RESEARCH-ARTICLE

Architectural CAD/CAM in Robotic Fabrication via Differentiable Energy Optimization and Few-Shot Spatiotemporal Graph RL Policy

SHIHYU LAI, National Taiwan University, Taipei, Taiwan

CHIACHING YEN, National Cheng Kung University, Tainan, Taiwan

YANGTING SHEN, National Cheng Kung University, Tainan, Taiwan

BINGYU CHEN, National Taiwan University, Taipei, Taiwan

Open Access Support provided by:

National Cheng Kung University

National Taiwan University

Published: 15 December 2025

SA Technical Communications '25: SIGGRAPH Asia 2025 Technical Communications
December 15 - 18, 2025
Hong Kong, Hong Kong

Conference Sponsors:
SIGGRAPH

SA Technical Communications '25: Proceedings of the SIGGRAPH Asia 2025 Technical Communications (December 2025)
https://doi.org/10.1145/3757376.3771390
ISBN: 9798400721366

Architectural CAD/CAM in Robotic Fabrication via Differentiable Energy Optimization and Few-Shot Spatiotemporal Graph RL Policy

Shih-Yu Lai
National Taiwan University
Taipei City, Taiwan
akinesia112@gmail.com

Chia-Ching Yen
National Cheng Kung University
Tainan, Taiwan
ccyen@gs.ncku.edu.tw

Yang-Ting Shen
National Cheng Kung University
Tainan, Taiwan
bowbowshen@gmail.com

Bing-Yu Chen
National Taiwan University
Taipei, Taiwan
robin@ntu.edu.tw

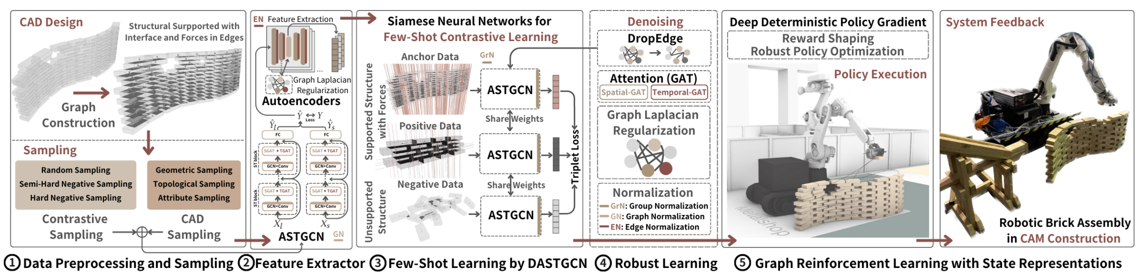

Figure 1: The system architecture. We integrate CAD/CAM with robotic brick assembly using graph-based few-shot contrastive reinforcement learning with DASTGCN, SNN, and DDPG. This approach improves decision-making and strategies for sparse rewards, enhancing the automated adaptation and robust assembly capabilities of the manipulator for varied geometric designs.

Abstract

With the integration of robotic manipulation into digital fabrication, the convergence of Computer-Aided Design and Manufacturing (CAD/CAM) faces new challenges. The highly customized nature of architectural design results in diverse brick shapes and often sparse data within categories, posing significant impediments in robotic assembly tasks to learn efficiently from limited samples about new design types. Challenges include structure support judgment and noise interference in simulating assembly processes with complex Spatial-Temporal Graph (STG) relationships, as well as lengthy CAD iteration cycles, and CAM structural testing practice leading to sparse rewards. Our innovative robotic assembly of geometric brick utilizes graph-based few-shot reinforcement learning with Denoising Attention STG Convolutional Networks, Siamese Neural Networks, and Deep Deterministic Policy Gradient with Differentiable Energy Minimization to enhance operational efficiency and robustness. Our method outperforms in the aspects of STG, few-shot learning, and reinforcement learning baselines, and the effectiveness of modules is demonstrated in an ablation study. This system enables designers to translate complex designs into automation for diverse assembly patterns, ensuring the manipulator's precise placement, stability, and aesthetics, even in highly adaptable conditions, empowering efficient collaboration of the users, including the designer, engineer, manufacturer, etc.

CCS Concepts

• **Applied computing → Computer-aided design**.

