dynLattice: A finite element environment for dynamic simulation of beam networks and lattice metamaterials

Til Gärtner[1,2,3] **and Frans P. van der Meer**[1]

1 Delft University of Technology, The Netherlands **2** Netherlands Organisation for Applied Scientific Research, The Netherlands **3** Leibniz University Hannover, Germany

DOI: 10.xxxxxx/draft

Software

- Review
- Repository
- Archive

Editor: Mojtaba Barzegari

Submitted: 15 September 2025
Published: unpublished

Summary

Beams and beam structures undergoing fast deformation often experience large strains and inelastic material behaviour. Understanding this behaviour enables engineers and designers to design beam networks resulting in so-called mechanical metamaterials. Metamaterials offer unique properties not found in naturally occurring materials. Harnessing these properties presents new possibilities for a wide range of fields (Davies et al., 2025; Jiao et al., 2023). In order to enable engineers to design mechanical metamaterials, the relation between the geometry of the beam network and the resulting properties needs to be understood. For large deformations at high rates, this usually requires destructive testing, limiting the speed of developments. This limitation can be overcome using efficient numerical tools, allowing for both the accurate and fast description of the inelastic deformation at high rates (Bonfanti et al., 2024).

Background

The following is a short description of the beam kinematics and cross-sectional kinetics implemented in this software, with a more detailed description given in (Gärtner, 2025).

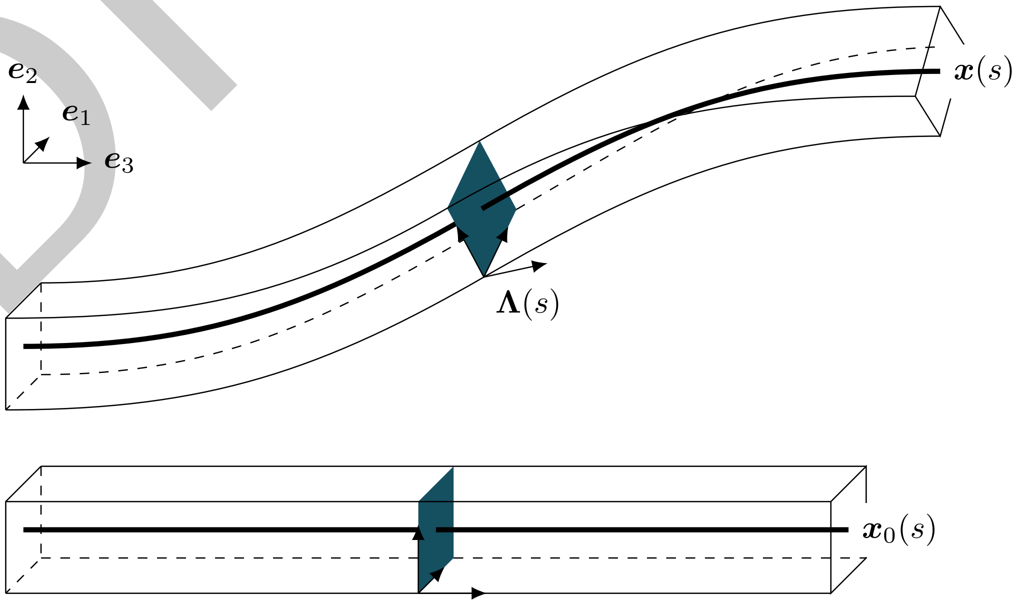

Figure 1: A beam undergoing large deformation.

In Figure 1, a beam undergoing large deformation is shown. A beam is described with the centerline $\mathbf{x}(s)$, where $s \in [s_0, s_1]$ is the measure along the length of the beam, with two orthonormal directors attached to it $\mathbf{d}_1(s), \mathbf{d}_2(s)$. Together with a third orthonormal director $\mathbf{d}_3(s) = \mathbf{d}_1(s) \times \mathbf{d}_2(s)$, these are the column vectors of the rotation matrix between the global reference frame and the local coordinate frame of the beam:

$$\Lambda(s) = \mathbf{d}_i(s) \otimes \mathbf{e}_i = [\mathbf{d}_1(s) \quad \mathbf{d}_2(s) \quad \mathbf{d}_3(s)] \,.$$

From these measures, strain prescriptors, describing stretching, shearing, bending, and twisting of the beam can be derived. These strain prescriptors are then used together with the material model of the beam, be it elastic (cf. (Eugster, 2015; Simo & Vu-Quoc, 1986)) or elasto-plastic (cf. (Gärtner, van den Boom, et al., 2025a; Herrnböck et al., 2022; Smriti et al., 2021)), to assemble the global force vector and—in the static, implicit case—the tangent stiffness matrix using standard finite element procedures.

