Blocked Gibbs meets Diffusion Transformers: Unsupervised Learning for Constraint Optimization

Yudong Will Xu, Wenhao Li, Xiaoyu Wang, Scott Sanner, Elias B. Khalil

UNIVERSITY OF TORONTO

NEURAL INFORMATION PROCESSING SYSTEMS

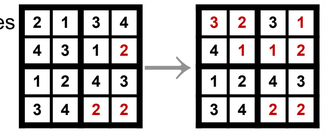
Key Takeaways

- *Method:* Blocked Gibbs diffusion enables **iterative block resampling**, from large early edits to small targeted refinements.
- *Results:* **Nearly doubles** hard Sudoku accuracy (48.6% → 94.1%) and cuts large-scale MaxCut gap by 80% vs. DiT.

Motivation

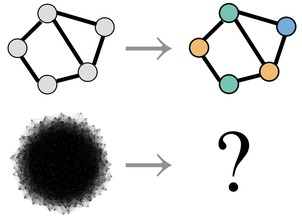
Why Blocked Gibbs?

- Standard Diffusion: Denoises all variables, potentially disrupting correct solutions.
- BloGDiT: Resamples selected blocks while preserving the rest.



Why Unsupervised?

- Supervised Learning: labels are often *difficult to obtain* and sometimes *ambiguous*.
- BloGDiT: Learns from problem energies without solution labels.



Background

Constraint Optimization

$$\min_{X \in \mathcal{X}} \; f(X) \quad \text{s.t.} \quad c_j(X) = \text{true} \quad \forall j$$

Lagrangian Relaxation

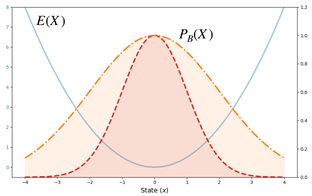
$$\min_{X \in \mathcal{X}} \; \underbrace{f(X) + \sum_j \lambda_j \phi_j(X)}_{E(X)},$$

"Energy Function"

$$\phi_j(X) = 0 \iff c_j(X) = \text{True}$$

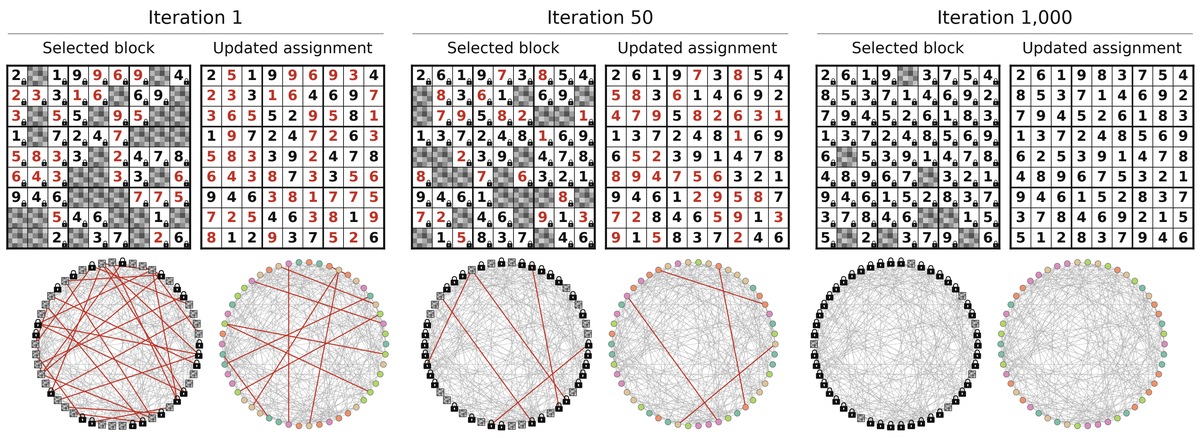
Distribution of Solutions

$$p_B(X) = \frac{\exp\left(-E(X)/\tau\right)}{\sum_{X' \in \mathcal{X}} \exp\left(-E(X')/\tau\right)}$$



