High-performance large-scale simulation of multi-stable metastructures

Myungwon Hwang[a], Carlo Scalo[a], Andres F. Arrieta[a,*]

[a]*School of Mechanical Engineering, Purdue University, West Lafayette, IN 47907, USA*

Abstract

In this paper, we have developed a solver based on the message-passing interface (MPI) to enable rapid large-scale simulation of generic metastructures composed of bi- or multi-stable elements. The in-house solver has been thoroughly validated against a commercial numerical solver (Abaqus) and the well-established serial codes from the previous studies. We can achieve up to 4^{th}-order solution accuracy with fully explicit Runge-Kutta (RK) methods, exceeding what many commercial structural analysis tools provide. With our parallel code dedicated to solving specific problem types, the absolute computational speed can be improved by three orders of magnitude, enabling the investigation of a large parameter space. More importantly, the in-house implementation enables an effective distribution of the computational load following the intrinsic structural periodicity, thus achieving efficient parallel scalability. To demonstrate our code's capability to handle massively large problems previously unattainable with existing solvers, we investigate the amplitude-dependent energy transmissibility of bi-stable metabeams and the stability of the transition wave's propagation speed. The achieved numerical and computational performance gains drastically expand the accessible analysis domains of general nonlinear metamaterial and metastructure architectures, thus opening up the potential to uncover new dynamics and enable practical implementations.

Keywords: Metamaterials, Nonlinear waves, Multi-stable systems, MPI applications

*Corresponding author.
E-mail address: aarrieta@purdue.edu

PROGRAM SUMMARY

Program Title: NM^3 (Nonlinear MetaMaterials MPI) solver
CPC Library link to program files: (to be added by Technical Editor)
Developer's repository link: https://github.com/wonnie87/NMCube
Code Ocean capsule: (to be added by Technical Editor)
Licensing provisions: MIT
Programming language: Fortran
Nature of problem: NM^3 enables massively parallel simulations of strongly nonlinear metamaterials and metastructures, including 1D multi-stable lattice with coupled pendula (discrete sine-Gordon model), 1D lattice with quartic on-site potentials (discrete ϕ-4 model), and metabeam with a bi-stable microstructure.
Solution method: Up to the 4^{th}-order explicit Runge-Kutta (RK) methods are implemented in NM^3. The Newmark-β (implicit) method with constant average acceleration is also available if unconditional numerical stability is desired.
Additional comments including restrictions and unusual features: Running NM^3 requires installation of Python (with NumPy library), MPI, and HDF5. A Python script is used to generate input files. The code use MPI system calls to allow a massive parallelization among the compute processes. The code uses HDF5 file format for data storage.

1. Introduction

Mechanical metamaterials are arrangements of engineering units designed to realize effective, unconventional macroscopic properties [1, 2, 3]. One class of such metamaterials featuring bi-stable or multi-stable microstructures [4, 5, 6, 7] displays nonlinearity leading to uncommon dynamics including extreme wave directionality [8, 9, 10], bandgap transmission [11, 12], input-independent dynamics [13], which can be utilized for various engineering applications, such as energy trapping [14, 15, 16], energy harvesting [13, 17], structural programmability [18, 19, 20, 21, 22, 23], and mechanologic [9, 24, 25]. Metamaterials are typically designed with periodic microstructures, and the resulting dynamics are best explained by the associated dispersion relations. For many simple periodic structures, the linear and weakly nonlinear wave dispersion relations [26, 27] can be readily obtained through systematic approaches on a single unit cell, such as Bloch wave theorem [28] or transfer function method [29]. Although simple mathematical designs provide an inexhaustible framework for understand-

ing the fundamental physics in metamaterials, more complex microstructure designs [30, 31, 32, 33, 34, 35, 36, 37, 38] or perturbations in the lattices [39, 40, 41, 20, 42] are typically desired for more practical applications. Obtaining the dispersion relations for such nonlinear metamaterials through analytical means becomes challenging, if not impossible, even with the computer-aided symbolic analysis tools. Furthermore, the recent study utilizing a bi-stable lattice with unconstrained transverse motion has enabled an input-independent energy transfer between incommensurate frequencies that can be orders of magnitude apart, called solitonic resonance [43]. Since the interaction with the transition waves, which are quasi-particles traveling through the metastructure, is fundamental to the appearance of this extreme behavior, its dynamics cannot be obtained from the analyses of its unit cell only.

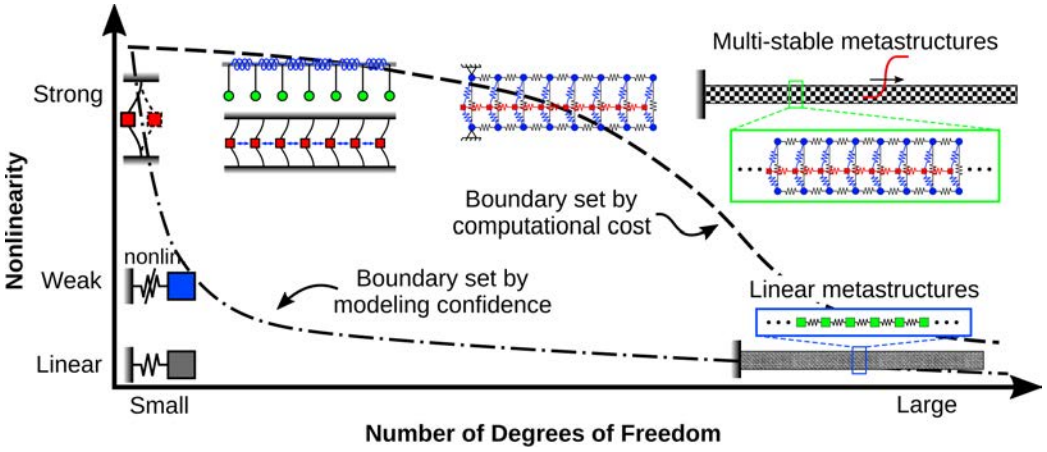

Figure 1: Landscape of nonlinear structural dynamics, modified from Ref. 44. The numerical simulations of multi-stable metastructures with a massively large number of degrees of freedom sit outside the boundary set by the computational cost, warranting the development of high-performance computation tools.