Statement of need

Research on the dynamic behaviour of mechanical metamaterials, especially beam lattices, relies on numerical tools that can capture large deformations, inelastic behaviour, and contact while remaining efficient and flexible enough for parameter studies (Bonfanti et al., 2024). General-purpose finite element frameworks, e.g. (Dadvand et al., 2010; Ferrándiz et al., 2025; Richart et al., 2024), provide broad capabilities but often lack a complete combination of non-linear Timoshenko-Ehrenfest beam formulations, robust explicit time integration, and inelastic constitutive behaviour. Here, a specialized lattice framework, e.g. (McDonnell & Ning, 2022), typically delivers high-fidelity beam kinematics but only focuses on elastic deformations or omits contact between the constituting beams of the lattice. Resulting from this, many researchers rely on their own implementation of Simo-Reissner beams, which are usually not available to the community and only consider a subset of the requirements needed for a general toolkit. An open and extensible environment that unifies geometrically exact beams, explicit dynamics, inelasticity, and contact is therefore central to advancing simulation-driven design of lattice metamaterials for high-rate and large deformation applications. Here, dynLattice takes a first step by building upon the `JIVE` framework (Dynaflow Research Group, 2021; Nguyen-Thanh et al., 2020), keeping the extensible object-oriented design philosophy, and providing extensive documentation in the style of the underlying framework.

State of the field

There is a wide plethora of existing finite element toolkits available in the market, such as MOOSE (Lindsay et al., 2022), FEniCS (Baratta et al., 2023), or deal.II (Arndt et al., 2023), to name several prominent ones. The usage of these libraries for explicit dynamics with inelastic beams however, would require substantial alteration of their code, as for example in FEniCS explicit dynamics is only available using lumped mass matrices, which would be non-physical for the rotational inertia contributions of the beams, in deal.II neither beam elements nor explicit solves are available and in MOOSE only a limited Timoshenko-Ehrenfest beam is implemented with only elastic behaviour. Other finite element toolkits having built in beam-elements and explicit dynamics include Kratos (Dadvand et al., 2010; Ferrándiz et al., 2025) and Akantu (Richart et al., 2024). For Kratos similar problems as for MOOSE can be seen in only a limited set of beam elements being available and none supporting inelastic behaviour in the cross-sectional strain measures. Akantu, focusing on fracture mechanics and providing a solid representation of contact, on the other hand only provides Euler-Bernoulli beam elements, neglecting all shear deformation within the elements. Next to general finite element toolkits, there are specialized beam toolkits, most prominently for shear-deformable beams, GXBeam (McDonnell & Ning, 2022). Here, the same beam theory employed in this contribution is used, however due to its focus on composite beams, it is limited to (anisotropic) elastic material behaviour and does not model contact.

Software design

In order to design metamaterials undergoing large deformations at high rates, dynLattice provides both researchers and engineers in the field a toolkit for numerical experiments. dynLattice is built on top of the openly available C++ JIVE finite element toolkit (Dynaflow Research Group, 2021; Nguyen-Thanh et al., 2020) and follows in its design the object-oriented framework of JIVE to describe the partial differential equations and corresponding solution procedures. The toolkit allows for the implementation of original methods through the development of custom *modules*, for parts of the program flow in the simulation framework, and of custom *models*, for anything related to the numerical representation of the system of partial differential equations. In the present addition to the JIVE framework, the *models* further contain a separate *materials* class, representing the constitutive models considered in the problem, separating geometric from material considerations. This separation allows both the easy and user-friendly extension of the frameworks by simple inheritance of the provided classes as well as usability by structuring the inputs needed for the program flow and the problem description. dynLattice additionally uses GMSH (Geuzaine & Remacle, 2009) to mesh the imported geometries efficiently.

Research impact statement

dynLattice has enabled a series of research efforts investigating lattice metamaterials Gärtner (2025). In addition to investigating lattice materials, it has been used to study the behaviour of beams as a multi-scale problem (Gärtner, van den Boom, et al., 2025a) and has undergone a thorough comparison with experimental data and results from commercial tools (Gärtner, Dekker, et al., 2025). Collaborating researchers could easily install and run the software using the provided documentation and `apptainer`.

