G-SpaNet: Generalized Permutationless Set Assignment for Particle Physics using Symmetry-Preserving Attention

Alexander Shmakov *
Department of Computer Science
University of California Irvine
Irvine, CA
ashmakov@uci.edu

Michael James Fenton *
Department of Physics and Astronomy
University of California Irvine
Irvine, CA
mjfenton@uci.edu

Ta-Wei Ho
Department of Physics and Astronomy
National Tsing Hua University
Hsinchu City, Taiwan

Shih-Chieh Hsu
Department of Physics and Astronomy
University of Washington
Seattle, WA

Daniel Whiteson
Department of Physics and Astronomy
University of California Irvine
Irvine, CA

Pierre Baldi
Department of Computer Science
University of California Irvine
Irvine, CA

Abstract

We introduce a novel method for constructing symmetry-preserving attention networks which reflect the natural invariances of the jet-parton assignment problem to efficiently find assignments without evaluating all permutations. This general approach is applicable to arbitrarily complex configurations and significantly outperforms current methods, improving reconstruction efficiency between 19% - 35% on benchmark problems while decreasing inference time by two to five orders of magnitude, making many important and previously intractable cases tractable. A full code repository containing a general library, the specific configurations used, and a complete dataset release, are available at https://github.com/Alexanders101/SPANet.

1 Introduction

Reconstructing all-jet events at the Large Hadron Collider (LHC) consists of assigning specific labels to a *variable-size set* of observed jets, each represented by a fixed-size vector of physical measurements of the jet. Each label represents a decay product of an intermediate heavy particle and must be uniquely assigned to one of the jets if the mass and momentum of the heavy particle is to be measured. Current solutions consider all possible assignment permutations, an ineffective strategy for which the combinatorial computation cost grows so rapidly with the number of jets that they are rendered unusable in particle collisions with more than a handful of jets.

Attention-based methods have achieved state-of-the-art results in natural language processing problems such as translation [1, 2, 3, 4], where variable-length sequences are common. Transformers [5] stand out as particularly promising for set assignment due to their fundamental invariance with respect to the order of the input sequence [6]. Transformers are especially effective at modeling

*= These Authors contributed Equally

Fourth Workshop on Machine Learning and the Physical Sciences (NeurIPS 2021).

variable-length sets because they can learn combinatorial relationships between set elements with polynomial run-time. We present a novel attention-based method which expands on the transformer to tackle the unique symmetries and challenges present in LHC event reconstruction.

2 Event Reconstruction at the LHC

The detectors at the Large Hadron Collider measure particles produced in high energy proton collisions. In each collision event, heavy, unstable particles such as top quarks, Higgs-bosons, or $W-$ & Z-bosons may be created. These *resonance particles* decay too quickly ($< 10^{-20}$s) to be directly detected [7]. To study them, experimentalists must reconstruct them from their decay products, which we will refer to as *partons*. When these partons are *quarks*, they appear in the detectors as *jets*; collimated streams of particles. However, collisions commonly produce additional jets other than just those from the resonance particles. In order to reconstruct the resonance particles, the jets produced from the partons must be identified. Event reconstruction therefore reduces to uniquely assigning a collection of truth labels - the parent partons - to a collection of observed jets. We refer to this as the *jet-parton assignment* problem.

In many cases, the reconstruction task is insensitive to swapping labels. For example, since jet charge can usually not be reliable measured, the top and anti-top in a $t\bar{t}$ event are effectively indistinguishable. We refer to this kind of invariance in the resonance particles as *event-level symmetries*. Similarly, a W-boson decays to a quark and anti-quark, and inverting the labels leads to the same reconstructed W-boson. We refer to these lower-level invariances on the jet-labels as *particle-level symmetries*. Exploiting such symmetries is crucial for effective event reconstruction, especially in complex events with many jets where these invariances greatly reduce the number of possible jet assignments. By incorporating symmetries into our reconstruction models, we may substantially simplify the modeling task. We refer to the complete specification of an event's particles and all of their associated symmetries as its *topology*.

We study event reconstruction for three benchmark processes, although the techniques generalize to arbitrary event topologies; top-antitop production ($t\bar{t}$), top-associated Higgs production ($t\bar{t}H$), and 4-top production ($t\bar{t}t\bar{t}$). We exploit the jet symmetries between the quark jets from the bosons as well as the particle symmetries between the top quarks to aid us in finding solutions to these problems.

As a baseline comparison, we also implement the χ^2 method [8, 9]. This is an example of a permutation approach to set assignment, in which every possible jet permutation is explicitly tested to produce the highest scoring assignment. While effective, this method suffers from exponential run-time with respect to the number of jets. This quickly becomes a limiting factor in large datasets, and makes complex topologies intractable.

3 Symmetry-Preserving Attention Networks

We introduce a general architecture for jet-parton assignment named G-SPANET: a generalization of an attention-based neural network first described for a specific topology in [10]. The transformer encoders employ multi-head self-attention [5] with *position-independent* jet embeddings to preserve permutation invariance on the input. We solve the jet-parton assignment per-particle using a generalized form of *Symmetric Tensor Attention* (STA) [10] and combine them using an improved *Combined Symmetric Loss* which can learn from partial events. This two-step approach allows us to naturally handle both symmetries described in Section 2 in a general manner.

Symmetric Tensor Attention Every resonance particle p has associated with it k_p partons. STA takes a set of transformer-encoded jets $X_p \in \mathbb{R}^{N\times D}$ - with N the total number of jets and D the latent dimensionality — to produce a rank-k_p tensor $\mathcal{P}_p \in \mathbb{R}^{N\times N\times\cdots\times N}$ such that $\sum \mathcal{P}_p = 1$. $\mathcal{P}_p$ represents a joint distribution over k_p-jet assignments indicating the probability that any particular combination of jets is the correct sub-assignment for particle p. We represent jet symmetries applicable to the current partition with a *particle-level* permutation group $G_p \subseteq S_{k_p}$ which acts on k_p-tuples and defines an equivalence relation over indistinguishable jet assignments. In practice, this equivalence relation is satisfied when the indices of $\mathcal{P}_p$ commute with respect to G_p.

$$\forall\sigma \in G_p \ \left(j_1, j_2, \ldots, j_{k_p}\right) \simeq \left(j_{\sigma(1)}, j_{\sigma(2)}, \ldots, j_{\sigma(k_p)}\right) \iff \mathcal{P}_p^{j_1 j_2 \ldots j_{k_p}} = \mathcal{P}_p^{j_{\sigma(1)} j_{\sigma(2)} \cdots j_{\sigma(k_p)}} \quad (1)$$