Due to the strong nonlinearities involved in achieving the desired dynamics, the use of systematic analytical approaches is limited, leading to the widespread adoption of numerical approaches, such as the finite element method [45, 46, 47]. Consequently, the analysis of complex metamaterials or metastructures are typically conducted using either commercial solvers, such as COMSOL [32, 33, 34, 37] and Abaqus [31, 19, 48, 20, 43], or in-house implementations [49, 50, 39, 13, 36, 21, 22]. However, employing general-purpose

software packages presents challenges stemming from the use of solvers that are not tailored to work optimally for specific problem classes. Despite the progress afforded by in-house implementations, less attention has been focused on computational performance as most analyses are limited to small finite-sized lattices only. Research in metamaterials and matastructures ultimately aims to realize functional structures using miniaturized metamaterial arrangements as constitutive building blocks, thereby requiring simulations of millions of unit cells. Such massive problems were once deemed impossible to solve from the direct computation of its microscopic components, resulting in the development of approximate continuum-limit [51, 52] or reduced-order [53, 54, 55] models. However, important phenomena, such as those arising from the lattice discreteness [56, 57] or engineered defects [58, 59, 60, 13], can be lost in the process. The limitation of solving problems with high complexities is illustrated by the landscape of nonlinear structural dynamics (Fig. 1) [44], showing the numerical simulation boundaries set by the modeling confidence and computational cost in relation to the nonlinearity and problem complexity (e.g., the number of degrees of freedom). For example, the bi-stable architectures with a massively large number of degrees of freedom would sit on the top-right corner of the plot, outside the boundary set by the computational cost. However, as cutting-edge computer technology becomes more affordable and accessible to the general public, the full-scale characterization of such problems is no longer unimaginable with dedicated solvers. Moreover, recent technological advancements have focused on process parallelization as a route to achieving greater computational efficiency [61, 62, 63].

The repeating building blocks of metamaterials and metastructures inherently lend themselves for efficient parallel implementation, such that the computational load can be evenly distributed over multiple local processes. In this paper, we adopt the message-passing interface (MPI) to develop a high-fidelity simulation tool that can potentially run on millions of parallel processes for analyzing generic multi-stable metastructures. Our in-house implementation of Nonlinear MetaMaterials MPI (NM^3) solver allows for accurately analyzing massively-sized problems featuring strong nonlinearity while drastically reducing simulation and analysis time. We leverage the performance gains afforded by NM^3 to investigate the extreme dynamics of multi-stable metastructures. Specifically, massive parallelization allows us to investigate the amplitude-dependent energy transmission in a bi-stable metastructure and the transition waves' propagation speed in a discrete

medium, both of which are infeasible to obtain with general-purpose tools. More generally, the acceleration in simulation time enabled by NM^3 thus allows for investigating dynamics involving different length scales in strongly nonlinear media, opening up an avenue for the analysis and design of systems made of nonlinear mechanical metamaterials.

2. Metastructures of multi-stable unit cells

We consider three different systems sharing multi-stable constitutive unit cells: a one-dimensional (1D) multi-stable lattice with coupled pendula (discrete sine-Gordon model), a 1D lattice with quartic on-site potentials (discrete ϕ-4 model), and a metabeam with a bi-stable microstructure. Although the physical interpretations differ, the governing equations of these architectures share the following form:

$$\boldsymbol{M}_i \frac{d^2\boldsymbol{u}_i}{dt^2} + \boldsymbol{B}_i \frac{d\boldsymbol{u}_i}{dt} = \boldsymbol{f}_i^{\text{intersite}}(\boldsymbol{u}_{i-1}, \boldsymbol{u}_i, \boldsymbol{u}_{i+1}) + \boldsymbol{f}_i^{\text{onsite}}(\boldsymbol{u}_i) + \boldsymbol{p}_i, \qquad (1)$$

where $\boldsymbol{M}_i$, $\boldsymbol{B}_i$, $\boldsymbol{f}_i^{\text{intersite}}$, $\boldsymbol{f}_i^{\text{onsite}}$, $\boldsymbol{p}_i$, and $\boldsymbol{u}_i$ are the inertial matrix, damping matrix, inter-site force vector, on-site force vector, external force vector, and displacement vector of the i^{th} unit cell, respectively. We solve for the global displacement vector $\boldsymbol{u} = \left[\boldsymbol{u}_1^\intercal, ..., \boldsymbol{u}_{N_{\text{UC}}}^\intercal\right]^\intercal = \left[u_1, ..., u_{N_{\text{DoF}}}\right]^\intercal$, where N_{UC} and N_{DoF} are the total number of unit cells and the total number of degrees of freedom (e.g., $N_{\text{DoF}} = 6 \times N_{\text{UC}}$ for the metabeam problem), respectively. The detailed designs and the specific expressions for the governing equations of the systems considered in this study can be found in Appendix B and the respective references [43, 51]. This set of governing equations is highly coupled and nonlinear, for which the closed-form solutions are not readily available. Thus, the implementation of an efficient numerical solution tool is warranted to investigate their rich dynamics fully. Although only three multi-stable architectures are introduced in this study, the implementation strategy to follow is applicable to any other systems following the form of Eq. (1), rendering the code highly extensible. NM^3 is capable of handling tunable lattice properties (e.g., engineered defects or graded properties) and various boundary conditions (e.g., cantilevered, simply-supported, or fixed-fixed conditions) to encompass a variety of applications.

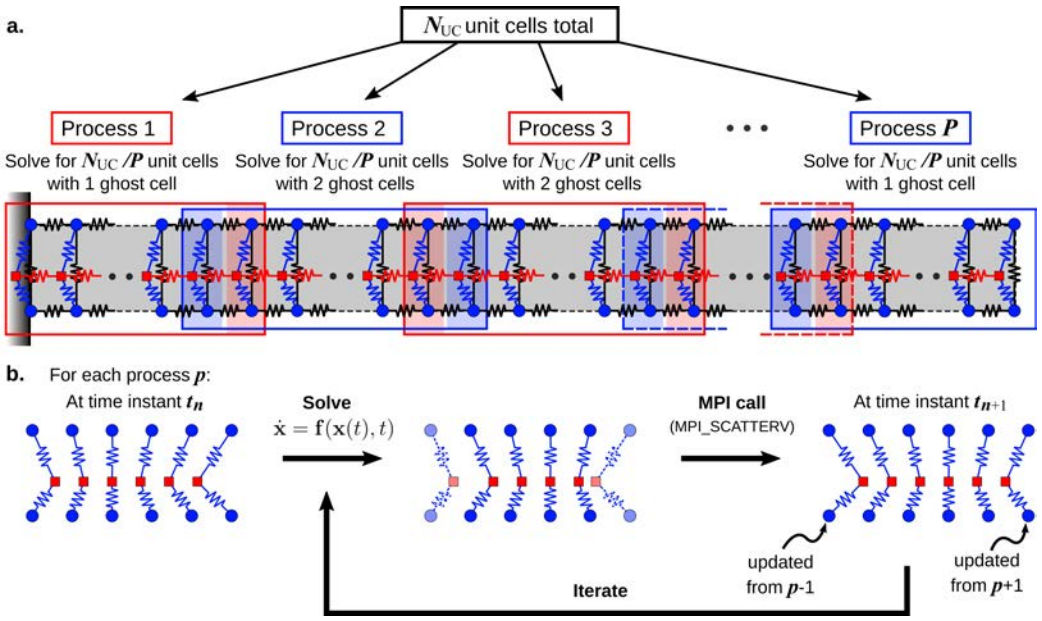

Figure 2: (**a**) Schematic representation of the domain-level parallelization scheme on the metabeam. Each process analyzes the elements in each boxed set only, and the elements in the shaded regions are ghost cells that need to be communicated between the neighboring processes. (**b**) Illustration of the solution algorithm for the explicit method.