Sampling from this distribution provides solutions to the constraint problem

Experiments



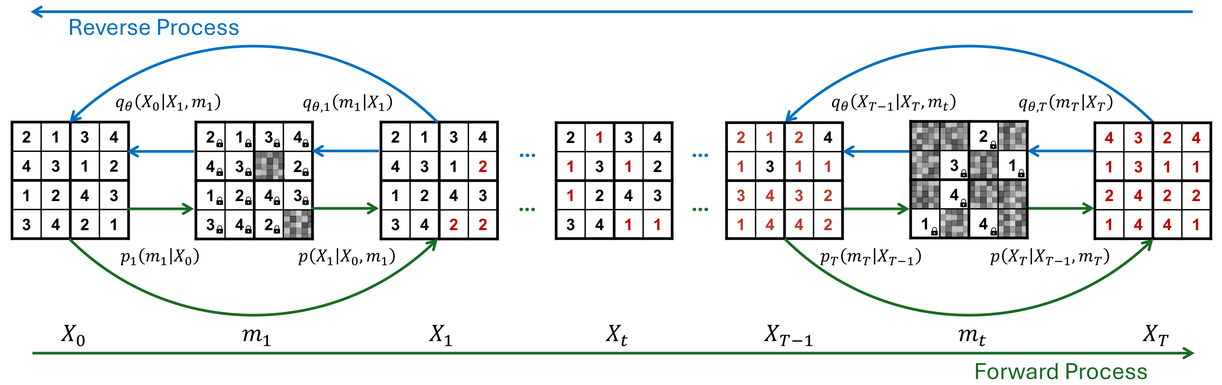
Sudoku	Easy ↑	Hard ↑
OR-Tools	**100**	**100**
SATNet	98.3	3.2
RRN	99.8	28.6
Yang et al. (2023)	**100**	32.9
IRED	99.4	62.1
RLSA	4.2	1.8
ConsFormer	**100**	77.7
DiT	**100**	48.6
BloGDiT	**100**	94.1

Graph Coloring	5 Colors		10 Colors	
	$n = 50$ ↑	$n = 100$ ↑	$n = 100$ ↑	$n = 200$ ↑
OR-Tools	**83.08**	**57.16**	52.41	10.25
Greedy	32.42	0.00	0.75	0.00
Feasibility Jump	82.83	54.50	35.66	6.00
ANYCSP	79.17	34.83	0.00	0.00
RLSA	37.83	0.42	10.75	2.50
ConsFormer	81.00	47.33	52.60	11.92
DiT	79.75	48.91	50.50	14.25
BloGDiT	83.00	54.67	**53.92**	**18.58**

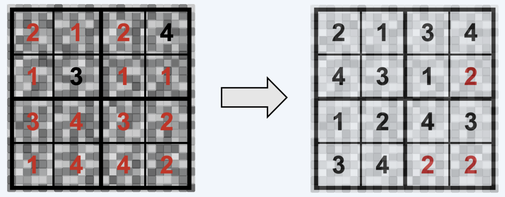
MIS	RB-small ↑	RB-large ↑
OR-Tools	**20.10**	**42.66**
LwD	19.01	32.32
INTEL	18.47	34.47
DGL	17.36	34.50
LTFT	19.18	37.48
DIFUSCO	18.52	–
RLSA	19.97	40.19
X^2GNN	20.07	39.28
DiffUCO	18.88	38.10
DiT	8.05	0.29
BloGDiT	19.52	37.05

MaxCut	$\lvert V\rvert = 800$ ↓	$\lvert V\rvert = 1$K ↓	$\lvert V\rvert = 2$K ↓	$\lvert V\rvert \geq 3$K ↓
OR-Tools	143.89	112.78	365.89	378.62
RUNCSP	185.89	156.56	357.33	401.00
ECO-DQN	65.11	54.67	157.00	428.25
ECORD	8.67	8.78	39.22	187.75
ANYCSP	1.22	2.44	13.11	51.63
RLSA	3.22	4.00	13.0	66.75
X^2GNN	21.11	6.00	22.44	46.38
DiffUCO	4.11	6.33	31.67	116.75
ConsFormer	16.33	12.44	52.11	115.25
DiT	56.11	50.00	101.33	162.00
BloGDiT	**0.11**	**1.33**	**7.88**	**32.75**

Blocked Gibbs Diffusion Transformer (BloGDiT)