We enforce this commutativity using general dot-product attention [1], where the mixing weights mimic the output's symmetries. An STA layer contains a single rank-p_k parameter tensor $\Theta \in$

$\mathbb{R}^{D\times D\times\cdots\times D}$ and performs the following computations, expressed using Einstein summation notation:

$$\mathcal{S}^{i_1 i_2 \ldots i_{k_p}} = \sum_{\sigma \in G_p} \Theta^{i_{\sigma(1)} i_{\sigma(2)} \cdots i_{\sigma(k_p)}} \tag{2}$$

$$\mathcal{O}^{j_1 j_2 \ldots j_{k_p}} = X^{j_1}_{i_1} X^{j_2}_{i_2} \ldots X^{j_{p_k}}_{i_{p_k}} \mathcal{S}^{i_1 i_2 \ldots i_{k_p}} \tag{3}$$

$$\mathcal{P}_p^{j_1 j_2 \ldots j_{k_p}} = \frac{\exp\left(\mathcal{O}^{j_1 j_2 \cdots j_{k_p}}\right)}{\sum_{j_1, j_2, \ldots, j_{p_k}} \exp\left(\mathcal{O}^{j_1 j_2 \cdots j_{k_p}}\right)} \tag{4}$$

STA first constructs a G_p-symmetric tensor $\mathcal{S}$ to ensure the output's indices will also be G_p-symmetric. STA then performs a generalized dot-product attention which represents all k_p-wise similarities in the input sequence (Equation 3). This is the most expensive operation in network, with a time and space complexity of $O(N^{k_p})$. We then normalize the output tensor $\mathcal{O}$ by performing a k_p-dimensional softmax, producing a final joint distribution $\mathcal{P}_p$ (Equation 4).

Combined Symmetric Loss and Partial Event Reconstruction The symmetric attention layers produce assignments $\{\mathcal{P}_1, \mathcal{P}_2, \ldots, \mathcal{P}_m\}$ for each particle. With true sub-assignments targets $\{\mathcal{T}_1, \mathcal{T}_2, \ldots, \mathcal{T}_m\}$, the loss is simply the cross entropy, $CE(\mathcal{P}_p, \mathcal{T}_p)$, for each resonance particle p. We encode particle symmetries (Section 2) using an *event-level* permutation group $G_E \subseteq S_m$ and a symmetrized loss. G_E induces an equivalence relation over particles in a manner similar to Equation 1: $\forall \sigma \in G_E,\ (\mathcal{T}_1, \mathcal{T}_2, \ldots, \mathcal{T}_m) \simeq (\mathcal{T}_{\sigma(1)}, \mathcal{T}_{\sigma(2)}, \ldots, \mathcal{T}_{\sigma(m)})$. We incorporate these symmetries into the loss function by allowing the network to fit to *any* equivalent jet assignment by fitting to the minimum attainable loss within a given equivalence class.

Though each parton is usually expected to produce a jet, one or more of these may sometimes not be detected and therefore the event becomes impossible to reconstruct. This may be due to limited detector acceptance, merging jets, or other idiosyncrasies. Events with more partons have a higher probability that one or more of the particles will be missing a jet. Limiting our dataset to only complete events significantly reduces the available training examples in complex event configurations. Baseline permutation methods struggle with partial events because their scoring functions are typically only valid for full permutations. Due to G-SPANET's partitioned approach to jet-parton assignment, we can modify our loss to recover any particles which *are* present in the event and still provide a meaningful training signal from these *partial events*. This not only reduces the required training dataset size, but also may reduce generalization bias because such events occur in real collision data.

We mark particles in an event with a masking value $\mathcal{M}_p \in \{0, 1\}$ and we only include the loss contributed by reconstructable particles. We scale the loss based on the distribution of events present in the training dataset by computing the effective class count for each partial combination $CB(\mathcal{M}_1, \mathcal{M}_2, \ldots, \mathcal{M}_m)$ [11]. The loss function is thus defined as

$$\mathcal{L}^{masked}_{min} = \min_{\sigma \in G_E} \left(\sum_{i=1}^{m} \frac{\mathcal{M}_{\sigma(i)} CE(\mathcal{P}_i, \mathcal{T}_{\sigma(i)})}{CB\left(\mathcal{M}_{\sigma(1)}, \mathcal{M}_{\sigma(2)}, \ldots, \mathcal{M}_{\sigma(m)}\right)} \right) \tag{5}$$

In the event that we assign a jet to more than one parton, we select the higher probability assignment first and re-evaluate the remaining $\mathcal{P}$'s to select the best non-contradictory assignments.

4 Experiments

Datasets All processes are generated at $\sqrt{s} = 13$ TeV using MadGraph_aMC@NLO [12] (v2.7.2, NCSA license), Pythia8 [13] (v8.2, GPL-2), and Delphes [14] (v3.4.1, GPL-3) with the ATLAS parameterization. All W-bosons are forced to decay to a pair of light quarks, and Higgs Bosons to b-quarks. Jets are reconstructed with the $R = 0.4$ anti-k_T algorithm [15], $p_T \geq 25$ GeV and $|\eta| < 2.5$. The 4-vector (p_T, η, ϕ, M) of each jet, as well as the boolean result of the a p_T dependent b-tagging algorithm, are inputs to the networks[2]. Truth assignments are generated by exclusively matching the partons from the event record to the reconstructed jets via the requirement $\sqrt{\Delta\eta^2 + \Delta\phi^2} < 0.4$. Every event must contain at least as many jets as expected in the final state, at least two of which must be b-tagged. We thus keep 10M, 14.3M, and 5.8M events out of a total 60M, 100M, and 100M events generated for $t\bar{t}$, $t\bar{t}H$, and $t\bar{t}t\bar{t}$ respectively. We used 90% of events for training, 5% for validation and hyperparameter optimization, and the final 5% for testing.

[2] Additional input features may trivially be added in future experiments.

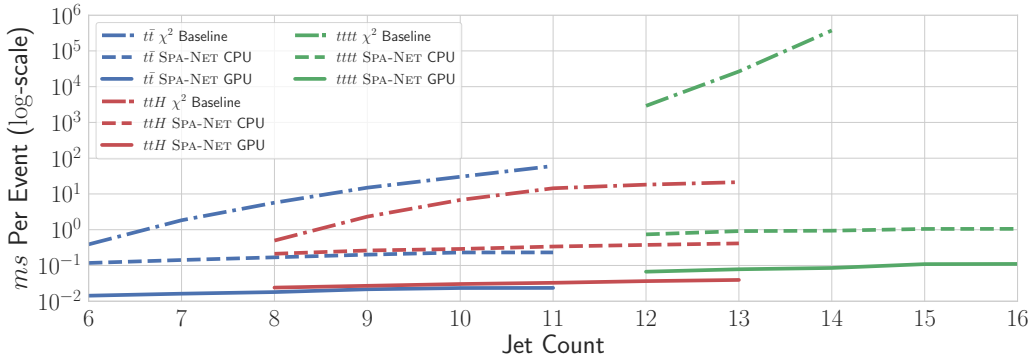

Figure 1: Average inference time over 1000 events across models and jet multiplicities, G-SPANET evaluated with a batch size of 1024 events.