3. Numerical and parallelization strategy

The standard explicit 4^{th}-order Runge-Kutta (RK4) method is implemented as the primary numerical solution method. The developed code allows to use any set of the parameters for a general RK4 method, but the classical method (c_1=1/2, c_2=1/2, c_3=1, a_{21}=1/2, a_{32}=1/2, a_{43}=1, b_1=1/6, b_2=1/3, b_3=1/3 and b_4=1/6, following Butcher's notation in Ref. 64) has been adopted in this paper. Lower-order RK methods (1^{st}, 2^{nd} and 3^{rd}) are also optionally available for faster solution convergence at the expense of the numerical accuracy. The governing equations (N_{DoF} second-order differential equations) are recast into a state-space form ($2N_{\text{DoF}}$ first-order differential equations), where the global state vector $\boldsymbol{x} = \left[x_1, x_2, ..., x_{2N_{\text{DoF}}}\right]^{\mathsf{T}}$ contains pairs of the displacements and velocities such that $x_{2j-1} = u_j$ and $x_{2j} = du_j/dt$.

The periodically repeating building blocks of the metamaterial and metastructure architectures provide an excellent framework for a parallelized computational model since the computational load can be readily balanced. To

analyze problems with large numbers of degrees of freedom, we adopt the MPI standard to decompose the problem domain into subdomains to be handled by multiple individual processes in parallel. Starting with N_{UC} unit cells of the global structure, we divide the problem domain into P smaller subdomains, where P is the number of compute processes. The unit cells are equally distributed such that each process computes N_{UC}/P unit cells (assuming N_{UC} is evenly divisible by P) as shown in Fig. 2a for the metabeam. Each process, except the ones containing the structural boundaries, stores two additional unit cell data (ghost cells) to the left and right of the subdomain (the shaded regions in Fig. 2a) since the governing equation requires the states of the adjacent unit cells. An example algorithm for the parallel code with explicit methods is graphically illustrated in Fig. 2b.

Also, a 2^{nd}-order implicit method is implemented for better numerical stability. As the default implicit method, we implement the Newmark-β (NB) method with constant average acceleration (γ=1/2 , β=1/4) [65], which is unconditionally stable. This unconditional stability is conducive to solving stiff problems, thereby enhancing the range of the problems NM^3 can handle and allowing cross-check of the solutions from explicit methods. The displacement vector is decomposed in the same way as that of the explicit method. With the NB method, matrix inversion operations are unavoidable during the Newton-Raphson iterations. Since the tangent stiffness (Jacobian) matrix $\hat{\boldsymbol{k}}_T$ is sparse (e.g., at most 18 nonzero elements for each row for the metabeam problem) but large, the conjugate gradient method is used for matrix inversion. For the parallel implementation of the conjugate gradient method, the block-decomposed vector algorithm [66] is adopted. The decomposition scheme for conjugate gradient method in NM^3 is illustrated in Appendix C.

In order to reduce the latency from the overall data communications among the processes, the MPI calls are made contiguous and invoked sparingly; for example, the number of the RK solver's communication calls for each time iteration is twice the order of the numerical accuracy of the adopted RK method. As the data storage format, HDF5 file format is chosen in anticipation of using parallel I/O capability and storing extended metadata. The baseline output data structure is composed of an $N_t \times N_f$ external force array, $N_t \times 1$ time array, $N_t \times N_{\mathrm{DoF}}$ displacement array, $N_t \times N_{\mathrm{DoF}}$ velocity array, and metadata containing the geometric and compositional information, where N_t and N_f are the numbers of time steps and input sources, respectively. For all the simulations to follow, double-precision (64 bit in the

tested environment) floating-point values are used for both the computation and data storage.

4. Code validation

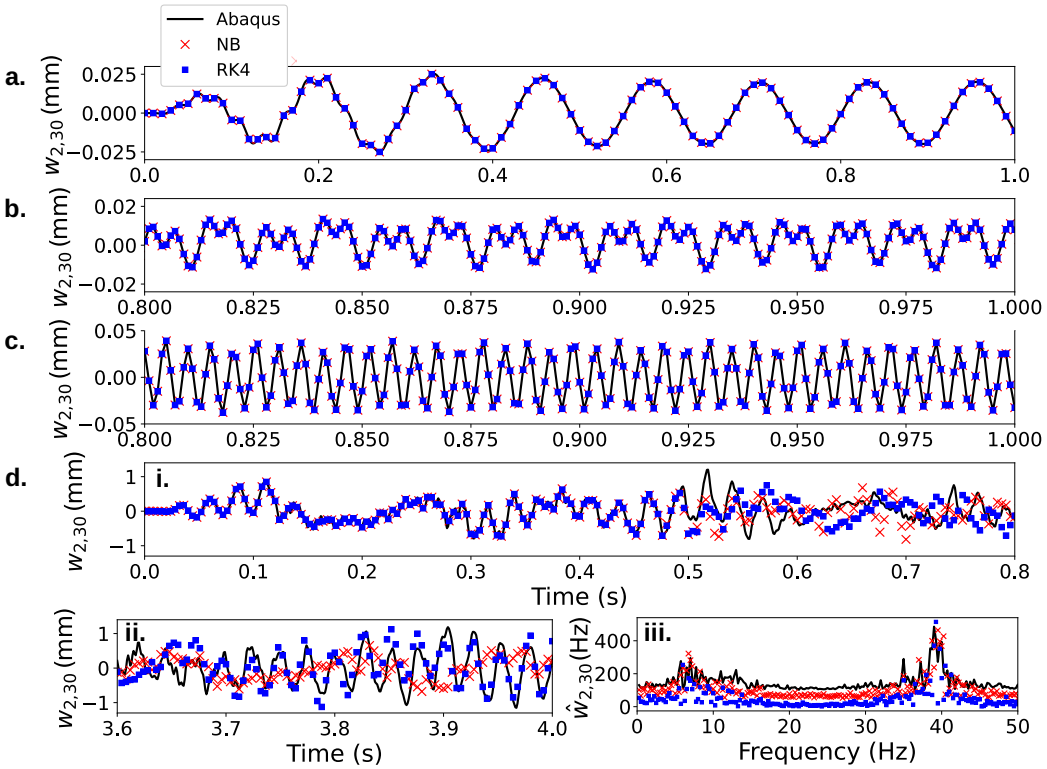

Figure 3: Comparison of the time responses obtained from Abaqus and NM^3's NB and RK4 methods for representative (**a**) low-amplitude response (0.1 N at 8 Hz), (**b**) second harmonic generation (0.5 N at 76 Hz), (**c**) frequency doubling (0.6 N at 96 Hz), and (**d**-i) solitonic resonance (1.4 N at 35 Hz). (ii) The responses of the solitonic resonance at a later simulation time and (iii) their frequency spectra. The frequency spectra plots are shifted vertically for visualization purpose.