In addition to external collaboration, dynLattice was also used as a ready-made software package by multiple student theses (IJzendoorn, 2024; Niessen, 2022; Smit, 2024). During these projects, dynLattice was used on a large cluster ((DHPC), 2024) and on various local machines using the provided `apptainer`. This demonstrates its availability and usability as stand-alone software without the need for further modifications.

Changes to the source code that enable further research are made possible by the object-oriented philosophy inherited from the JIVE package and the clear structure of the code into *modules*, *models* and *materials*, as described in Software design. The provided documentation, available in the source and on the project website, facilitates quick and straightforward adoption by third parties. The included benchmarks serve as a verification tool for the correctness of the implementation and as a guide for creating and extending simple problems.

AI Usage Disclosure

During the creation of this software, GitHub Copilot (using different versions of Claude Sonnet) was used to enhance the documentation of the source code. In particular, it helped streamline all documentation efforts into Doxygen compatible comments. All generated documentation strings were read and approved by the author.

Acknowledgements

The research of Til Gärtner is financed by TNO through the PhD program of the Dutch Ministry of Defence.

The Journal of Open Source Software

References

Arndt, D., Bangerth, W., Bergbauer, M., Feder, M., Fehling, M., Heinz, J., Heister, T., Heltai, L., Kronbichler, M., Maier, M., Munch, P., Pelteret, J.-P., Turcksin, B., Wells, D., & Zampini, S. (2023). The deal.II library, version 9.5. *J. Numer. Math.*, *31*(3), 231–246. https://doi.org/10.1515/jnma-2023-0089

Baratta, I. A., Dean, J. P., Dokken, J. S., Habera, M., Hale, J. S., Richardson, C. N., Rognes, M. E., Scroggs, M. W., Sime, N., & Wells, G. N. (2023). *DOLFINx: The next generation FEniCS problem solving environment*. Zenodo. https://doi.org/10.5281/zenodo.10447666

Bonfanti, S., Hiemer, S., Zulkarnain, R., Guerra, R., Zaiser, M., & Zapperi, S. (2024). Computational design of mechanical metamaterials. *Nat. Comput. Sci.*, *4*(8), 574–583. https://doi.org/10.1038/s43588-024-00672-x

Dadvand, P., Rossi, R., & Oñate, E. (2010). An object-oriented environment for developing finite element codes for multi-disciplinary applications. *Arch. Comput. Methods Eng.*, *17*(3), 253–297. https://doi.org/10.1007/s11831-010-9045-2

Davies, B., Szyniszewski, S., Dias, M. A., Waal, L. de, Kisil, A., P Smyshlyaev, V., Cooper, S., Kamotski, I. V., Touboul, M., Craster, R. V., Capers, J. R., Horsley, S. A. R., Hewson, R. W., Santer, M., Murphy, R., Thillaithevan, D., Berry, S. J., Conduit, G. J., Earnshaw, J., … Zadpoor, A. (2025). Roadmap on metamaterial theory, modelling and design. *J. Phys. D: Appl. Phys.*, *58*(20). https://doi.org/10.1088/1361-6463/adc271

(DHPC), Delft High Performance Computing Centre. (2024). *DelftBlue Supercomputer (Phase 2)*. https://www.tudelft.nl/dhpc/ark:/44463/DelftBluePhase2.

Dynaflow Research Group. (2021). *JIVE* (Version 3.0). https://dynaflow.com/software/jive/

Eugster, S. R. (2015). *Geometric continuum mechanics and induced beam theories*. Springer. https://doi.org/10.1007/978-3-319-16495-3

Ferrándiz, V. M., Bucher, P., Zorrilla, R., Warnakulasuriya, S., Cornejo, A., Rossi, R., Roig, C., jcotela, Maria, J., tteschemacher, Masó, M., Casas, G., Núñez, M., Dadvand, P., Latorre, S., Pouplana, I. D., González, J. I., AFranci, Arrufat, F., … jgonzalezusua. (2025). *KratosMultiphysics/kratos: v10.2.3* (Version v10.2.3). Zenodo. https://doi.org/10.5281/zenodo.15687676

Gärtner, T. (2025). *Are auxetics better for protection? On the behaviour of architected metamaterials under high-rate loading conditions* [PhD Thesis, Technische Universiteit Delft]. https://doi.org/10.4233/uuid:9d0b230e-1008-46a2-a193-b9dd3f8c472f

Gärtner, T., Dekker, R., van Veen, D., van den Boom, S. J., & Amaral, L. (2025). (In)efficacy of auxetic metamaterials for impact mitigation. *Int. J. Impact Eng.*, *206*. https://doi.org/10.1016/j.ijimpeng.2025.105402