G-SPANET Training Hyperparameters are chosen using the Sherpa hyperparameter optimization library [16]. Each network was trained using four Nvidia GeForce 3090 GPUs for 50 epochs using the *AdamW* optimizer [17] with L_2 regularization on all parameter weights. To improve transformer convergence we anneal the learning rate following a cosine schedule [18], performing a warm restart every 10 epochs. Training took a total of 4 to 6 hours depending on topology.

5 Performance

Reconstruction Efficiency We measure model performance via *reconstruction efficiency*, the proportion of correctly assigned jets. *Complete Events* are those in which all resonance particles are fully reconstructable while *Partial Events* are those events in which at least one but not all resonance particles are reconstructable.The *Event Fraction* is the percentage of total events included in the denominator for the efficiency calculations. *Event Efficiency* is defined as the proportion of events in which all jets associated with reconstructable particles are correctly assigned. We also report the per-particle efficiencies.

Benchmark $t\bar{t}$ reconstruction efficiency is presented in Table 1. We found that G-SPANET outperforms the χ^2 method in every category, with efficiencies consistently around 20% higher with overall performance on all events. As expected, efficiencies drop off as N_{jets} increases, and are generally higher in Complete Events. $t\bar{t}H$ reconstruction efficiency is presented in Table 2, and $t\bar{t}t\bar{t}$ reconstruction efficiency is presented Table 3, both in events with ≥ 4 b-jets. Note that while G-SPANET is trained on events with ≥ 2 b-jets, the χ^2 method is intractable here due to the additional ambiguities which generate more permutations. We do not show results for the χ^2 in the $t\bar{t}t\bar{t}$ case because the CPU time required simply made it intractable to calculate sufficient statistics for this problem. The performance on this dataset emphasizes the

	N_{jets}	Event Fraction	G-SPANET Efficiency Event	G-SPANET Efficiency Top Quark
All Events	== 12	0.219	0.276	0.484
	== 13	0.304	0.247	0.474
	$\geq$ 14	0.450	0.198	0.450
	Inclusive	**0.974**	**0.231**	**0.464**
Complete Events	== 12	0.005	0.350	0.617
	== 13	0.016	0.249	0.567
	$\geq$ 14	0.044	0.149	0.504
	Inclusive	**0.066**	**0.191**	**0.529**

Table 3: G-SPANET Reconstruction Efficiency for $t\bar{t}t\bar{t}$ events at various jet selections.

	N_{jets}	Event Fraction	G-SPANET Efficiency Event	G-SPANET Efficiency Top Quark	χ^2 Efficiency Event	χ^2 Efficiency Top Quark
All Events	== 6	0.245	0.643	0.696	0.461	0.523
	== 7	0.282	0.601	0.667	0.408	0.476
	$\geq$ 8	0.320	0.528	0.613	0.313	0.395
	Inclusive	**0.848**	**0.586**	**0.653**	**0.387**	**0.457**
Complete Events	== 6	0.074	0.803	0.837	0.664	0.696
	== 7	0.105	0.667	0.754	0.457	0.556
	$\geq$ 8	0.145	0.521	0.662	0.281	0.429
	Inclusive	**0.325**	**0.633**	**0.732**	**0.426**	**0.532**

Table 1: Reconstruction Efficiency for $t\bar{t}$ events at various jet selections.

	N_{jets}	Event Fraction	G-SPANET Efficiency Event	Higgs	Top	χ^2 Efficiency Event	Higgs	Top
All Events	== 8	0.261	0.370	0.497	0.540	0.056	0.193	0.092
	== 9	0.313	0.343	0.492	0.514	0.053	0.160	0.102
	$\geq$ 10	0.313	0.294	0.472	0.473	0.031	0.150	0.056
	Inclusive	**0.972**	**0.330**	**0.485**	**0.502**	**0.045**	**0.164**	**0.081**
Complete Events	== 8	0.042	0.532	0.657	0.663	0.040	0.220	0.135
	== 9	0.070	0.422	0.601	0.596	0.019	0.152	0.079
	$\geq$ 10	0.115	0.306	0.545	0.523	0.004	0.126	0.073
	Inclusive	**0.228**	**0.383**	**0.583**	**0.572**	**0.016**	**0.153**	**0.087**

Table 2: Reconstruction Efficiencies for $t\bar{t}H$ events at various jet selections.

importance of the partial-event training approach introduced in Section 3, given that only 6.6% of the training events were complete. We further find that including partial events in the training improves performance on these events by around 5% for $t\bar{t}H$ events, without affecting performance on complete events.

Timing Figure 1 shows the average evaluation time per event for each benchmark topology, as a function of N_{jets}, for the χ^2 method as well as G-SPANET evaluated on both a CPU and GPU. G-SPANET represents an exponential improvement in run-time on larger events, with an additional factor of 10 improvement when using a GPU. Run-time evaluation was performed on an Intel I7 10700K CPU with 64 GB of RAM and an Nvidia 3080 GPU with 10 GB of VRAM.

6 Conclusions

We have introduced G-SPANET, a network architecture based on a novel attention mechanism with embedded symmetries, that performs set assignment tasks in a highly efficient manner. We have further released a BSD-3 licensed python package which can generate appropriate architectures for arbitrary topologies given a simple configuration file. We have presented three benchmark use cases of varying complexity from the world of particle physics that demonstrate excellent performance, both in terms of the efficiency to predict the correct assignments as well as the computational overhead.

Crucially, the computational overhead scales efficiently with the complexity of the problem compared to existing benchmarks. Applications are not limited to the specific benchmarks we have presented, and the techniques may be generalized to many other LHC use-cases. We have further developed novel techniques which reduce the amount of required training data relative to how neural network training is usually performed in high energy physics, something that is crucial as simulation requirements at the LHC continue to grow. All of these developments combined make new analyses tractable for the first time, and may thus be crucial in the discovery of new physics in the LHC era and beyond.