We validate our NM^3 code by obtaining the various nonlinear responses of the 1D lattices and metabeam—see Appendix B for the list of the design parameters. For the 1D lattices, the responses are validated with either Matlab or Python codes built from the previous studies—see Appendix D for details. For the metabeam with a bi-stable microstructure, its responses under various in-plane sinusoidal loads (applied at the leftmost unit cell) in a

fixed-free configuration are compared with those from Abaqus simulations to validate the code. Specifically, this type of a metabeam is known to exhibit a wide span of linear and nonlinear dynamics from a single metastructure design, affording different opportunities for energy exchange between disparate frequency ranges. Thus, it is imperative to verify if NM^3 can capture all the key extreme wave phenomena. To that end, four example cases with different characteristic behaviors of the metabeam are tested: low-amplitude, linear-like behavior (0.1 N at 8 Hz), weakly nonlinear behaviors yielding superharmonic generation (0.5 N at 76 Hz) and frequency doubling (0.6 N at 96 Hz), and chaotic behavior leading to solitonic resonance (1.4 N at 35 Hz). For all the example cases, the responses are simulated for 6 s at Δt=10^{-5} s.

Fig. 3 shows the transverse responses $w_{2,30}$ of the top mass at the rightmost (30^{th}) unit of the metabeam. The blue squares and red crosses correspond to the results from the RK4 and the NB methods, respectively. Both follow the black lines (obtained from Abaqus) almost exactly except the solitonic resonance case. The root-mean-square (RMS) errors of the time response between the solution sets from the RK4 method and Abaqus are 1.206×10^{-6} mm (0.1 N at 8 Hz), 9.03×10^{-6} mm (0.5 N at 76 Hz), 3.81×10^{-5} mm (0.6 N at 96 Hz), and 4.71×10^{-1} mm (1.4 N at 35 Hz), respectively; those from the NB method and Abaqus are 5.53×10^{-9} mm (0.1 N at 8 Hz), 2.21×10^{-7} mm (0.5 N at 76 Hz), 7.50×10^{-7} mm (0.6 N at 96 Hz), and 5.18×10^{-1} mm (1.4 N at 35 Hz), respectively. The RMS errors for the three non-chaotic behaviors (Fig. 3a-c) are minimal for the chosen output response ranges. On the other hand, the responses for the solitonic resonance obtained with the three compared methods diverge after several oscillations (Fig. 3d-i,ii). This is not surprising since the solitonic resonance involves a chaotic response that is highly sensitive to any slight perturbations in the initial condition or intermediate values. Thus, different numerical solution algorithms and parameters make it improbable to yield identical responses. Nevertheless, the crucial property of this metastructure architecture is its input-independent characteristic of the output frequency when transition waves are generated within a metabeam. This crucial characteristic is well-captured by NM^3 when analyzing the frequency contents between 2-6 s, obtained with fftpack from Python's scipy library at 1000 Hz sampling frequency without a window function. Indeed, the general shapes of the output frequency spectra from Abaqus simulation, NB, and RK4 method are similar, and the most dominant output frequencies occur at the same frequency ($\sim$40 Hz), confirming the validity of the implemented solution

methods (Fig. 3d-iii).

5. Numerical performance

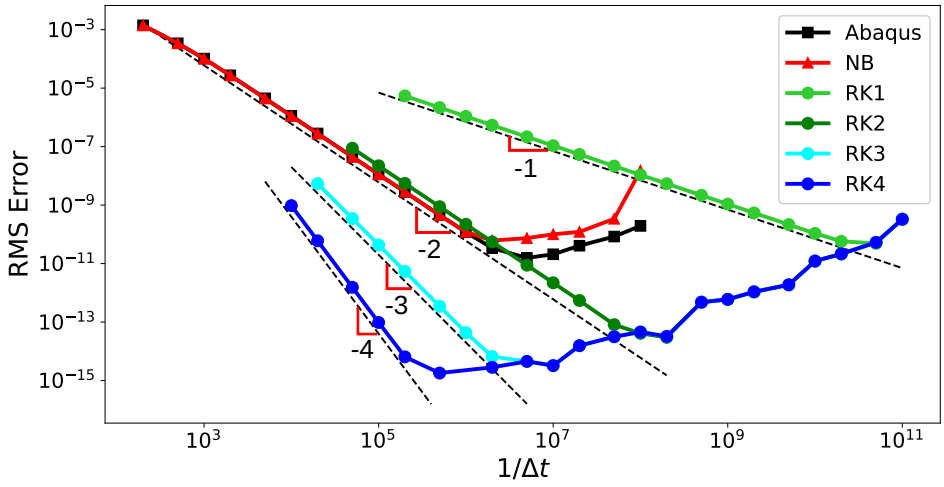

Figure 4: RMS error as a function of time step size for each of the numerical solution methods (Abaqus Implicit, NB, RK1, RK2, RK3, RK4), showing the global truncation errors.

The default implicit Abaqus solver uses Hilber-Hughes-Taylor (HHT-α) method [67] to find solutions. The HHT-α method is a variant of the NB method and thus has the 2^{nd}-order global truncation error. With the flexibility in choosing the numerical solution methods in NMˆ3, the solution accuracy can be extended to higher orders than 2.

For each of the aforementioned solution methods, we solve the solutions of the low-amplitude case introduced in Section 4 at the time instant t=0.01 s, using time steps ranging from 5×10^{-3} s to 10^{-11} s. Since the exact theoretical solution is not available for this type of problem, the numerical solution obtained at one of the small time steps is used as the reference set for the convergence. We expect implicit methods to be more susceptible to the machine's precision limit due to the necessary higher dimensional matrix operations (i.e., the matrix inversions during the Newton-Raphson iterations) and reliance on the residual tolerance. Thus, we rule them out from the choices. Due to the accumulation of round-off errors as approaching the machine's numerical precision limit, the time step cannot be reduced indefinitely to improve the solution accuracy. Hence, the reference solution set is selected

by checking the convergence of the solutions from the RK methods as the time step is reduced. We choose the reference solution as the one obtained at Δt showing the minimum rate of the change of the magnitude, which occurs at $\Delta t_{\text{ref}}=10^{-6}$ s with the RK4 method for this example.