ACM Reference Format:
Shih-Yu Lai, Chia-Ching Yen, Yang-Ting Shen, and Bing-Yu Chen. 2025. Architectural CAD/CAM in Robotic Fabrication via Differentiable Energy Optimization and Few-Shot Spatiotemporal Graph RL Policy. In *SIGGRAPH Asia 2025 Technical Communications (SA Technical Communications '25), December 15–18, 2025, Hong Kong, Hong Kong*. ACM, New York, NY, USA, 4 pages. https://doi.org/10.1145/3757376.3771390

1 Introduction

In the rapidly evolving construction industry, the challenge of integrating robotic technologies with advanced computer-aided design (CAD) to facilitate and optimize physical building tasks has become increasingly critical [Stumm et al. 2016]. This paper introduces a cutting-edge approach that harnesses the convergence of CAD

SA Technical Communications '25, Hong Kong, Hong Kong

ACM ISBN 979-8-4007-2136-6/25/12
https://doi.org/10.1145/3757376.3771390

and computer-aided manufacturing (CAM) with robotic systems to enhance the assembly of geometric bricks in diverse and intricate architectural designs. By leveraging graph-based data structures and innovative learning methodologies, our research aims to streamline and refine the assembly processes, ensuring high precision and operational efficiency in automated construction. Current methodologies in robotic assembly, have struggled to transition from controlled laboratory environments to actual construction sites [Atad et al. 2023]. This is primarily due to the limitations inherent in traditional learning algorithms (e.g., pre-training, fine-tuning, and transfer learning), which fail to effectively handle the complexity and variability of architectural designs found in real-world scenarios. To address these challenges, our approach (**Fig. 1**) employs a novel spatiotemporal graph-based few-shot learning technique termed Denoising Attention Based Spatial-Temporal Graph Convolutional Networks (DASTGCN)[Guo et al. 2019], which adapts quickly to new design types by leveraging minimal data samples. The method utilizes the inherent temporal and spatial relationships within construction sequences to adjust to new designs rapidly. Moreover, we introduce a robust learning framework that incorporates denoising mechanisms within our Siamese Neural Networks (SNN) [Chicco 2021], and apply the features to Deep Deterministic Policy Gradient (DDPG) [Lillicrap et al. 2019] models with Differentiable Energy Minimization (DEM). These features significantly enhance the learning efficiency and decision-making quality of our system under conditions characterized by sparse data and high variability in design structures. By employing innovative sampling strategies like contrastive and CAD sampling, our method effectively handles the data, which is common in highly customized design and construction tasks. Our approach focuses on:

- **Enhanced Data Processing and Sampling:** By implementing advanced sampling methods, our system efficiently preprocess complex CAD data, improving the quality and usability of data for learning applications, reducing computing costs and data requirements.
- **Feature Extraction and Robust Spatiotemporal Graph Few-Shot Learning:** Utilizing DASTGCN, our system extracts critical features from limited data in the learning process, allowing for rapid adaptation to new and complex designs in CAD/CAM construction.
- **Reinforcement Learning with State Representations and Differentiable Energy Minimization:** This technique optimizes long-term operational strategies, enhancing the robot's ability to perform in diverse construction scenarios while considering physical constraints of mobile manipulator by Differentiable Energy Minimization (DEM), and maintaining accuracy and robustness in assembly capabilities.

2 Method

2.1 Data Preprocessing and Sampling

2.1.1 Dataset. In the pursuit of integrating CAD/CAM for robotic assembly with geometric bricks, we address the problem of assembly using spatiotemporal graph-based models for complex architectural structures. After architects model the brick designs and converting them into mesh, we employ a graph representation for each brick modeled as a node with attributes encapsulating its position, shape, vertex coordinates, and interface forces as edges, formulated as $\mathcal{G} = (\mathcal{V}, \mathcal{E}, \mathbf{X}, \mathbf{F})$ into a dataset $\mathcal{D}$, where $\mathcal{V}$ represents vertices (bricks), $\mathcal{E}$ denotes edges (structural support interactions), $\mathbf{X} \in \mathbb{R}^{n \times d}$ captures the positional and shape features of bricks, and $\mathbf{F} \in \mathbb{R}^{m \times k}$ describes forces and interface attributes across edges.