7 Acknowledgements

DW and MF are supported by DOE grant DE-SC0009920. S.-C. Hsu is supported by DOW Award DE-SC0015971. AS and PB are in part supported by grants NSF NRT 1633631 and ARO 76649-CS to PB. T.-W.H. is supported by Taiwan MoST grant MOST-107-2112- M-007-029-MY3.

References

[1] Thang Luong, Hieu Pham, and Christopher D. Manning. Effective approaches to attention-based neural machine translation. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 1412–1421, Lisbon, Portugal, September 2015. Association for Computational Linguistics.

[2] Dzmitry Bahdanau, Kyung Hyun Cho, and Yoshua Bengio. Neural machine translation by jointly learning to align and translate. January 2015. 3rd International Conference on Learning Representations, ICLR 2015 ; Conference date: 07-05-2015 Through 09-05-2015.

[3] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota, June 2019. Association for Computational Linguistics.

[4] Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9, 2019.

[5] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. Attention is all you need. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems 30*, pages 5998–6008. Curran Associates, Inc., 2017.

[6] Juho Lee, Yoonho Lee, Jungtaek Kim, Adam Kosiorek, Seungjin Choi, and Yee Whye Teh. Set transformer: A framework for attention-based permutation-invariant neural networks. volume 97 of *Proceedings of Machine Learning Research*, pages 3744–3753, Long Beach, California, USA, 09–15 Jun 2019. PMLR.

[7] P.A. Zyla et al. Review of Particle Physics. *PTEP*, 2020(8):083C01, 2020.

[8] ATLAS Collaboration. Measurements of top-quark pair single- and double-differential cross-sections in the all-hadronic channel in pp collisions at $\sqrt{s} = 13$ TeV using the ATLAS detector. *JHEP*, 01:033, 2021.

[9] ATLAS Collaboration. Top-quark mass measurement in the all-hadronic $t\bar{t}$ decay channel at $\sqrt{s} = 8$ TeV with the ATLAS detector. *JHEP*, 09:118, 2017.

[10] Michael James Fenton, Alexander Shmakov, Ta-Wei Ho, Shih-Chieh Hsu, Daniel Whiteson, and Pierre Baldi. Permutationless many-jet event reconstruction with symmetry preserving attention networks, October 2020. arXiv:2010.09206.

[11] Yin Cui, Menglin Jia, Tsung-Yi Lin, Yang Song, and Serge Belongie. Class-balanced loss based on effective number of samples. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9268–9277, 2019.

[12] J. Alwall, R. Frederix, S. Frixione, V. Hirschi, F. Maltoni, O. Mattelaer, H. S. Shao, T. Stelzer, P. Torrielli, and M. Zaro. The automated computation of tree-level and next-to-leading order differential cross sections, and their matching to parton shower simulations. *JHEP*, 07:079, 2014.

[13] Torbjörn Sjöstrand, Stefan Ask, Jesper R. Christiansen, Richard Corke, Nishita Desai, Philip Ilten, Stephen Mrenna, Stefan Prestel, Christine O. Rasmussen, and Peter Z. Skands. An introduction to PYTHIA 8.2. *Comput. Phys. Commun.*, 191:159–177, 2015.

[14] J. de Favereau, C. Delaere, P. Demin, A. Giammanco, V. Lemaître, A. Mertens, and M. Selvaggi. DELPHES 3, A modular framework for fast simulation of a generic collider experiment. *JHEP*, 02:057, 2014.

[15] Matteo Cacciari, Gavin P. Salam, and Gregory Soyez. The anti-k_t jet clustering algorithm. *JHEP*, 04:063, 2008.

[16] Lars Hertel, Julian Collado, Peter Sadowski, Jordan Ott, and Pierre Baldi. Sherpa: Robust hyperparameter optimization for machine learning. *SoftwareX*, 2020. Also arXiv:2005.04048. Software available at: https://github.com/sherpa-ai/sherpa.

[17] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *International Conference on Learning Representations*, 2019.

[18] Ilya Loshchilov and Frank Hutter. SGDR: Stochastic gradient descent with restarts. In *International Conference on Learning Representations*, 2017.

NIPS Checklist

1. For all authors...
   (a) Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   [Yes] We introduce a novel method for the problem described and demonstrate that it outperforms currently used baselines on several benchmark problems in both performance and speed.

   (b) Did you describe the limitations of your work?

   [Yes] We note in the text that we have only tested the method on quark jets, and other experiment types could be analyzed in the future. We also mention potential improvements to the truth-matching approach that we use for labelling which limits the possible final assignments. Finally, we discuss potential problems of using models trained using simulator data on real world observations.

   (c) Did you discuss any potential negative societal impacts of your work?

   [N/A] Our work applies primarily to fundamental problems in physics and is far removed from any data specific to people or society in general. Additionally, our work aims to reduce computational time currently spent at the LHC, thereby reducing the environmental costs of operating the LHC.

   (d) Have you read the ethics review guidelines and ensured that your paper conforms to them?

   [Yes]

2. If you are including theoretical results...
   (a) Did you state the full set of assumptions of all theoretical results?

   [Yes] We define the formal specification of our problem statement and the required mathematical structures necessary to define the architecture in Section 3. All of our theoretical claims about symmetry preservation originate from these formal assumptions.

   (b) Did you include complete proofs of all theoretical results?

   [Yes] We include proofs for the assertions that we make when describing the novel attention mechanism. These are described in Section 3, and proofs are provided in Appendix A.

3. If you ran experiments...
   (a) Did you include the code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL)?

   [Yes] We have prepared a public code repository for training and evaluating the architectures presented. This repository also includes links to the specific datasets that we generated for each event type. We will include a link to this repository once the paper is deanonymized.

   (b) Did you specify all the training details (e.g., data splits, hyperparameters, how they were chosen)?

   [Yes] We describe the exact method for simulating the data and the training/testing splits in Section 4. We include some high level information about the training process in Section 3. We define all domain-specific parameters for baseline models, a detailed description of hyperparameters, and mathematical details for the architecture in the appendices.

   (c) Did you report error bars (e.g., with respect to the random seed after running experiments multiple times)?

[No] We do not include error bars are not included (eg Tables 1,2,3), they are negligible as we have sufficient testing data statistics.

(d) Did you include the total amount of compute and the type of resources used (e.g., type of GPUs, internal cluster, or cloud provider)?

[Yes] We include our training hardware in Section 4 when we describe the training procedure, as well as the total time for hyperparameter optimization and training. We also include the hardware which was used to perform our speed benchmarks in the "Timing" paragraph of Section 5.

4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets...

(a) If your work uses existing assets, did you cite the creators?

[Yes] We use many physics packages to simulate our data and we cite the original papers as well as the specific versions used for our experiments in Section 4.