The RMS errors between the solutions obtained at trial Δt's and the reference solution are plotted in a log-log scale in Fig. 4 for different solution methods: Abaqus Implicit, NB, explicit RK1, explicit RK2, explicit RK3, explicit RK4. The negative slope of the curve corresponds to the error's logarithmic rate of decrease as the time step is reduced, hence mathematically indicating the global truncation error. The black dashed lines are the reference lines indicating the slopes of 1, 2, 3, and 4 in order, confirming the desired global truncation errors of the implemented solution methods. The accuracy of the in-house implementation of the NB method closely matches that of Abaqus since Abaqus uses the HHT-α method by default. The small discrepancy in the achievable error is due to the different specification of the residual tolerance. With the RK4 method, we observe a very fast convergence to the accurate solution as expected. Furthermore, Abaqus requires to be run with $\Delta t=2 \times 10^{-7}$ s to reach a similar level of accuracy available to our NMˆ3's RK4 implementation with $\Delta t=5 \times 10^{-5}$ s. Thus, the resulting numerical accuracy plot confirms the extension of the solution accuracy to higher orders than what a commercial solver offers. The fast approach to the exact solution allows the solver to use a larger time step with less susceptibility to the hardware precision error, which is useful when solving stiff problems.

6. Computational performance

The computational performance of NMˆ3 is demonstrated by comparing the measured wallclock times obtained from the in-house solvers and Abaqus solver in computing both fixed-size problems (showing the strong scaling) and scaled-size problems (showing the weak scaling) with respect to the number of processes. Since the primary interest of NMˆ3 is in the analyses of strongly nonlinear behavior, we use the same unit cell design as the solitonic resonance example in Section 4—the computational performance for a linear-like behavior is also provided in Appendix E for reference. The responses are simulated for 1 s at $\Delta t=10^{-4}$ s, and file I/O operations have been disabled in order to focus the comparison on the CPU time when measuring the wallclock times. Both for the NB and the RK4 methods, the wallclock times are

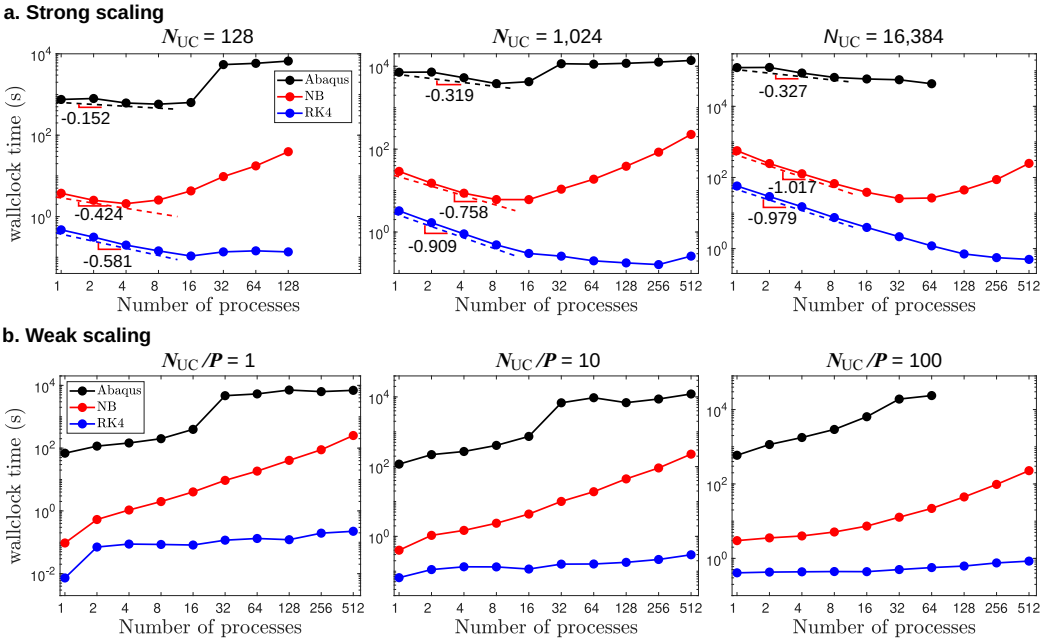

Figure 5: Wallclock time comparison among Abaqus Implicit solver and NM^3's NB and RK4 methods, showing (**a**) the strong scalings and (**b**) the weak scalings with the number of processes.

measured by comparing the timestamps from the MPI-intrinsic subroutine MPI_WTIME placed before and after the main computation loop while all the HDF5 calls are commented out. For Abaqus simulations, the frequencies for the field and history output and restart request have been set to zeros, and the output diagnostics feature is disabled. The wallclock times from Abaqus simulations are directly read off from the generated .dat (or .msg) files. All the simulations are tested on dedicated compute nodes (Two Sky Lake CPUs at 2.6 GHz with 24 cores and 96 GB memory per node) at Purdue University, with no other user programs running.

For the strong scaling analysis [68], three differently sized (128, 1,024, and 16,384 unit cells, each) metabeam problems are simulated with a varying number of processes. To avoid any MPI load balancing issue, only the metabeams composed of power of 2 unit cells are tested for a power of 2 number of processes. The wallclock times are measured three times for each case for a statistical purpose, and their averages are plotted with respect to the number of processes in Fig. 5a on a log-log scale. NM^3's RK4 solver

exhibits absolute computation times about three orders of magnitude faster than those of the Abaqus solver for all three problem sizes. Since NM^3 is streamlined to solve a particular type of problem, the absolute computation times from the NB method are also faster than those from the Abaqus simulations. These results are expected since Abaqus is a general-purpose commercial solver, which performs many cross-checks and other tasks nontransparent to the end-users. Also, Abaqus is not aware of the specific periodic arrangement of the structure, and thus it may not adopt the best load balancing strategy. Hence, a better comparison measure is how the wallclock time scales with the increased number of the processes. To that end, the linear regression (Appendix F) of the first few data points of the wallclock times for each method is plotted with the dashed line slightly beneath the corresponding curve. The slopes of the regressed lines (as indicated in Fig. 5a) immediately show that the computation time of NM^3's RK4 method scales much better with the number of processes and more consistently throughout the problem sizes than those of Abaqus. The NB solver's performance tends to scale adversely when the number of processes becomes very large compared to the given problem size. This is caused by the increased MPI overhead of the global matrix inversion process.

The weak scaling exhibits the best utilization of the available resources rather than a reduction in the absolute computation times with the increased number of processes [69]. For the weak scaling analysis, the problem size (N_{UC}) is proportionally scaled with the number of processes P. For example, if we desire each process to compute 10 unit cells, we measure the wallclock times for a 10-unit cell metabeam with one process, a 20-unit cell metabeam with two processes, and a 40-unit cell metabeam with four processes. Figure 5b shows the weak scalings for the problem densities (N_{UC}/P) of 1, 10, and 100 unit cells. The implementation of the explicit solution schemes does not contain inherently sequential tasks if the input processing is disregarded. Thus, the measured wallclock times from the RK4 method for the scaled problem sizes remain nearly constant throughout the number of processes. The slight increase is deemed to arise from the latency of the underlying computer architecture's interconnection network scheme. For the implicit methods (Abaqus and NB), parallel implementation of matrix inversion is required, which is dependent on the global problem size (e.g., data communications need to be made among all the processes for the calculation of the residual sums), hence their weak scalabilities are not as balanced as the scalability of the RK4 method.

