol{\theta}_T - 0.5)$
Output: binary configuration $\mathbf{s}_T$

Experimental results (see our paper for more results)

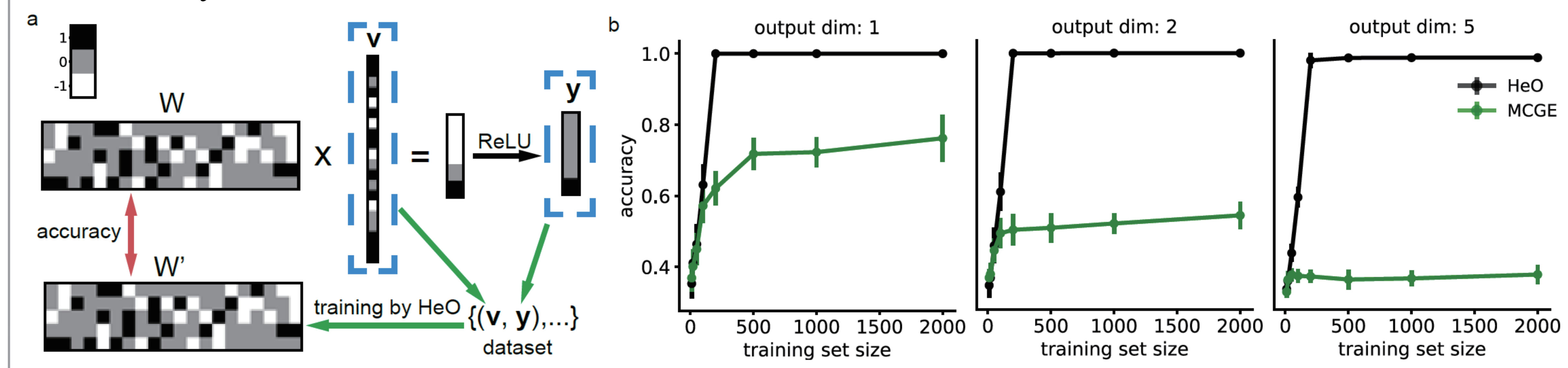
Max-cut



3-SAT



Train tenary networks



Variable selection for linear regression