(b) Did you mention the license of the assets?

[Yes] We specify the licenses of all software used during simulation as well as the license for our own code release.

(c) Did you include any new assets either in the supplemental material or as a URL?

[Yes] We provide all training datasets that we generated for each event type in the public code release.

(d) Did you discuss whether and how consent was obtained from people whose data you're using/curating?

[N/A]

(e) Did you discuss whether the data you are using/curating contains personally identifiable information or offensive content?

[N/A]

5. If you used crowdsourcing or conducted research with human subjects...

(a) Did you include the full text of instructions given to participants and screenshots, if applicable?

[N/A]

(b) Did you describe any potential participant risks, with links to Institutional Review Board (IRB) approvals, if applicable?

[N/A]

(c) Did you include the estimated hourly wage paid to participants and the total amount spent on participant compensation?

[N/A]

8 Appendix A: Symmetric Tensor Attention Proofs

Theorem 8.1. *Given a permutation group $G \subseteq S_k$ for any integer k, a rank-k parameter tensor $\Theta \in \mathbb{R}^{D \times D \times \cdots \times D}$, and a set of input vectors $X \in \mathbb{R}^{N \times D}$ the following set of operations*

$$\mathcal{S}^{i_1 i_2 \ldots i_k} = \sum_{\sigma \in G} \Theta^{i_{\sigma(1)} i_{\sigma(2)} \cdots i_{\sigma(k)}} \tag{6}$$

$$\mathcal{O}^{j_1 j_2 \ldots j_k} = X_{i_1}^{j_1} X_{i_2}^{j_2} \ldots X_{i_k}^{j_k} \mathcal{S}^{i_1 i_2 \ldots i_k} \tag{7}$$

$$\mathcal{P}^{j_1 j_2 \ldots j_k} = \frac{\exp\left(\mathcal{O}^{j_1 j_2 \ldots j_k}\right)}{\sum_{j_1, j_2, \ldots, j_k} \exp\left(\mathcal{O}^{j_1 j_2 \ldots j_k}\right)} \tag{8}$$

will produce a an output tensor, $\mathcal{P}$, that is G-symmetric. That is,

$$\forall \sigma \in G,\ \mathcal{P}^{j_1 j_2 \ldots j_k} = \mathcal{P}^{j_{\sigma(1)} j_{\sigma(2)} \cdots j_{\sigma(k)}}$$

Proof. In order to prove that the output, $\mathcal{P}$, is G-symmetric, it is sufficient to prove that every step produces a G-symmetric tensor. We will now prove that the result from all three steps will be G-symmetric.

- We will first prove that the output to Equation 6, which is known as the (unnormalized) *symmetric part* of tensor Θ, will be G-symmetric. That is,

  $$\forall \tau \in G,\ \mathcal{S}^{j_1 j_2 \ldots j_k} = \mathcal{S}^{j_{\tau(1)} j_{\tau(2)} \cdots j_{\tau(k)}}$$

  Since G is a group, for every element $\nu \in G$, there exists a unique $\sigma \in G$ such that $\nu = \sigma\tau$. This is a consequence of the unique inverse property of groups, forcing that element to be $\sigma = \nu\tau^{-1}$. Therefore,

  $$\begin{aligned}
  \mathcal{S}^{j_{\tau(1)} j_{\tau(2)} \cdots j_{\tau(k)}} &= \sum_{\sigma \in G} \Theta^{i_{\sigma(\tau(1))} i_{\sigma(\tau(2))} \cdots i_{\sigma(\tau(k))}} \\
  &= \sum_{\nu \in G} \Theta^{i_{(\nu\tau^{-1}\tau)(1)} i_{(\nu\tau^{-1}\tau)(2)} \cdots i_{(\nu\tau^{-1}\tau)(k)}} \\
  &= \sum_{\nu \in G} \Theta^{i_{\nu(1)} i_{\nu(2)} \cdots i_{\nu(k)}} \\
  &= \mathcal{S}^{j_1 j_2 \ldots j_k}
  \end{aligned}$$

- For Equation 7, we use the same X tensor k times in the expression. Since these tensors are all identical, they are trivially symmetric since and we can freely swap the order of the X tensors as long as we apply an inverse permutation to another set of indices. Furthermore, since $\mathcal{S}$ is G-symmetric from the previous step, it can also freely permute its indices according to G. Therefore,

  $$\begin{aligned}
  \forall \sigma \in G, \mathcal{O}^{j_{\sigma(1)} j_{\sigma(2)} \cdots j_{\sigma(k)}} &= X_{i_1}^{j_{\sigma(1)}} X_{i_2}^{j_{\sigma(2)}} \ldots X_{i_k}^{j_{\sigma(k)}} \mathcal{S}^{i_1 i_2 \ldots i_k} \\
  &= X_{i_{\sigma^{-1}(1)}}^{j_1} X_{i_{\sigma^{-1}(2)}}^{j_2} \ldots X_{i_{\sigma^{-1}(k)}}^{j_k} \mathcal{S}^{i_1 i_2 \ldots i_k} \\
  &= X_{i_1}^{j_1} X_{i_2}^{j_2} \ldots X_{i_k}^{j_k} \mathcal{S}^{i_{\sigma(1)} i_{\sigma(2)} \cdots i_{\sigma(k)}} \\
  &= X_{i_1}^{j_1} X_{i_2}^{j_2} \ldots X_{i_k}^{j_k} \mathcal{S}^{i_1 i_2 \ldots i_k} \\
  &= \mathcal{O}^{j_1 j_2 \ldots j_k}
  \end{aligned}$$

- For Equation 8, the operations are performed element-wise to every element in $\mathcal{O}$ and the normalisation term is simply the sum of all elements in $\exp(\mathcal{O})$. Since summation is commutative,

  $$\forall \sigma \in G,\ \sum_{j_1, j_2, \ldots, j_k} \exp\left(\mathcal{O}^{j_1 j_2 \ldots j_k}\right) = \sum_{j_{\sigma(1)}, j_{\sigma(2)}, \ldots, j_{\sigma(k)}} \exp\left(\mathcal{O}^{j_{\sigma(1)} j_{\sigma(2)} \cdots j_{\sigma(k)}}\right)$$

Combining the fact that both the normalisation term and $\mathcal{O}$ are both G-symmetric, we find that the output is also G-symmetric.

$$\begin{aligned}\forall \sigma \in G,\ \mathcal{P}^{j_{\sigma(1)} j_{\sigma(2)} \cdots j_{\sigma(k)}} &= \frac{\exp\left(\mathcal{O}^{j_{\sigma(1)} j_{\sigma(2)} \cdots j_{\sigma(k)}}\right)}{\sum_{i_{\sigma(1)}, i_{\sigma(2)}, \ldots, i_{\sigma(k)}} \exp\left(\mathcal{O}^{i_{\sigma(1)} i_{\sigma(2)} \cdots i_{\sigma(k)}}\right)} \\ &= \frac{\exp\left(\mathcal{O}^{j_1 j_2 \cdots j_k}\right)}{\sum_{i_1, i_2, \ldots, i_k} \exp\left(\mathcal{O}^{i_1 i_2 \ldots i_k}\right)} \\ &= \mathcal{P}^{j_1 j_2 \cdots j_k}\end{aligned}$$

□

8.1 Run-time Complexity

For the following sections, we will treat the network's hidden representation dimension D as a constant. This is because this value is a hyperparameter which may be adjusted freely, although we also provide the run-time expressions with D present.

