Supplementary material for Variational Monte Carlo on a Budget – Fine-tuning pre-trained Neural Wavefunction

A Molecule datasets

Bicyclobutane For the Bicyclobutane to 1,3-butadiene transition we use the geometries from Kinal et al. [30] and compare against the reference energies stated in Spencer et al. [5].

N_2 For the N_2 potential energy surface with various bond-lengths we used the geometries including reference calculations from [8].

Propadiene The global rotation of 360° degrees for propadiene is performed on the geometry which is part of the test set for 3 heavy atoms. For the torsion experiment we used the equilibrium geometry and rotated the torsion angle by 90° degrees in steps of 10° degrees.

Zero-shot and fine-tuning dataset The results on zero-shot and few-shot predictions for increasing number of heavy atoms are performed on random subsets of molecules. For 5-7 heavy atoms we sample 4 unique and distorted molecules from QM7-X [26]. For 4 heavy atoms we use all geometries from the Bicyclobutane dataset. For 3 heavy atoms we use the ablation dataset.

Ablation dataset For the ablation study, we use one geometry per molecule from the out-of-distribution test set from Scherbela et al. [25], leading to a set of four distinct molecules. We ensure that these molecules are not part of the training set.

Large scale experiment For the large scale experiment we used a stratified random sample of 250 molecules from QM7 [31]. It contains all molecules with up to 4 heavy atoms, and additionally 65 randomly chosen molecules for 5, 6 and 7 heavy atoms each.

Pre-training dataset for transferable neural wavefunctions To train our pre-trained wavefunctions we use two datasets, consisting of 18 and 98 disparate molecules. For part of the ablation we use the dataset proposed in [25] and an extended version with 80 additional molecules. The additional compounds are a combination of all valid SMILES generated with RDKit [34] with 3 heavy atoms, allowing only Nitrogen, Oxygen and Carbon with single-, double- or triple-bonds, and all molecules up to four heavy atoms from QM7-X [26] (excluding molecule containing Fluorine). To prevent a train-test leakage, we remove Bicyclobutane (including all conformations) and the four molecules from the ablations dataset. Since the normal-mode-distortions by design do not generate strongly distorted geometries, we augment the 98-molecule-dataset with rotated dihedral angles. To generate a subset of all possible dihedral angles for a heavy-atom bond we first generate samples with equidistant angles for all possible dihedral angles and compute Hartree-Fock energies with a minimal basis-set. We include the equilibrium geometry and all extrema of the potential energy surface with respect to the rotation of a single dihedral angle if the energy of the extrema is significantly different to already included geometries of the same molecule. Additionally, we include the transition geometry towards the respective extrema and again only include energetic diverse states. Finally, to make sure that certain molecules are not underrepresented in the dataset we make sure that all molecules have at least 5 geometries that get distorted during pre-training by adding copies of the equilibrium geometry. Overall this yields 699 initial geometries $\boldsymbol{R}^0$ for pre-training.

B Electron MCMC initialization

To investigate the impact of the initial distribution of electron positions on the equilibration of the Markov Chain, we run two evaluations for a glycine molecule, using a pre-trained wavefunction. We perform no initial burn-in and use every 50th sample for energy evaluation. If the chain was perfectly equilibrated right after initialization, all sampled energies would fluctuate around the mean energy. However as Fig. 6b shows, it takes several thousand steps for the sampled energies to converge to the correct mean. This is particularly pronounced with Gaussian initialization of electron positions,

which is the default in state-of-the-art DL-VMC codes such as FermiNet [3]. Using an exponential distribution of the initial electron positions much more closely resembles the correct electron density ψ^2 (cf. Fig. 6a) and thus reaches equilibrium substantially faster.