7. Nonlinear energy transmissibility of bi-stable metabeam

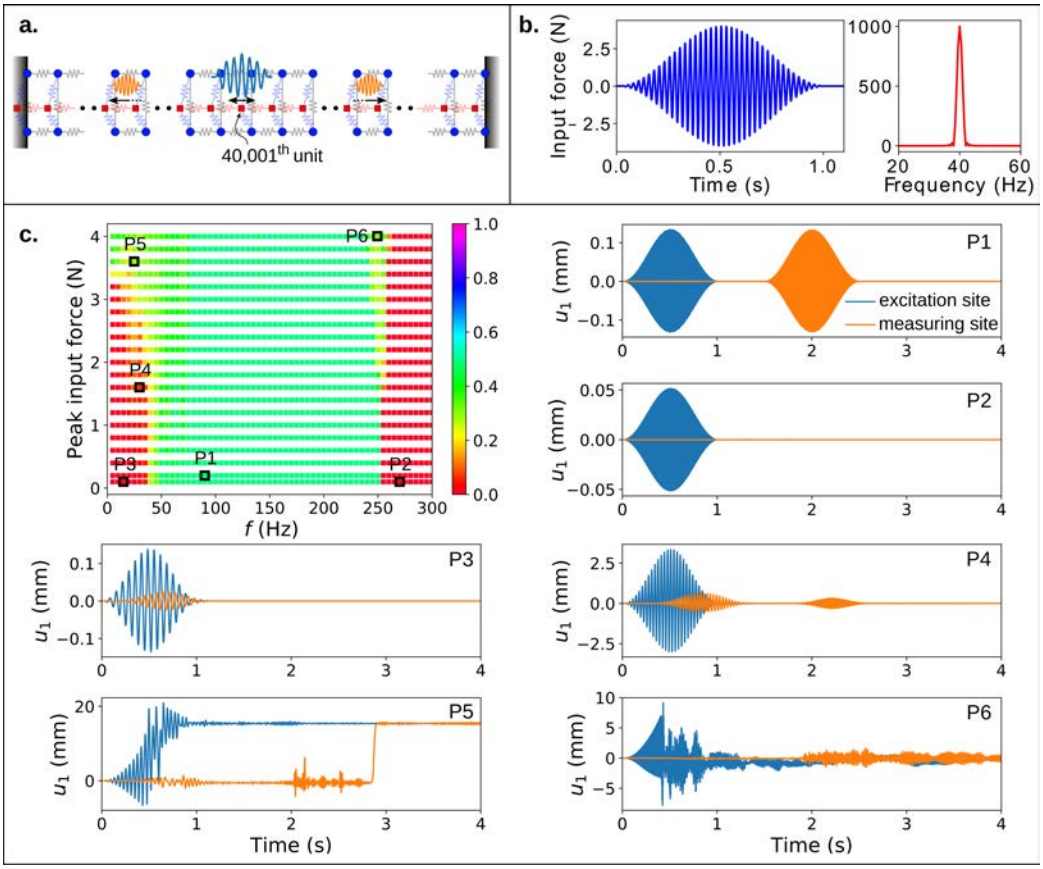

Figure 6: (**a**) Schematic diagram of a very long metabeam under a modulated sinusoidal input force at the midpoint of the metabeam. (**b**) Representative input force, localized both in temporal and spatial domains. (**c**) Energy transmissibility diagram, showing the ratio between the transmitted energy to a distant site and the input energy. Around the diagram are the in-plane displacements u_1 at the excitation site (40,001$^{\text{st}}$ unit) and the measuring site (41,001$^{\text{st}}$ unit) for the selected forcing conditions.

The improved computational capability from the massively parallelized spatial domain is especially useful to explore the nonlinear behaviors of periodic structures, for which the theoretical solutions are in general not available. Our code enables users to access previously unreachable analysis regimes. An important example is the nonlinear amplitude-dependent energy propagation in a metabeam with bi-stable microstructure. To that end,

a very long metabeam (N_{UC}= 80,001) with a fixed-fixed boundary is excited by a modulated sinusoidal force of the form $F_{\mathrm{in}}(t) = F\sin(2\pi f_{\mathrm{in}}t)\sin^2(t - c_1)/(c_2 - c_1)$ for $c_1 \leq t \leq c_2$ at the center mass of 40,001$^{\mathrm{st}}$ unit cell in the in-plane direction (Fig. 6a), where F and f_{in} are the forcing amplitude and the input frequency. A representative input signal is shown in Fig. 6b, where the values for c_1 (10 ms) and c_2 (1,010 ms) are chosen such that the signal is localized in both time and frequency domains of interest. We limit the simulation time such that no major reflections occur during the analysis. The responses are simulated with NM^3's RK4 solver for 10 s at Δt=10^{-4} s. The metabeam is undamped except at both the leftmost and rightmost 2,000 unit cells, where mass proportional damping coefficients gradually increasing from 0.0661 s^{-1} to 132.1 s^{-1} are applied to eliminate unphysically fast low-frequency waves reflecting back toward the origin.

To quantify the energy propagation in the metabeam, we define energy transmissibility as the ratio between the energy transmitted to the measuring site (41,001$^{\mathrm{st}}$ unit) and the input energy into the system—see Appendix G for details. The reason for considering the energy transmissibility as the wave propagation measure rather than the shape of the wave pulse itself is that the characteristics of the waves change depending on the forcing amplitude; for example, the propagated pulse does not retain the original modulated sinusoidal input for cases P4, P5, and P6 in Fig. 6c. Figure 6c summarizes the energy transmissibility for various combinations of the input force and frequency. Note that, in general, the transmissibility does not exceed 50% since the metabeam is symmetrically excited at the midpoint, making the waves travel both to the left and right. Thus, the closer the color is to green (red), the more (less) input energy survives through the measuring site. In the sense that the transmissibility shows the original energy proportion transferred to a distant location, it can be regarded as a practical measure of the pass or stop bands of the metastructure. The boundary of the pass band under low input forces closely matches that of the analytically obtained linear dispersion relation—see Appendix H for details. Several representative time responses of the waves at the excitation and measuring sites are plotted alongside the energy transmissibility diagram—see the supplementary videos S1-S6 for the responses in the space configuration. Case P1 exhibits undisturbed transmission of the original input pulse through the metabeam, while P2 exhibits a stop band behavior. P3 shows a similar stop band behavior, although a small portion of the input energy can survive through a coupled wave mode. P4 shows the separation into two pulse groups, the details of which are addressed

shortly. For a large enough forcing amplitude, we observe the formation of transition waves (P5) or transformation into nondefinite waveforms (P6).