- Equation 6. This expression is simply an element-wise sum over all possible elements of group G and tensor Θ. The run-time of this step is therefore exponential with respect to the number of partons in each partition.

$$\begin{aligned}O\left(|G| D^k\right) &= O\left(k! D^k\right) \\ &= O\left(k^k D^k\right) \\ &= O\left((Dk)^k\right) \\ &= O\left(k^k\right)\end{aligned}$$

- Equation 7. This expression evaluates a generalized tensor-product between k rank-2 tensors and one rank-k tensor. The output will be rank-k tensor with sizes $N \times N \times \cdots \times N$. For each of these outputs, the operation must perform a rank-k tensor multiplication with sizes $D \times D \times \cdots \times D$. The run-time of this step is therefore exponential with respect to the number of partons in each partition. We note that this is only the naive run-time and many tensor-multiplication libraries will not use divide-and-conquer algorithms to reduce the $O\left(D^k\right)$ multiplication operation.

$$\begin{aligned}O\left(N^k D^k\right) &= O\left((ND)^k\right) \\ &= O\left(N^k\right)\end{aligned}$$

- Equation 8. The normalization factor can be pre-computed once for every element of $\mathcal{O}$. This expression then reduces to simply an element-wise exponentiation and division over all $O\left(N^k\right)$ elements in $\mathcal{O}$

The total run-time complexity of the symmetric tensor attention layer assuming that D is constant is therefore simply

$$O(k^k + N^k)$$

9 Appendix B: SPANet Modifications

9.1 Soft Loss Function

When constructing the symmetric loss function, we use the minimum loss over all equivalent particle orderings as our optimization objective. However, this might cause instability on events where the network is unsure, causing the loss function to flip every epoch for that event. In order to prevent this and maintain a continually differentiable loss function, we experiment with an alternative loss

based on the soft min function.

$$\mathcal{L}_{softmin} = \underset{\sigma \in G_E}{\text{soft min}} \sum_{i=1}^{m} CE(\mathcal{P}_i, \mathcal{T}_{\sigma(i)})$$

where

$$\text{soft min}\{x_1, x_2, \ldots, x_k\} = \sum_{j=1}^{k} \frac{e^{-x_j}}{\sum_{i=1}^{k} e^{-x_i}} x_j$$

9.2 Balanced Loss Scaling

We experiment with balancing the loss based on the prevalence of each combination of particles in the target set. This is primarily to prevent the network from ignoring rare events such as the complete $t\bar{t}t\bar{t}$ event when performing partial event training. If there is a large imbalance between classes, such as when events with fewer particle are more prevalent, this could cause the network to bias its results towards those more common events and worsen performance on full events.

We compute the class balance term $CB(\mathcal{M}_1, \mathcal{M}_2, \ldots, \mathcal{M}_m)$ where the $\mathcal{M}_p$ terms represent binary values indicating if a particle p is present or not in the event and m is the total number of particles. If $\mathcal{M}_p = 1$, then p is fully reconstructable in the given event, and if $\mathcal{M}_p = 0$, then at least one parton associated with particle p is not detectable.

Assume we have a dataset of size N of such events, each with their own masking vector for each possible particle $\mathcal{M}_p^j$ for $1 \leq j \leq N$ and $1 \leq p \leq m$. We will keep the particle indices in the subscript and the dataset indices in the superscript. Assume we also have an *event-level* permutation group $G_E \subseteq S_m$ (Section 3). We define our class balance, CB, based on a symmetric version of *effective class count* [11].

First, we will define a counting function. Let $\mathbb{1}_P$ be the selection function for predicate P. This is,

$$\mathbb{1}_P = \begin{cases} 1 & \text{if } P \text{ is } True \\ 0 & \text{Otherwise} \end{cases}$$

Next, define label-counting function C which simply counts how many times a particular arrangement of masking values appears in our dataset.

$$C(\mathcal{M}_1, \mathcal{M}_2, \ldots, \mathcal{M}_m) = \sum_{j=1}^{N} \prod_{p=1}^{m} \mathbb{1}_{\mathcal{M}_p^j = \mathcal{M}_p}$$

Such a counting function does not account for the equivalent particle assignments that are induced by our event-level group G_E. To accommodate particle symmetries, we create a symmetric counting function S which counts not only the presence of any particular arrangement of masking values, but also all *equivalent* arrangements.

$$S(\mathcal{M}_1, \mathcal{M}_2, \ldots, \mathcal{M}_m) = \sum_{\sigma \in G_E} \sum_{j=1}^{N} \prod_{p=1}^{m} \mathbb{1}_{\mathcal{M}_p^j = \mathcal{M}_{\sigma(p)}}$$

Notice that this definition guarantees that any two equivalent masking value sets will have identical symmetric class counts.

$$\forall \sigma \in G_E,\ S(\mathcal{M}_1, \mathcal{M}_2, \ldots, \mathcal{M}_m) = S(\mathcal{M}_{\sigma(1)}, \mathcal{M}_{\sigma(2)}, \ldots, \mathcal{M}_{\sigma(m)})$$

We set the scale β in effective class definition based on the size of our dataset N.

$$\beta = 1 - 10^{-\log_{10} N}$$

Finally, We define the class balance (CB) as the normalized values of the effective class counts (ECC) [11]

$$ECC(\mathcal{M}_1, \mathcal{M}_2, \ldots, \mathcal{M}_m) = \frac{1 - \beta^{S(\mathcal{M}_1, \mathcal{M}_2, \ldots, \mathcal{M}_m)}}{1 - \beta}$$

$$CB(\mathcal{M}_1, \mathcal{M}_2, \ldots, \mathcal{M}_m) = \frac{ECC(\mathcal{M}_1, \mathcal{M}_2, \ldots, \mathcal{M}_m)}{\sum_{\mathcal{M}' \in \{0,1\}^m} ECC(\mathcal{M}'_1, \mathcal{M}'_2, \ldots, \mathcal{M}'_m)}$$