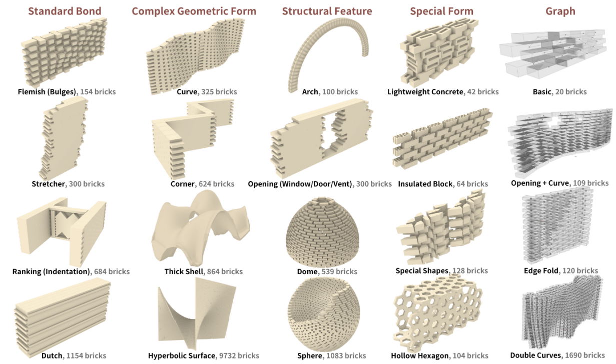

Figure 2: Samples of customized CAD for brick assembly in diverse types and graphs with interfaces.

2.1.2 Task. Regular assessments in both controlled environments and actual construction sites test models against benchmarks of geometric brick types such as Flemish and Stretcher Bond designs. These designs often include intricate architectural features like curved or spherical openings for doors and windows, even in hollow hexagonal brick (**Fig. 2**). Our primary goal is to assess the precision with which the robotic manipulator positions each brick, ensuring millimeter-level accuracy in brick placements and structural stability, while achieving the default and customized design and fabrication with various requirements.

2.1.3 Contrastive and CAD Sampling. To select data samples from whole structured bricks planning to emphasize differences between positive and negative pairs within a learning model by **contrastive sampling**, it is crucial for training discriminative models. This process (**Fig. 1**) involves various strategies like (1) **Random Sampling:** $\{x_i\}_{i=1}^{n} \overset{\text{iid}}{\sim} \mathcal{D}$, where samples x_i are randomly chosen from a dataset $\mathcal{D}$; (2)**Semi-Hard Negative Sampling**: $\{(x_i, x_j) \mid d(x_i, x_j) < \theta, x_i \in \mathcal{X}_{\text{pos}}, x_j \in \mathcal{X}_{\text{neg}}\}$, which selects pairs (x_i, x_j) such that the distance metric $d(x_i, x_j)$ is less than a threshold θ; and (3) **Hard Negative Sampling**: $\{(x_i, x_j) \mid d(x_i, x_j) < \epsilon, x_i \in \mathcal{X}_{\text{pos}}, x_j \in \mathcal{X}_{\text{neg}}\}$, where pairs are chosen similarly but the distance must be less than a smaller threshold ϵ. Here, x_i and x_j belong to sets of positive $\mathcal{X}_{\text{pos}}$ and negative samples $\mathcal{X}_{\text{neg}}$, respectively. **CAD sampling** focuses on extracting representative samples from the diverse types of customized CAD model C (**Fig. 2**), including (1) **Geometric Sampling**: x is sampled based on geometric features of C, (2) **Topological Sampling**: x is sampled from topological relations within C, and (3) **Attribute Sampling**: x is sampled from the attribute characteristics like structural and forces properties of C. Ultimately, we concatenate the representative samples from contrastive and CAD sampling. This strategy aims to provide diverse

Table 1: Model Performance Comparison

(a) Spatiotemporal Graph Learning

Model	MSE	MAE	TCC
STGCN	0.040	0.15	0.82
GAT	0.035	0.14	0.85
PTDNet	0.030	0.15	0.81
ASTGCN	0.031	0.12	0.89
GraphSAGE-GAT	0.026	0.15	0.90
Ours	**0.025**	**0.10**	**0.92**

(b) Few-Shot Learning

Model	Accuracy	F1 Score	NMI	Triplet Loss
SNN	0.75	0.70	0.60	0.25
LEO	0.79	0.74	0.67	0.20
ProtoNet	0.80	0.75	0.70	0.24
Ours	**0.85**	**0.80**	**0.75**	**0.18**

(c) Graph Reinforcement Learning

Model	R	$\bar{r}$	c
DDPG-GNN	160	1.6	0.25
DDPG-GAT	162	1.62	0.23
DDPG-PTDNet	159	1.59	0.22
MA-GCN	157	1.57	0.28
GNN-UCB	155	1.65	0.23
Ours	**170**	**1.7**	**0.20**

and challenging datasets that encapsulate the structural complexities and design variabilities inherent in robotic brick assembly tasks, thereby enhancing the model's learning efficiency and robustness.