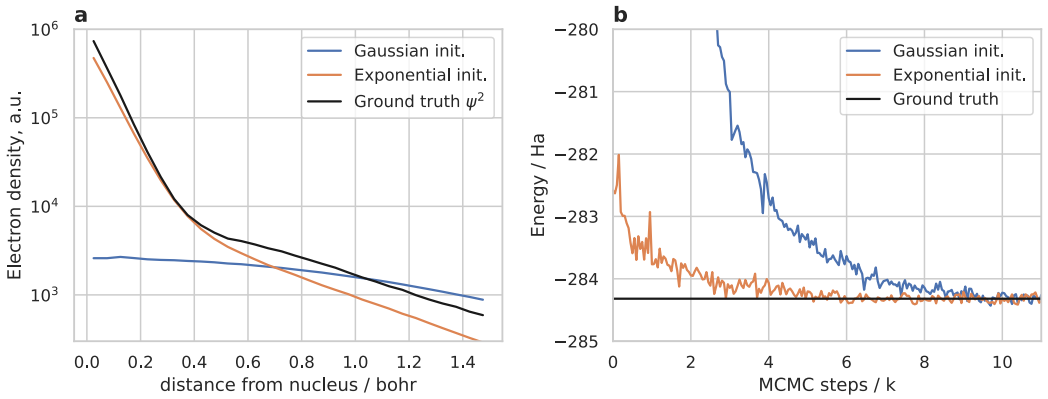

Figure 6: **Effect of electron initialization**: Initializing the electron positions using an exponential distribution instead of Gaussian, better fits the actual density (a), and thus leads to faster equilibration of observables during evaluation (b).

C Orbital localization

Our model uses orbital embeddings $\boldsymbol{x}^{\text{orb}}$ as inputs to parameterize the backflows $\boldsymbol{f}^{\text{orb}}$, and exponents $\boldsymbol{g}^{\text{orb}}$ of the orbitals. These orbital embeddings were introduced by Scherbela et al. [25] in the form of molecular orbital expansion coefficients, obtained from a self consistent Hartree-Fock calculation. In this setting, the coefficients $\boldsymbol{x}^{\text{orb}}$ are not uniquely defined, but only up to a linear transformation U with determinant ± 1

$$\boldsymbol{x}^{\text{orb}}_{Ik} = \sum_{n=1}^{N_{\text{orb}}} U_{kn} \hat{\boldsymbol{x}}^{\text{orb}}_{Ik}, \qquad U \in \mathbb{R}^{N_{\text{orb}} \times N_{\text{orb}}}, \qquad \det U = \pm 1. \tag{15}$$

This stems from the fact that the corresponding Hartree-Fock wavefunction is invariant under such a transformation. Consequently there is free choice, which linear combination of embeddings $\boldsymbol{x}^{\text{orb}}$ to choose from without any loss of information. We follow the approach of [25], by choosing U such that the corresponding Hartree-Fock orbitals are maximally localized according to the Foster-Boys metric, i.e. minimize the spatial variance $\mathcal{L}$:

$$\phi_k(\boldsymbol{r}, U) = \sum_{I=1}^{N_{\text{nuc}}} \sum_{\mu=1}^{N_{\text{basis}}} b_{I\mu}(\boldsymbol{r}) U_{kn} \hat{x}^{\text{orb}}_{In\mu} \tag{16}$$

$$\mathcal{L}(U) = \sum_k \int \phi_k^2(\boldsymbol{r}, U) \boldsymbol{r}^2 d\boldsymbol{r} - \left(\int \phi_k^2(\boldsymbol{r}, U) \boldsymbol{r} d\boldsymbol{r} \right)^2 \tag{17}$$

Here $b_{I\mu}(\boldsymbol{r})$ denotes μ-th basis function of the Hartree-Fock expansion, centered on the I-th nucleus. In practice the integrals of Eq. 17 do not have to be evaluated explicitly, but can instead be computed via the overlap matrix $\boldsymbol{S}$. The minimization of $\mathcal{L}$ is typically done iteratively, requires on the order of 10 steps, and is readily implemented in many open-source quantum chemistry codes such as pySCF [35].

D Tables of energies

D.1 Conformers of Bicyclobutane

In Tab. 1 we list the relative energies of our method and all reference energies corresponding to Fig. 3.