From the energy transmissibility diagram, we can observe the overall effect of the amplitude dependence on the energy propagation: the pass band expands with an increasing forcing amplitude. What is more remarkable than the observed dynamics is that the diagram not only allows us to observe the different nonlinear behaviors of this metastructure but also epitomizes the capability afforded by NMˆ3. Every square in the energy transmissibility diagram represents the individual dynamic (100,000 time steps) simulation of a metabeam with 80,001 unit cells (480,006 DoFs). There are 1,260 such squares (i.e., individual simulations) in this energy transmissibility diagram, which is a scale too large to be conceivable with a serial implementation or with commercial solvers. On our tested platform, it takes about 12 hours to obtain this diagram when run concurrently with 21 sets of 120 processes, including file I/O operations.

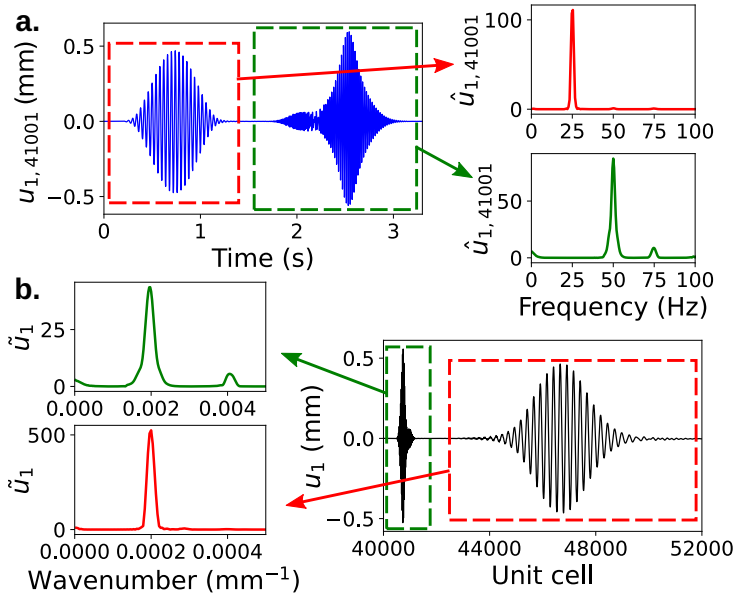

Figure 7: Separation of the input pulse into multiple pulses for F=1.8 N at 25 Hz. (**a**) In-plane response $u_{1,41001}$ at the measuring site (41,001$^{\text{st}}$ unit cell) with the corresponding temporal frequency of each pulse group. (**b**) In-plane displacement u_1 in the spatial domain with the corresponding spatial frequency of each pulse group.

With proper terminating conditions (e.g., artificial damping), the same energy transmissibility diagram can be obtained without using a massive

domain since only the time data passing through the specific points of interest are needed to calculate the transmitted energy. However, we can obtain information about the associated waves in its entirety with the additional solution dimension in the spatial domain. When the forcing amplitude is moderately high, we observe that the input pulse separates into multiple pulses near the boundary of the low-frequency stop band and the pass band (e.g., P4 in Fig. 6c). We employ the term moderate in this paper to describe a force not large enough to trigger transition waves. One such phenomenon occurs for F=1.8 N at 25 Hz, and the corresponding time response of the 41,001$^{\text{st}}$ unit cell is plotted in Fig. 7a. We can identify two main pulse groups. To be precise, the second group itself is composed of two separate identifiable subgroups. Since one of the subgroups dominates the other, we disregard such a detailed description of the pulses for analytical simplicity. The dominant frequency of the first group is the same as the carrier frequency, and the dominant frequency of the second is twice the carrier frequency, showing harmonic generation of the main input frequency.

Since our parallel solver allows for efficient analyses of very large spatial domains, the corresponding wavenumber of each pulse group can also be identified from the analysis of waveform in the space configuration. Fig. 7b shows the responses in the space configuration, obtained at t=2 s. The same pulse groups are indicated by matching colors in Figs. 7a,b (red dashed box for the fast-moving pulse and green dashed box for the slow-moving pulse). We obtain FFT of each pulse in the space configuration to extract spatial frequency spectrum, yielding wavenumbers ν=$\sim$0.0002 m^{-1} for 25 Hz pulse and ν=$\sim$0.002 m^{-1} for 50 Hz pulse. By collecting the wavefrequency-wavenumber pairs for each input excitation, NM^3 allows for numerically constructing the medium's nonlinear dispersion relation, which fundamentally governs its dynamic behaviors. Thus, we provide a route to obtaining nonlinear dispersion relations given the intractability of an analytical treatment for this type of strongly nonlinear system.

8. Propagation speeds of the transition waves in a ϕ-4 model

For the continuous ϕ-4 model with symmetric on-site potential, there is a known theoretical solution for the displacement $u(x,t)$ in the following canonical form: $u(x,t) = \tanh \frac{x-vt}{\sqrt{2}\sqrt{1-v^2}}$ [52], where x and v are the spatial coordinate and the constant propagation speed of the transition wave. This particular solution indicates that the transition wave has an invariant wave-

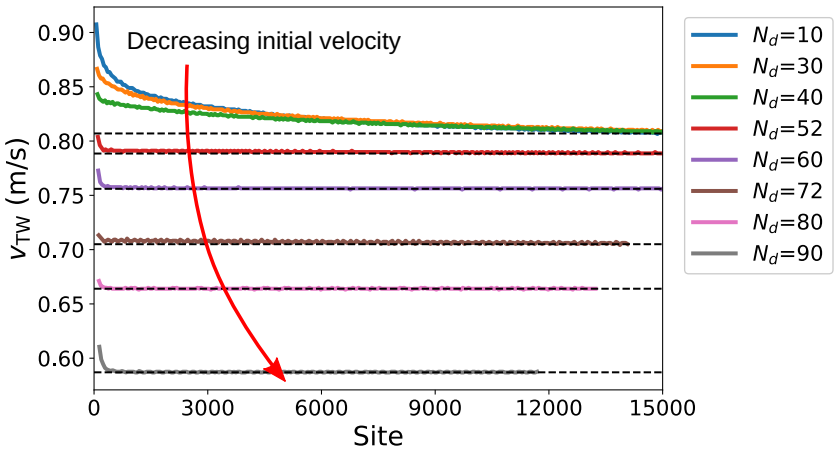

Figure 8: Propagation speeds of the transition waves in a discrete ϕ-4 model under various initial velocities (obtained with the different numbers of units N_d in the dissipation section) of the transition waves. Black dashed lines are constant reference lines.

form for any subsonic propagation. It is of interest to investigate if such a d'Alembert's (traveling at a constant speed) solution exists for the discrete ϕ-4 model as well. To that end, the responses of a periodic lattice composed of 20,000 bi-stable unit cells are simulated for 20,000 s at Δt=10^{-3} s, where a discrete symmetric unit cell design (in Tab. B.1) is used. A transition wave can be easily generated by imposing an initial velocity on the first unit cell (1.5 m/s in this example). However, this imposed initial velocity does not directly translate into the initial propagation speed of the transition wave due to the chaotic snap-through dynamics involved when it is first triggered. In this type of lattice, dissipation is known to slow down the transition waves [40]. Hence, to manipulate the propagation speed of the transition wave, we intentionally impose on-site damping (0.01 Ns/m in this example) to the first N_d unit cells. The longer the dissipation region spans, the slower the transition wave becomes.