2.2 Feature Extractor and Few-Shot Learning

2.2.1 ASTGCN. After the sampling process, the feature extraction is conducted by the ASTGCN and autoencoders (**Fig. 1**). The spatial-temporal graph is constructed in ASTGCN using the inverse kinematics of the manipulator and the inverse reachability map of the vehicle based on the bricks' central positions. This graph construction is crucial for accurately modeling the physical interactions required for the robotic assembly. This network processes two sets of inputs: long-term historical data X_l and short-term immediate data X_s. These inputs are transformed through several layers of Graph Convolution Networks combined with convolutional layers (GCN+Conv), followed by a combination of Spatial Graph Attention (SGAT) and Temporal Graph Attention (TGAT) mechanisms. The whole module is called the ST block. The outputs from ASTGCN represented as $\hat{Y}_l$ and $\hat{Y}_s$, correspond to the processed long-term and short-term data inputs, respectively. These outputs are compared against their respective targets to minimize the discrepancy between Y_l and Y_s, utilizing RMSE as the loss function.

2.2.2 Autoencoders with Graph Laplacian Regularization(GLR). The features processed are further refined through graph autoencoder structures that incorporate Graph Laplacian Regularization [Ando and Zhang 2006]. This regularization technique imposes a smoothness constraint on the feature representation by penalizing the Laplacian quadratic form: $L = D - A$, where L is the Laplacian matrix, D is the diagonal degree matrix, and A is the adjacency matrix of the graph. Regularization helps retain essential topological characteristics, promotes generalization, and reduces overfitting.

2.2.3 Siamese Neural Networks. Following the feature extraction, the Siamese Neural Networks (SNN) utilize these refined features for few-shot contrastive learning (**Fig. 1**). The SNN focuses on learning from minimal data samples by comparing pairs of similar and dissimilar instances, thus optimizing the model's ability to generalize from a few examples. This is essential for quickly adapting to new types of brick designs with limited available data. To adapt to the data sparsity for new brick designs, we implement metrics such as **Accuracy, F1 Score, Normalized Mutual Information (NMI), and Triplet Loss** among the Denoising-based ASTGCN (DASTGCN) models with shared weights within the SNN structure.

2.3 Denoising ASTGCN and Robust Learning

2.3.1 DropEdge and GAT in SNN. The integration of DropEdge [Rong et al. 2019] and Graph Attention Networks (GAT) [Veličković et al. 2018] with the DASTGCN plays a pivotal role in the SNN architecture (**Fig. 1**). DropEdge randomly removes a certain percentage of edges from the input graph before each training epoch. This randomness helps prevent overfitting and makes the model more generalizable, effectively acting as a data augmentation that introduces variability and reduces the likelihood of learning spurious correlations: $\hat{G} = G - \{e_{ij} \mid e_{ij} \in E, p < \tau\}$, where $\hat{G}$ represents the modified graph, G is the original graph, E is the set of edges, e_{ij} is an edge between nodes i and j, p is a randomly generated probability, and τ is the threshold probability for edge removal. Simultaneously, GAT is employed to dynamically focus on the most important nodes within the manipulated graph structure, thereby allowing for a more nuanced understanding and enhancement of the important features through learnable attention coefficients. The attention mechanism helps in prioritizing nodes based on their contribution to task performance, effectively learning a weighted representation of local neighborhoods: $h'_i = \sigma\left(\sum_{j\in\mathcal{N}(i)} \alpha_{ij} W h_j\right)$, where h'_i is the updated feature of node i, σ is a nonlinear activation function, $\mathcal{N}(i)$ denotes the neighbors of node i, α_{ij} are the attention coefficients learned by the network, W is a learnable weight matrix, and h_j are the features of the neighboring nodes. This approach represents a significant advancement over traditional methods, offering a robust solution to the challenges posed by complex architectural designs, enhancing the denoising capability.

2.4 GRL with State Representations

2.4.1 Deep Deterministic Policy Gradient. For the robotic assembly tasks, the system employs a mobile manipulator that operates within a continuous state space and a discrete action space. The DDPG approach is applied to optimize motion costs, facilitate dynamic adaptation to continuously varying design configurations, and ensure structural stability from the bottom up in the predetermined brick wall designs. The DDPG formulation is as follows: $\delta_t = r_t + \gamma Q(s_{t+1}, \mu(s_{t+1}|\phi^{\mu})) - Q(s_t, a_t|\phi^{Q}); \phi^{Q'} \leftarrow \tau\phi^{Q} + (1-\tau)\phi^{Q'}; \phi^{\mu'} \leftarrow \tau\phi^{\mu} + (1-\tau)\phi^{\mu'}$, where s_t and a_t denote the state and action at time t, r_t is the reward, Q represents the critic network, μ is the actor network, ϕ^{Q} and ϕ^{μ} are the network parameters, and τ is the learning rate.