Table 1: Energies relative to the energy of bicyclobutane in mHa, including the zero-point vibrational energy correction from Kinal et al. [30]

structure	CCSD(T) [30]	DMC [30]	FermiNet 200k [5]	FermiNet 10k [5]	Our work zero-shot	Our work 700 per geom
con_TS	64.4	64.4	64.1	63.9	94.0	66.6
dis_TS	34.7	93.4	92.0	87.1	183.8	94.5
g-but	-40.0	-40.2	-40.3	-44.9	-48.5	-40.4
gt-TS	-35.5	-35.4	-35.9	-42.9	-46.9	-36.7
t-but	-44.6	-44.5	-45.3	-47.5	-51.8	-43.2

E Reference energies

CCSD(T) All CCSD(T) energies – except explicitly stated otherwise – were obtained using ORCA [36] starting from a restricted Hartree-Fock calculation. We use correlation consistent basis sets of the cc-pCVXZ family, with X in {2, 3, 4}. To extrapolate to the complete basis set limit (CBS), we use the approach outlined in [3] and fit the following functions with free parameters $E^{\text{HF}}_{\text{CBS}}, E^{\text{corr}}_{\text{CBS}}, a, b, c$:

$$
\begin{aligned}
E^{\text{HF}}_X &= E^{\text{HF}}_{\text{CBS}} + ae^{-bX} \\
E^{\text{corr}}_X &:= E^{\text{HF}}_X - E^{\text{CCSD(T)}}_X = E^{\text{corr}}_{\text{CBS}} + cX^{-3} \\
E^{\text{CCSD(T)}}_{\text{CBS}} &= E^{\text{HF}}_{\text{CBS}} + E^{\text{corr}}_{\text{CBS}}
\end{aligned}
$$

We stress that although CCSD(T)-energies are often considered as "gold-standard", they do not necessarily represent the actual ground-state energy. There are many cases, where CCSD(T) either overestimates the true ground-state energy, or even underestimates it, because CCSD(T) does not yield upper bounds to the true ground-state energy.

PsiFormer For Fig. 2 we used the open-source FermiNet codebase [37]. The codebase didn't allow for inference calculation, therefore a slight fix was applied. All calculations were performed with the small settings as proposed in von Glehn et al. [6].

F Adaption of PhisNet

We heavily rely on PhisNet by Unke et al. [20] to obtain orbital descriptors without the need for a separate SCF calculation. Compared to their original work, we made several simplifications, which are motivated by the fact that we do not predict final high-accuracy orbitals in a large basis set, but only use PhisNet as a feature extractor by predicting orbitals in a minimal basis-set:

- **Layer Norm** We found deep variants of PhisNet to be unstable to train and mitigated the issue by adding an (equivariant) layer norm after each PhisNet module.
- **Simplified Fock matrix prediction** The original PhisNet implementation uses a final interaction between the node embeddings, before predicting the elements of the Fock matrix. We found this interaction to be superfluous for our purposes and left it out for simplicity.
- **Separate energy head** The original PhisNet computes energies via the eigenvalues obtained by diagonalization of the Fock matrix. We instead predict energies using a separate head on top of the scalar features of the node embeddings.
- **Smaller network** We changed the hyperparameters to obtain a smaller and faster version of PhisNet which obtained sufficient accuracy for our purposes. We used 2 layers (instead of 5) and $L_{\text{max}} = 2$ (instead of 4). This reduces the number of parameters from 17M to 3M.
- **Diverse training set** While the original work optimized separate models for each molecule (e.g. by training on different geometries of a molecular dynamics simulation), we optimize a single model to predict $\boldsymbol{F}$, $\boldsymbol{S}$, $\boldsymbol{E}$, and $\boldsymbol{\nabla E}$ across a dataset of 47k geometries sampled from QM7-X [26].
- **JAX re-implementation** We re-implemented PhisNet in JAX, using the e3nn library [38] to construct the SE(3)-equivariant operations.

We train the PhisNet-model on a dataset of 47k molecules from QM7X [26], using the Adam optimizer [39] on the following loss

$$\mathcal{L} = \sum_n \left(E^{\text{phis}}(\boldsymbol{R}^n, \boldsymbol{Z}^n) - E^{\text{ref},n}\right)^2 + \tag{18}$$

$$+ \sum_{nI\zeta} \left(\frac{\partial}{\partial R^n_{I\zeta}} E^{\text{phis}}(\boldsymbol{R}^n, \boldsymbol{Z}^n) - G^{\text{ref},n}_{I\zeta}\right)^2 + \tag{19}$$

$$+ \sum_{nIJ\mu\nu} \left(F^{\text{phis}}_{IJ\mu\nu}(\boldsymbol{R}^n, \boldsymbol{Z}^n) - F^{\text{ref},n}_{IJ\mu\nu}\right)^2 + \tag{20}$$

$$+ \sum_{nIJ\mu\nu} \left(S^{\text{phis}}_{IJ\mu\nu}(\boldsymbol{R}^n, \boldsymbol{Z}^n) - S^{\text{ref},n}_{IJ\mu\nu}\right)^2 . \tag{21}$$

Here E denotes energies, G denotes gradients of energies, F Fock matrices, and S overlap matrices. The indices I, J run over nuclei, the indices μ, ν over basis functions, and the index n over samples in a batch.