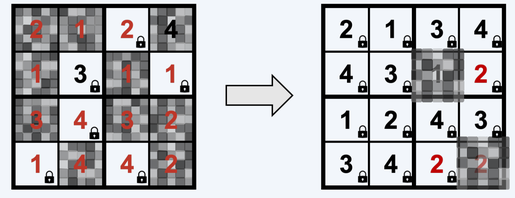
Standard Diffusion



Early: Large Magnitude Noise to ALL Variables

Late: Small Magnitude Noise to ALL Variables

Blocked Gibbs Diffusion



Early: Constant Noise to LARGE Block

Late: Constant Noise to SMALL Block

Forward Process

$$p(X_{0:T}) = p_B(X_0) \prod_{t=1}^{T} \underbrace{p(X_t \mid X_{t-1})}_{\text{noise all}},$$

$$p(X_{0:T}, m_{1:T}) = p_B(X_0) \prod_{t=1}^{T} \underbrace{p_t(m_t \mid X_{t-1})}_{\text{select block}} \; \underbrace{p(X_t \mid X_{t-1}, m_t)}_{\text{noise block}}$$

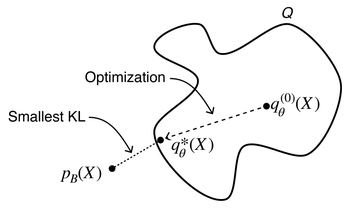
Reverse Process

$$q_\theta(X_{0:T}) = q(X_T) \prod_{t=1}^{T} \underbrace{q_\theta(X_{t-1} \mid X_t)}_{\text{denoise all}}$$

$$q_\theta(X_{0:T}, m_{1:T}) = q(X_T) \prod_{t=1}^{T} \underbrace{q_{\theta,t}(m_t \mid X_t)}_{\text{select block}} \; \underbrace{q_\theta(X_{t-1} \mid X_t, m_t)}_{\text{denoise block}}$$

Auxiliary masks enable blocked resampling and can be marginalized out exactly, preserving the target distribution.

Unsupervised Training Objective



Forward KL needs data samples:

$$\mathrm{KL}(p_B(X_0) \| q_\theta(X_0)) = \mathbb{E}_{X_0 \sim p_B}\left[\log \frac{p_B(X_0)}{q_\theta(X_0)}\right]$$

Reverse KL uses model samples:

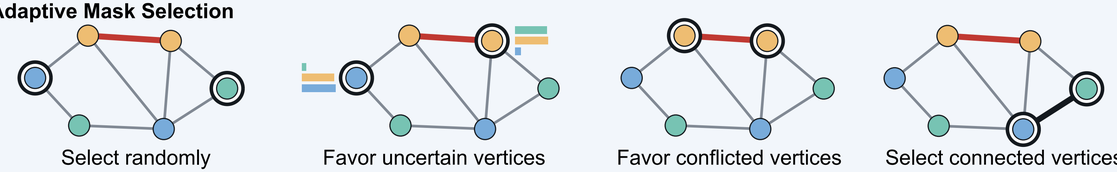
$$\mathrm{KL}(q_\theta(X_0) \| p_B(X_0)) = \mathbb{E}_{X_0 \sim q_\theta}\left[\log \frac{q_\theta(X_0)}{p_B(X_0)}\right]$$

Minimize the upper bound: $\mathrm{KL}(q_\theta(X_0) \| p_B(X_0)) \leq \mathrm{KL}(q_\theta(X_{0:T}, m_{1:T}) \| p(X_{0:T}, m_{1:T}))$

Simplifies to:

$$\underbrace{\mathbb{E}_q E(X_0)}_{\text{Sample Energy}} \; \underbrace{- \tau \sum_t \mathbb{E}_q h\left(q_\theta(\cdot \mid X_t, m_t)\right)}_{\text{Entropy}} \; \underbrace{- \tau \sum_t \mathbb{E}_q \log p(X_t \mid X_{t-1}, m_t)}_{\text{forward noise matching}} \; \underbrace{+ \tau \sum_t \mathbb{E}_q \log \frac{q_{\theta,t}(m_t \mid X_t)}{p_t(m_t \mid X_{t-1})}}_{\text{mask matching}}$$

Adaptive Mask Selection



Select randomly | Favor uncertain vertices | Favor conflicted vertices | Select connected vertices