2.4.2 State and Action Space Representation. The state space encapsulates the physical configuration of the assembly process, incorporating vectors for each brick's centroid coordinates, vertex

Table 2: Ablation Study: DASTGCN w/ DropEdge, GAT, & GLR

Configuration	Accuracy	F1 Score	NMI	Triplet Loss
Baseline (Ours w/o all denoising)	0.80	0.75	0.68	0.22
Ours w/o Graph Laplacian Regularization	0.81	0.76	0.72	0.20
Ours w/o DropEdge	0.82	0.76	0.71	0.21
Ours w/o GAT	<u>0.84</u>	<u>0.78</u>	<u>0.73</u>	<u>0.19</u>
Ours	**0.85**	**0.80**	**0.75**	**0.18**

coordinates, and orientation matrices capturing rotational degrees of freedom. The vectors detail mechanical forces like normal force, friction, tension, compression, shear force, and torque, which are crucial for ensuring structural stability. The action space is modeled as a combination of discrete choices from predefined positions and continuous adjustments in vectors, optimizing layout to minimize motion costs and enhance precision: $\mathbf{s}_t = [\mathbf{c}, \mathbf{v}, \mathbf{o}, \mathbf{f}]$, $\mathbf{a}_t = [\mathbf{p}, \Delta\mathbf{v}]$, where $\mathbf{c}$ represents the centroid coordinates, $\mathbf{v}$ the vertex coordinates, $\mathbf{o}$ the orientation matrices, $\mathbf{f}$ the mechanical forces, $\mathbf{p}$ predefined positions, and $\Delta\mathbf{v}$ continuous adjustments. This comprehensive approach ensures that the model captures necessary temporal and spatial relationships in the graph data while adhering to the underlying structural properties, significantly enhancing the overall predictive performance and robustness of the robotic assembly.

2.4.3 Reward Shaping and Robust Policy Optimization. We modify the reward function to guide the policy towards desired behaviors and handle the sparse reward: $r_t = r(s_t, a_t, s_{t+1}) + \lambda \cdot \text{Penalty}(s_t, a_t)$, where λ is a weighting factor that balances the immediate rewards and penalties to induce robust policy optimization. The execution of the policy and the system feedback are crucial for the continuous learning and adaptation of the robotic system: $a_t = \mu(s_t|\phi^{\mu}) + \epsilon, \quad \epsilon \sim \mathcal{N}(0, \sigma)$, where ϵ represents noise added to the policy action to explore the action space effectively, and σ denotes the standard deviation of the exploratory noise.

2.4.4 Differentiable Energy Minimization. The DEM is enhanced to incorporate complex robotic dynamics and environmental interactions, ensuring precise control and efficient energy usage in assembly. The energy function is extended to include terms that represent the kinematic and dynamic constraints: $E(\mathbf{x}, \dot{\mathbf{x}}, \ddot{\mathbf{x}}) = \sum_{i=1}^{n} k_i \cdot \exp\left(-\frac{||\mathbf{x}-\mathbf{x}_i||^2}{2\sigma^2}\right) + \lambda_1 \|\mathbf{J}(\mathbf{x})\dot{\mathbf{x}}\|^2 + \lambda_2 \|\ddot{\mathbf{x}}\|^2$,where $\mathbf{x}$ is the configuration vector of the mobile manipulator, $\dot{\mathbf{x}}$ and $\ddot{\mathbf{x}}$ are the velocity and acceleration vectors respectively, $\mathbf{J}(\mathbf{x})$ is the Jacobian matrix of the manipulator for inverse kinematics, and λ_1, λ_2 are weighting factors for velocity and acceleration constraints.