G Hyperparameters

A detailed description of the hyperparameter used in this work can be found below (cf. Tab. 2). For the mapping of the orbital descriptors to the electron embeddings to build the orbitals we rely on the hyperparameter from [25]. For optimization we rely on the second-order method KFAC [29] and use their Python implementation [40]. During the continuous sampling of the geometries we allow each geometry to perform a maximum of 20 steps of normal-mode distortion from the initial geometry and reset to the original one once the threshold is reached.

H Computational resources

We used $\approx$ 5k GPUhs (A100) for development and training of our base models, and another 5k GPUhs (A40) on evaluations and fine-tuning. Additionally we required $\approx$ 20k CPUhs for CCSD(T) reference calculations.

I Code and data availability

All code, configuration files, geometries, datasets and obtained energies are available in the supplementary information.

Table 2: Hyperparameter settings used in this work

Electron Embedding	Hidden dimension N_{emb}	256
	№ iterations	4
Nuclear Embedding	Hidden dimension $\tilde{\boldsymbol{x}}^{nuc}$	64
	№ layer MLP	1
Message passing	Activation function	SiLU
	№ layer edge embedding	3
	Dimension edge embedding	64
	Dimension linear layer	32
Markov Chain Monte Carlo	№ walkers	2048
	№ decorrelation steps	50
	Target acceptance prob.	50%
PhisNet [20]	Pre-trained against basis set	STO-6G
	№ iterations	2
	Harmonic degree L	2
	№ radial basis functions	128
	Hidden dimension of $\boldsymbol{x}^{nuc}$	128
	Distance cutoff (bohr)	30
Transferable atomic orbitals [25]	№ determinants N_{det}	8
	№ hidden layers $\boldsymbol{f}^{\text{orb}}$	2
	Hidden dimension of $\boldsymbol{f}^{\text{orb}}$	256
	№ hidden layers $\boldsymbol{g}^{\text{orb}}$	2
	Hidden dimension $\boldsymbol{g}^{\text{orb}}$	128
	№ iterations MPNN	2
	№ radial basis functions	16
	Hidden edge embedding dimension	32
	Hidden node embedding dimension	16
	Activation function	SiLU
Variational pre-training	Optimizer	KFAC
	Batch size	2048
	Norm constraint	3×10^{-3}
	Initial damping d_0	1
	Minimal damping d_{min}	0.001
	Damping rate decay	$\text{d}(t) = \text{d}_0 \exp(-t/20000)$
	Initial learning rate lr_0	0.1
	Learning rate decay	$\text{lr}(t) = \text{lr}_0(1 + t/6000)^{-1}$
	Optimization steps	128,000 - 256,000
Changes for fine-tuning	Learning rate decay	$\text{lr}(t) = \text{lr}_0(7 + t/6000)^{-1}$
	Optimization steps	0 - 32,000
Sampling geometries	Distortion energy β	0.005 Ha
	Max age	20
	Bias towards original geometry α	0.2

References

[1] M. Born and R. Oppenheimer. "Zur Quantentheorie der Molekeln". In: *Annalen der Physik* 389.20 (1927), pp. 457–484. DOI: https://doi.org/10.1002/andp.19273892002.

[2] László Gyevi-Nagy, Mihály Kállay, and Péter R. Nagy. "Accurate Reduced-Cost CCSD(T) Energies: Parallel Implementation, Benchmarks, and Large-Scale Applications". In: *Journal of Chemical Theory and Computation* 17.2 (Feb. 2021), pp. 860–878. DOI: 10.1021/acs.jctc.0c01077.

[3] David Pfau et al. "Ab Initio Solution of the Many-Electron Schrödinger Equation with Deep Neural Networks". In: *Phys. Rev. Res.* 2.3 (Sept. 2020), p. 033429. DOI: 10.1103/PhysRevResearch.2.033429.