10 Appendix C: Hyperparameters

Parameter	Benchmark Problems		
	$t\bar{t}$	$t\bar{t}H$	$t\bar{t}t\bar{t}$
Training Epochs	50	50	50
Learning Rate	0.0015	0.00302	0.0015
Batch Size	2048	2048	2048
Dropout Percentage	0.1	0.1	0.1
L_2 Gradient Clipping	N/A	0.1	N/A
L_2 Weight Normalization	0.0002	0.0000683	0.0002
Hidden Dimensionality	128	128	128
Central Encoder Count	6	5	2
Branch Encoder Count	3	5	7
Partial Event Training	Yes	Yes	Yes
Loss Scaling	Yes	Yes	Yes
Loss Type	$\mathcal{L}_{\text{min}}$	$\mathcal{L}_{\text{soft min}}$	$\mathcal{L}_{\text{soft min}}$
Cosine Annealing Cycles	5	5	5

Table 4: A complete table of all hyper-parameters used during G-SPANET training on all benchmark problems.

11 Appendix D: χ^2 Method Details

In Section 2, we introduce the χ^2 method as a baseline comparison for reconstructing $t\bar{t}$ and $t\bar{t}H$ events. This is a standard benchmark against which we can compare the results from G-SPANET, and has been used in multiple published results, such as [9, 8]. However, no such benchmark exists for the $t\bar{t}H$ and $t\bar{t}t\bar{t}$ topologies. We thus extend the χ^2 method to these topologies in a simple way in order to have a benchmark to compare against.

$$\chi^2_{t\bar{t}} = \frac{(m_{b_1q_1q_1} - m_t)^2}{\sigma_t^2} + \frac{(m_{b_2q_2q_2} - m_t)^2}{\sigma_t^2} + \frac{(m_{q_1q_1} - m_W)^2}{\sigma_W^2} + \frac{(m_{q_2q_2} - m_W)^2}{\sigma_W^2} \tag{9}$$

The $t\bar{t}$ formulation we use is given in Equation 9. In [10], a different formulation of χ^2 was used that more closely matches recent ATLAS results in which σ_t is not used explicitly. While this formulation reduces mass sculpting of incomplete and background events, it does not perform well on events partial events with only a single reconstructable top quark. Further, it is unclear how to optimally extend this formulation to the $t\bar{t}t\bar{t}$ case. Thus, in this work we prefer the formulation that explicitly includes m_t.

The χ^2 is evaluated on $t\bar{t}H$ events as:

$$\begin{aligned}\chi^2_{t\bar{t}H} = &\frac{(m_{b_1q_1q_1} - m_t)^2}{\sigma_t^2} + \frac{(m_{b_2q_2q_2} - m_t)^2}{\sigma_t^2} \\ &+ \frac{(m_{q_1q_1} - m_W)^2}{\sigma_W^2} + \frac{(m_{q_2q_2} - m_W)^2}{\sigma_W^2} + \frac{(m_{b_0b_0} - m_H)^2}{\sigma_H^2},\end{aligned} \tag{10}$$

where we have simply added an additional term to Equation 9 for the Higgs boson, analogously to the terms used for the W-bosons. We label the jets hypothesized to be the decay products of the Higgs boson as b_0 here and find $\sigma_H = 22.3$ GeV in our dataset.

The χ^2 for $t\bar{t}t\bar{t}$ is given by the expression

$$\begin{aligned}\chi^2_{t\bar{t}t\bar{t}} = &\frac{(m_{b_1q_1q_1} - m_t)^2}{\sigma_t^2} + \frac{(m_{b_2q_2q_2} - m_t)^2}{\sigma_t^2} + \frac{(m_{b_3q_3q_3} - m_t)^2}{\sigma_t^2} + \frac{(m_{b_4q_4q_4} - t)^2}{\sigma_t^2} \\ &+ \frac{(m_{q_1q_1} - m_W)^2}{\sigma_W^2} + \frac{(m_{q_2q_2} - m_W)^2}{\sigma_W^2} + \frac{(m_{q_3q_3} - m_W)^2}{\sigma_W^2} + \frac{(m_{q_4q_4} - m_W)^2}{\sigma_W^2},\end{aligned} \tag{11}$$

where we have simply added additional, identical terms for the third and fourth top quarks and W-bosons. We find that the complexity of the 12 parton final state makes this effectively intractable and thus do not present reconstruction performance with this formulation, presenting it only as a demonstration that the CPU overhead required in this topology means permutation methods do not scale to these events.

12 Appendix E: Additional Result Tables

	N_{jets}	Event Fraction	Event Efficiency	Top Quark Efficiency
All Events	== 6	0.245	0.643	0.696
	== 7	0.282	0.601	0.667
	≥ 8	0.320	0.528	0.613
	Inclusive	**0.848**	**0.586**	**0.653**
1 Top Events	== 6	0.171	0.574	0.574
	== 7	0.176	0.562	0.562
	≥ 8	0.175	0.534	0.534
	Inclusive	**0.524**	**0.556**	**0.556**
2 Top Events	== 6	0.073	0.803	0.837
	== 7	0.105	0.667	0.754
	≥ 8	0.144	0.521	0.662
	Inclusive	**0.325**	**0.633**	**0.732**

Table 5: G-SPANET results on $t\bar{t}$ using Pythia showering.

	N_{jets}	Event Fraction	Event Efficiency	Top Quark Efficiency
All Events	== 6	0.245	0.461	0.523
	== 7	0.848	0.282	0.476
	≥ 8	0.320	0.313	0.395
	Inclusive	**0.848**	**0.387**	**0.457**
1 Top Events	== 6	0.171	0.373	0.373
	== 7	0.530	0.176	0.379
	≥ 8	0.483	0.175	0.338
	Inclusive	**0.524**	**0.363**	**0.363**
2 Top Events	== 6	0.073	0.664	0.696
	== 7	0.105	0.457	0.556
	≥ 8	0.144	0.281	0.429
	Inclusive	**0.324**	**0.426**	**0.532**

Table 6: χ^2 method results on $t\bar{t}$ using Pythia showering.