3 Results

3.1 Model Comparisons.

The tables (**Table 1**) provide a comparison between our method and baseline models using the specified metrics. Across all models and metrics examined, Our Method consistently outperforms the baselines, confirming its robustness and adaptability to different learning paradigms in robotic brick assembly. Its superior performance in handling complex geometric designs and learning from limited examples highlights its potential for practical implementation in automated construction environments. In **Table 1**, Our Method again shows superior performance, leading with the highest expected return ($R = 170$), average reward per episode ($\bar{r} = 1.7$), and the lowest convergence rate ($c = 0.20$). This indicates a more efficient learning and decision-making process in graph-based reinforcement learning tasks, optimizing both short-term gains and long-term strategies. DASTGCN enhances spatiotemporal feature learning via DropEdge, GAT, and Graph Laplacian Regularization in ST-GFSL, critical for structural fidelity in robotic brick assembly. **DropEdge** improves generalization by reducing overfitting. **GAT** boosts precision through focused feature weighting. **Laplacian Regularization** smooths representations while preserving structural cues. Combined, these components yield the best performance across accuracy, F1, NMI, and triplet loss (**Table 2**). Sim2Real is achieved by integrating DASTGCN within an SNN-DDPG framework to transfer learning from simulation to real-world brick assembly. Spatiotemporal graph-based few-shot reinforcement learning enables generalization under variability, bridging simulation-reality gaps and supporting collaborative CAD/CAM workflows.

4 Limitations and Conclusions

Despite the advancements provided by our method, there are inherent limitations. The complexity of graph-based models and the computational demand of ST-GFSL and GRL can limit real-time applications. The implementation of CAD/CAM with DASTGCN, SNN, DDPG, and Differentiable Energy Minimization (DEM) in robotic brick assembly represents a significant step forward in the automation of complex construction tasks with mobile manipulator.

Acknowledgments

This work was supported by the National Science and Technology Council (NSTC) of Taiwan and the Sustainable Interdisciplinary Integrated Project, National Cheng Kung University. The authors express their gratitude to Mi-Chi Wang, Lien-Kai Huang, Cheng-En Tsai, and Chu-Hua Huang for their assistance with hardware and design support. Special thanks to Wen-Yu Su, Yu-Hsiang Tseng, and Ting-Tsung Lin for their collaboration on the mobile manipulator.

References

Rie Ando and Tong Zhang. 2006. Learning on graph with Laplacian regularization. *Advances in neural information processing systems* 19 (2006).

Matan Atad, Jianxiang Feng, Ismael Rodr'iguez, Maximilian Durner, and Rudolph Triebel. 2023. Efficient and Feasible Robotic Assembly Sequence Planning via Graph Representation Learning. *2023 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)* (2023), 8262–8269. https://api.semanticscholar.org/CorpusID:257623086

Davide Chicco. 2021. *Siamese Neural Networks: An Overview.* Springer US, New York, NY, 73–94. doi:10.1007/978-1-0716-0826-5_3

Shengnan Guo, Youfang Lin, Ning Feng, Chao Song, and Huaiyu Wan. 2019. Attention Based Spatial-Temporal Graph Convolutional Networks for Traffic Flow Forecasting. *Proceedings of the AAAI Conference on Artificial Intelligence* 33, 01 (Jul. 2019), 922–929. doi:10.1609/aaai.v33i01.3301922

Timothy P. Lillicrap, Jonathan J. Hunt, Alexander Pritzel, Nicolas Heess, Tom Erez, Yuval Tassa, David Silver, and Daan Wierstra. 2019. Continuous control with deep reinforcement learning. arXiv:1509.02971 [cs.LG] https://arxiv.org/abs/1509.02971

Yu Rong, Wenbing Huang, Tingyang Xu, and Junzhou Huang. 2019. Dropedge: Towards deep graph convolutional networks on node classification. *arXiv preprint arXiv:1907.10903* (2019).

Sven Stumm, Johannes Braumann, Martin von Hilchen, and Sigrid Brell-Çokcan. 2016. On-Site Robotic Construction Assistance for Assembly Using A-Priori Knowledge and Human-Robot Collaboration. In *International Conference on Robotics in Alpe-Adria-Danube Region.* https://api.semanticscholar.org/CorpusID:2825045

Petar Veličković, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Liò, and Yoshua Bengio. 2018. Graph Attention Networks. In *International Conference on Learning Representations.* https://openreview.net/forum?id=rJXMpikCZ