Gärtner, T., van den Boom, S. J., Weerheijm, J., & Sluys, L. J. (2024). Geometric effects on impact mitigation in architected auxetic metamaterials. *Mech. Mater.*, *191*. https://doi.org/10.1016/j.mechmat.2024.104952

Gärtner, T., van den Boom, S. J., Weerheijm, J., & Sluys, L. J. (2025a). A strategy for scaling the hardening behavior in finite element modelling of geometrically exact beams. *Comput. Mech.*, *75*, 1471–1482. https://doi.org/10.1007/s00466-024-02572-3

Gärtner, T., van den Boom, S. J., Weerheijm, J., & Sluys, L. J. (2025b). Force transmission and dissipation in dynamic compression of architected metamaterials. *Materials Today Advances*, *28*. https://doi.org/10.1016/j.mtadv.2025.100656

Geuzaine, C., & Remacle, J.-F. (2009). Gmsh: A 3-D finite element mesh generator with built-in pre- and post-processing facilities. *Int. J. Numer. Methods Eng.*, *79*(11), 1309–1331. https://doi.org/10.1002/nme.2579

Herrnböck, L., Kumar, A., & Steinmann, P. (2022). Two-scale off-and online approaches to geometrically exact elastoplastic rods. *Comput. Mech.*, *71*(1), 1–24. https://doi.org/10.1007/s00466-022-02204-8

IJzendoorn, P. van. (2024). *Multiscale modelling of lattice materials: A novel approach using beam neural networks* [MSc Thesis, Technische Universiteit Delft]. http://resolver.tudelft.nl/uuid:682795a0-108e-4f8f-a084-884445081d38

Jiao, P., Mueller, J., Raney, J. R., Zheng, X. R., & Alavi, A. H. (2023). Mechanical metamaterials and beyond. *Nat. Commun.*, *14*(1). https://doi.org/10.1038/s41467-023-41679-8

Lindsay, A. D., Gaston, D. R., Permann, C. J., Miller, J. M., Andrš, D., Slaughter, A. E., Kong, F., Hansel, J., Carlsen, R. W., Icenhour, C., Harbour, L., Giudicelli, G. L., Stogner, R. H., German, P., Badger, J., Biswas, S., Chapuis, L., Green, C., Hales, J., … Wong, C. (2022). 2.0 - MOOSE: Enabling massively parallel multiphysics simulation. *SoftwareX*, *20*. https://doi.org/10.1016/j.softx.2022.101202

McDonnell, T., & Ning, A. (2022). GXBeam: A pure julia implementation of geometrically exact beam theory. *Journal of Open Source Software*, *7*(73), 3997. https://doi.org/10.21105/joss.03997

Nguyen-Thanh, C., Nguyen, V. P., de Vaucorbeil, A., Kanti Mandal, T., & Wu, J.-Y. (2020). Jive: An open source, research-oriented C++ library for solving partial differential equations. *Adv. Eng. Softw.*, *150*. https://doi.org/10.1016/j.advengsoft.2020.102925

Niessen, A. (2022). *GNNs and beam dynamics: Investigation into the application of graph neural networks to predict the dynamic behaviour of lattice beams* [MSc Thesis, Technische Universiteit Delft]. http://resolver.tudelft.nl/uuid:b1cad9d4-48e9-4e6e-8760-5aabc3d8bd9f

Richart, N., Anciaux, G., Gallyamov, E., Frérot, L., Kammer, D., Pundir, M., Vocialta, M., Ramos, A. C., Corrado, M., Müller, P., Barras, F., Zhang, S., Ferry, R., Durussel, S., & Molinari, J.-F. (2024). Akantu: An HPC finite-element library for contact and dynamic fracture simulations. *J. Open Source Softw.*, *9*(94). https://doi.org/10.21105/joss.05253

Simo, J. C., & Vu-Quoc, L. (1986). A three-dimensional finite-strain rod model. Part II: Computational aspects. *Comput. Methods Appl. Mech. Eng.*, *58*(1), 79–116. https://doi.org/10.1016/0045-7825(86)90079-4

Smit, J. (2024). *Stiffness of architected materials* [BSc Thesis]. Technische Universiteit Delft.

Smriti, Kumar, A., & Steinmann, P. (2021). A finite element formulation for a direct approach to elastoplasticity in special cosserat rods. *Int. J. Numer. Methods Eng.*, *122*(3), 657–692. https://doi.org/10.1002/nme.6566