	N_{jets}	Event Fraction	Event Efficiency	Top Quark Efficiency
All Events	== 6	0.220	0.659	0.711
	== 7	0.222	0.629	0.678
	≥ 8	0.185	0.564	0.618
	Inclusive	**0.629**	**0.620**	**0.672**
1 Top Events	== 6	0.156	0.593	0.593
	== 7	0.163	0.614	0.614
	≥ 8	0.138	0.575	0.575
	Inclusive	**0.459**	**0.595**	**0.595**
2 Top Events	== 6	0.064	0.819	0.854
	== 7	0.059	0.672	0.765
	≥ 8	0.046	0.533	0.684
	Inclusive	**0.170**	**0.690**	**0.777**

Table 7: G-SPANET results on $t\bar{t}$ using Herwig showering.

	N_{jets}	Event Fraction	Event Efficiency	Top Quark Efficiency
All Events	== 6	0.220	0.505	0.560
	== 7	0.222	0.442	0.488
	≥ 8	0.185	0.338	0.386
	Inclusive	**0.629**	**0.434**	**0.484**
1 Top Events	== 6	0.156	0.434	0.434
	== 7	0.163	0.442	0.442
	≥ 8	0.138	0.363	0.363
	Inclusive	**0.459**	**0.415**	**0.415**
2 Top Events	== 6	0.064	0.678	0.713
	== 7	0.059	0.442	0.553
	≥ 8	0.046	0.263	0.419
	Inclusive	**0.170**	**0.483**	**0.577**

Table 8: χ^2 method results on $t\bar{t}$ using Herwig showering.

	N_{jets}	Event Fraction	Event Efficiency	Higgs Efficiency	Top Quark Efficiency
All Events	== 8	0.281	0.329	0.430	0.498
	== 9	0.316	0.304	0.430	0.476
	≥ 10	0.355	0.264	0.420	0.441
	Inclusive	**0.954**	**0.297**	**0.426**	**0.468**
Higgs Events	== 8	0.197	0.317	0.430	0.531
	== 9	0.227	0.295	0.430	0.504
	≥ 10	0.261	0.257	0.420	0.462
	Inclusive	**0.686**	**0.287**	**0.426**	**0.493**
1 Top Events	== 8	0.167	0.314	0.413	0.466
	== 9	0.177	0.297	0.409	0.448
	≥ 10	0.184	0.273	0.397	0.421
	Inclusive	**0.529**	**0.294**	**0.406**	**0.444**
2 Top Events	== 8	0.066	0.352	0.590	0.539
	== 9	0.092	0.295	0.540	0.504
	≥ 10	0.130	0.225	0.490	0.456
	Inclusive	**0.289**	**0.277**	**0.526**	**0.490**
Full Events	== 8	0.036	0.440	0.590	0.599
	== 9	0.057	0.344	0.540	0.542
	≥ 10	0.087	0.248	0.490	0.480
	Inclusive	**0.180**	**0.317**	**0.526**	**0.523**

Table 9: G-SPANET results on $t\bar{t}H$ with at least 2 btagged jets (all generated events).

	N_{jets}	Event Fraction	Event Efficiency	Higgs Efficiency	Top Quark Efficiency
All Events	== 8	0.260	0.370	0.497	0.540
	== 9	0.313	0.343	0.492	0.514
	≥ 10	0.397	0.294	0.472	0.473
	Inclusive	**0.972**	**0.330**	**0.485**	**0.502**
Higgs Events	== 8	0.209	0.380	0.497	0.580
	== 9	0.252	0.355	0.492	0.550
	≥ 10	0.320	0.302	0.472	0.501
	Inclusive	**0.782**	**0.340**	**0.485**	**0.535**
1 Top Events	== 8	0.153	0.335	0.479	0.494
	== 9	0.171	0.324	0.474	0.475
	≥ 10	0.199	0.296	0.448	0.446
	Inclusive	**0.524**	**0.316**	**0.466**	**0.469**
2 Top Events	== 8	0.061	0.435	0.657	0.597
	== 9	0.096	0.360	0.601	0.550
	≥ 10	0.153	0.269	0.545	0.491
	Inclusive	**0.310**	**0.330**	**0.583**	**0.530**
Full Events	== 8	0.042	0.532	0.657	0.663
	== 9	0.070	0.422	0.601	0.596
	≥ 10	0.116	0.306	0.545	0.523
	Inclusive	**0.228**	**0.383**	**0.583**	**0.572**

Table 10: G-SPANET results on $t\bar{t}H$ with at least 4 btagged jets (filtered events).

	N_{jets}	Event Fraction	Event Efficiency	Top Quark Efficiency
All Events	== 12	0.227	0.257	0.458
	== 13	0.309	0.232	0.453
	≥ 14	0.433	0.185	0.426
	Inclusive	**0.970**	**0.217**	**0.441**
1 Top Events	== 12	0.060	0.412	0.412
	== 13	0.069	0.399	0.399
	≥ 14	0.073	0.374	0.374
	Inclusive	**0.202**	**0.394**	**0.394**
2 Top Events	== 12	0.106	0.217	0.441
	== 13	0.136	0.206	0.430
	≥ 14	0.172	0.181	0.406
	Inclusive	**0.415**	**0.198**	**0.423**
3 Top Events	== 12	0.056	0.162	0.482
	== 13	0.089	0.148	0.471
	≥ 14	0.148	0.117	0.436
	Inclusive	**0.294**	**0.135**	**0.455**
4 Top Events	== 12	0.005	0.297	0.580
	== 13	0.014	0.211	0.543
	≥ 14	0.039	0.111	0.470
	Inclusive	**0.059**	**0.152**	**0.497**

Table 11: G-SPANET results on $t\bar{t}t\bar{t}$ with at least 2 btagged jets (all generated events).

	N_{jets}	Event Fraction	Event Efficiency	Top Quark Efficiency
All Events	== 12	0.219	0.276	0.484
	== 13	0.304	0.247	0.474
	≥ 14	0.450	0.198	0.450
	Inclusive	**0.974**	**0.231**	**0.464**
1 Top Events	== 12	0.055	0.422	0.422
	== 13	0.062	0.414	0.414
	≥ 14	0.0684	0.388	0.388
	Inclusive	**0.185**	**0.407**	**0.407**
2 Top Events	== 12	0.101	0.235	0.461
	== 13	0.132	0.222	0.445
	≥ 14	0.175	0.194	0.420
	Inclusive	**0.410**	**0.213**	**0.438**
3 Top Events	== 12	0.057	0.200	0.513
	== 13	0.094	0.172	0.492
	≥ 14	0.162	0.136	0.460
	Inclusive	**0.313**	**0.159**	**0.479**
4 Top Events	== 12	0.006	0.350	0.617
	== 13	0.016	0.249	0.567
	≥ 14	0.044	0.149	0.504
	Inclusive	**0.066**	**0.191**	**0.529**

Table 12: G-SPANET results on $t\bar{t}t\bar{t}$ with at least 4 btagged jets (filtered events).