[4] W. K. Hastings. "Monte Carlo Sampling Methods Using Markov Chains and Their Applications". In: *Biometrika* 57.1 (Apr. 1970), pp. 97–109. DOI: 10.1093/biomet/57.1.97.

[5] James S. Spencer et al. "Better, Faster Fermionic Neural Networks". In: (2020).

[6] Ingrid von Glehn, James S. Spencer, and David Pfau. *A Self-Attention Ansatz for Ab-initio Quantum Chemistry*. Nov. 2022. DOI: 10.48550/arXiv.2211.13672.

[7] Weiluo Ren et al. "Towards the ground state of molecules via diffusion Monte Carlo on neural networks". In: *Nature Communications* 14.1 (Apr. 2023), p. 1860. DOI: 10.1038/s41467-023-37609-3.

[8] Leon Gerard et al. "Gold-Standard Solutions to the Schrödinger Equation Using Deep Learning: How Much Physics Do We Need?" In: *Advances in Neural Information Processing Systems*. Oct. 2022.

[9] M. T. Entwistle et al. "Electronic excited states in deep variational Monte Carlo". In: *Nature Communications* 14.1 (Jan. 2023), p. 274. DOI: 10.1038/s41467-022-35534-5.

[10] Michael Scherbela et al. "Solving the Electronic Schrödinger Equation for Multiple Nuclear Geometries with Weight-Sharing Deep Neural Networks". In: *Nature Computational Science* 2.5 (May 2022), pp. 331–341. DOI: 10.1038/s43588-022-00228-x.

[11] Yubing Qian et al. "Interatomic Force from Neural Network Based Variational Quantum Monte Carlo". In: *The Journal of Chemical Physics* 157.16 (Oct. 2022), p. 164104. DOI: 10.1063/5.0112344.

[12] Gino Cassella et al. "Discovering Quantum Phase Transitions with Fermionic Neural Networks". In: *Physical Review Letters* 130.3 (Jan. 2023), p. 036401. DOI: 10.1103/PhysRevLett.130.036401.

[13] Xiang Li, Zhe Li, and Ji Chen. "Ab initio calculation of real solids via neural network ansatz". In: *Nature Communications* 13.1 (Dec. 2022), p. 7895. DOI: 10.1038/s41467-022-35627-1.

[14] Giuseppe Carleo and Matthias Troyer. "Solving the Quantum Many-Body Problem with Artificial Neural Networks". In: *Science* 355.6325 (Feb. 2017), pp. 602–606. DOI: 10.1126/science.aag2302.

[15] Giuseppe Carleo et al. "NetKet: A Machine Learning Toolkit for Many-Body Quantum Systems". In: *SoftwareX* 10 (July 2019), p. 100311. DOI: 10.1016/j.softx.2019.100311.

[16] Yuan-Hang Zhang and Massimiliano Di Ventra. "Transformer Quantum State: A Multipurpose Model for Quantum Many-Body Problems". In: *Physical Review B* 107.7 (Feb. 2023), p. 075147. DOI: 10.1103/PhysRevB.107.075147.

[17] Nicholas Gao and Stephan Günnemann. *Generalizing Neural Wave Functions*. Feb. 2023. DOI: 10.48550/arXiv.2302.04168.

[18] Simon Batzner et al. "E(3)-Equivariant Graph Neural Networks for Data-Efficient and Accurate Interatomic Potentials". In: *Nature Communications* 13.1 (May 2022), p. 2453. DOI: 10.1038/s41467-022-29939-5.

[19] Ilyes Batatia et al. "MACE: Higher Order Equivariant Message Passing Neural Networks for Fast and Accurate Force Fields". In: *Advances in Neural Information Processing Systems*. 2022.

[20] Oliver Unke et al. "SE(3)-Equivariant Prediction of Molecular Wavefunctions and Electronic Densities". In: *Advances in Neural Information Processing Systems*. 2021.

[21] M. Gastegger et al. "A Deep Neural Network for Molecular Wave Functions in Quasi-Atomic Minimal Basis Representation". In: *The Journal of Chemical Physics* 153.4 (July 2020), p. 044123. DOI: 10.1063/5.0012911.

[22] Albert Musaelian et al. “Scaling the Leading Accuracy of Deep Equivariant Models to Biomolecular Simulations of Realistic Size”. In: (Apr. 2023). DOI: 10.48550/arXiv.2304.10061. arXiv: 2304.10061.

[23] Nicholas Gao and Stephan Günnemann. “Ab-Initio Potential Energy Surfaces by Pairing GNNs with Neural Wave Functions”. In: *arXiv:2110.05064 [physics]* (Nov. 2021). arXiv: 2110.05064 [physics].

[24] Nicholas Gao and Stephan Günnemann. “Sampling-free Inference for Ab-Initio Potential Energy Surface Networks”. In: (2022). DOI: 10.48550/ARXIV.2205.14962.

[25] Michael Scherbela, Leon Gerard, and Philipp Grohs. “Towards a Foundation Model for Neural Network Wavefunctions”. In: *arXiv.org* (Mar. 2023).

[26] Johannes Hoja et al. “QM7-X, a comprehensive dataset of quantum-mechanical properties spanning the chemical space of small organic molecules”. In: *Scientific Data* 8.1 (Feb. 2021), p. 43. DOI: 10.1038/s41597-021-00812-2.

[27] János Pipek and Paul G. Mezey. “A Fast Intrinsic Localization Procedure Applicable for Ab Initio and Semiempirical Linear Combination of Atomic Orbital Wave Functions”. In: *The Journal of Chemical Physics* 90.9 (May 1989), pp. 4916–4926. DOI: 10.1063/1.456588.

[28] Claudia Filippi and C. J. Umrigar. “Correlated Sampling in Quantum Monte Carlo: A Route to Forces”. In: *Physical Review B* 61.24 (June 2000), R16291–R16294. DOI: 10.1103/PhysRevB.61.R16291.

[29] James Martens and Roger Grosse. “Optimizing neural networks with kronecker-factored approximate curvature”. In: *International conference on machine learning*. PMLR. 2015, pp. 2408–2417.

[30] Armağan Kinal and Piotr Piecuch. “Computational Investigation of the Conrotatory and Disrotatory Isomerization Channels of Bicyclo[1.1.0]butane to Buta-1,3-diene: A Completely Renormalized Coupled-Cluster Study”. In: *The Journal of Physical Chemistry A* 111.4 (Feb. 2007). Publisher: American Chemical Society, pp. 734–742. ISSN: 1089-5639. DOI: 10.1021/jp065721k. URL: https://doi.org/10.1021/jp065721k.

[31] Matthias Rupp et al. “Fast and Accurate Modeling of Molecular Atomization Energies with Machine Learning”. In: *Phys. Rev. Lett.* 108 (5 Jan. 2012), p. 058301. DOI: 10.1103/PhysRevLett.108.058301.

[32] Kristof T. Schütt et al. “SchNet: A Continuous-Filter Convolutional Neural Network for Modeling Quantum Interactions”. In: (June 2017). DOI: 10.48550/arXiv.1706.08566.

[33] Gabriel Pescia et al. *Message-Passing Neural Quantum States for the Homogeneous Electron Gas*. May 2023. DOI: 10.48550/arXiv.2305.07240.

[34] Greg Landrum. *RDKit: Open-source cheminformatics*. https://github.com/rdkit/rdkit. GitHub repository. 2009.

[35] Qiming Sun et al. “Recent Developments in the PySCF Program Package”. In: *The Journal of Chemical Physics* 153.2 (July 2020), p. 024109. DOI: 10.1063/5.0006074.

[36] Frank Neese et al. “The ORCA Quantum Chemistry Program Package”. In: *The Journal of Chemical Physics* 152.22 (June 2020), p. 224108. DOI: 10.1063/5.0004608.

[37] David Pfau James S. Spencer and FermiNet Contributors. *FermiNet*. 2020. URL: http://github.com/deepmind/ferminet.

[38] Mario Geiger and Tess Smidt. *E3nn: Euclidean Neural Networks*. July 2022. arXiv: 2207.09453 [cs].

[39] Diederik P. Kingma and Jimmy Ba. “Adam: A Method for Stochastic Optimization”. In: arXiv:1412.6980 (Jan. 2017). DOI: 10.48550/arXiv.1412.6980. arXiv: 1412.6980 [cs]. (Visited on 05/17/2023).

[40] Aleksandar Botev and James Martens. *KFAC-JAX*. Version 0.0.1. 2022. URL: http://github.com/deepmind/kfac-jax.