Numerically, the propagation speed v_{TW} of the transition wave can be obtained by finding the zero crossings of the time responses at two nearby sites and then dividing the distance between the two sites by their time difference. Figure 8 shows the instantaneous propagation speeds of the transition waves for various initial velocities (obtained with dissipation sections of 10, 30, 40, 52, 60, 72, 80, and 90 units long, respectively). As opposed to the theoretical traveling wave solution in the continuous ϕ-4 model, the transition wave in the discrete counterpart does not always propagate at a constant speed.

Rather, there exists a certain critical value, above which the propagation speed continuously decreases at a slow, diminishing rate (e.g., the top three curves) but below which the propagation speed settles relatively quickly to a nearly constant value (e.g., the bottom five curves). This behavior is presumably due to the lattice discreteness effect coupled with the interaction with the allowed phase speeds of the phonons. In a discrete lattice, oscillatory tails (radiated phonons) are generated as a result of snap-through transitions that the traveling transition wave yields. If the propagation speeds of these oscillatory tails are slower than the speed of the transition wave, the transition wave continuously loses a portion of its transport energy for the generation of these tails, hence slowing down, similar to the mechanism leading to the quasi-steady states in a discrete sine-Gordon model [57]. On the contrary, if the tails travel faster than the transition wave, the tails and the main transition wave always interact with each other, conserving its transport energy. The verification of this postulate and more in-depth investigation on the stability of the transition wave speed in a discrete ϕ-4 model will be addressed in a separate paper.

9. Conclusion

We adopt a message-passing model to achieve massively parallelized computation for dynamics of multi-stable architectures in metamaterials and metastructures. The implemented NM^3 solvers are thoroughly validated against existing solvers. The freedom of choice on the numerical solution schemes allows the global truncation error to be extended to higher orders, surpassing the 2^{nd}-order accuracy limit posed by implicit Abaqus solver, a package widely used in both academia and industry. The increased accuracy opens up the possibility for NM^3 to be readily coupled with existing high-fidelity solvers in other dynamical systems, providing solutions to more complex problems. Since NM^3 is dedicated to solving multi-stable architectures, the computation time can be reduced by as much as three orders of magnitude compared to that of a commercial solver. Additionally, the matamaterial and metastructure architectures' inherent periodicity provides an excellent framework for balanced computational loads among the compute processes, achieving a great scaling with the number of processes. The improved computational performance allows for analyzing a large number of large-scale problems expeditiously and investigating previously unreachable analysis regimes—the nonlinear energy transmissibility diagram of a

metabeam and the propagation speeds of the transition waves in a discrete ϕ-4 model showcase such examples.

The strongly nonlinear behaviors investigated in this paper are only a small fraction of the possible investigations enabled by this parallel implementation. The frontier of the extreme dynamics in 1D multi-stable metamaterials and metastructures still has much to be explored. Future implementation of nonlinear metamaterials and metastructures in practical systems will require a higher level of design complexity, for which analytical means to investigate their dynamics become unmanageable. The implemented computational design is generally applicable and thus readily extendable to any dynamical system as long as the governing equations and their Jacobians (for implicit numerical methods) are available—see supplementary material S7 for an example code update process. Particularly, the code's extension into higher-dimensional architectures, such as metaplates (i.e., 2D slender metastructures) with a multi-stable microstructure, can provide an interface to be used in conjunction with MPI-based high-performance fluid solvers [70, 71, 72], enabling fast simulations of flow stabilization problems through fluid-metastructure interactions. NM^3 is open-source and readily upgradable with additional computational schemes, such as OpenMP and dynamic domain decomposition, in response to increased communication overhead or non-homogeneous computational load distribution. In essence, NM^3 can provide an excellent numerical alternative to investigate previously unreachable analysis regimes for general nonlinear metamaterials and metastructures.

CRediT authorship contribution statement

Myungwon Hwang: Software, Validation, Formal analysis, Writing - Original Draft. **Carlo Scalo:** Conceptualization, Supervision, Resources, Funding acquisition, Writing - Review & Editing. **Andres F. Arrieta:** Conceptualization, Supervision, Resources, Funding acquisition, Writing - Review & Editing.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgements

The authors acknowledge the financial support from the National Science Foundation under grant CMMI-1935137.

Appendix A. Running MPI jobs

The parallel Fortran script is first compiled with an MPI Fortran wrapper compiler (Intel MPI library 2017 Update 1 is used throughout the examples in this paper). The input parameters for the numerical solution method and the multi-stable architecture design are then stored in a Python script. Running the Python script generates two input files: one defining the numerical solution method and the other defining the structural design. Both input files can be read by the compiled program. This entire process is automated through GNU make. The script can be compiled by executing a command make in NM^3's root directory; then, each analysis can be performed by running a command make run NP=(number of processes) INP=(input file name).

The source codes and example input files are available at `https://github.com/wonnie87/NMCube`. Additional datasets generated and analyzed in this paper are available from the corresponding author upon request.

Appendix B. Model description

We study three different strongly nonlinear systems featuring multi-stable constitutive units: 1) a 1D multi-stable lattice with coupled pendula, 2) a 1D lattice with quartic on-site potentials, and 3) a metabeam with a bi-stable constitutive microstructure.

The schematic representation of a multi-stable lattice with coupled pendula (discrete sine-Gordon model) is shown in Fig. B.9a. Its governing equation can be written as:

$$I\frac{d^2\theta_i}{dt^2} + b_\theta\frac{d\theta_i}{dt} = k_\theta(\theta_{i+1} - 2\theta_i + \theta_{i-1}) - mgl\sin\theta_i + q_i, \tag{B.1}$$

where θ_i is the rotation of the i^{th} pendulum, I is the moment of inertia of the pendulum about the rotational axis, b_θ is the rotational damping coefficient, k_θ is the stiffness of the torsional spring coupling two neighboring pendula, m is the tip mass of the pendulum, g is the gravitational constant, l is the length of the pendulum, and q_i is the applied moment at the i^{th} pendulum.

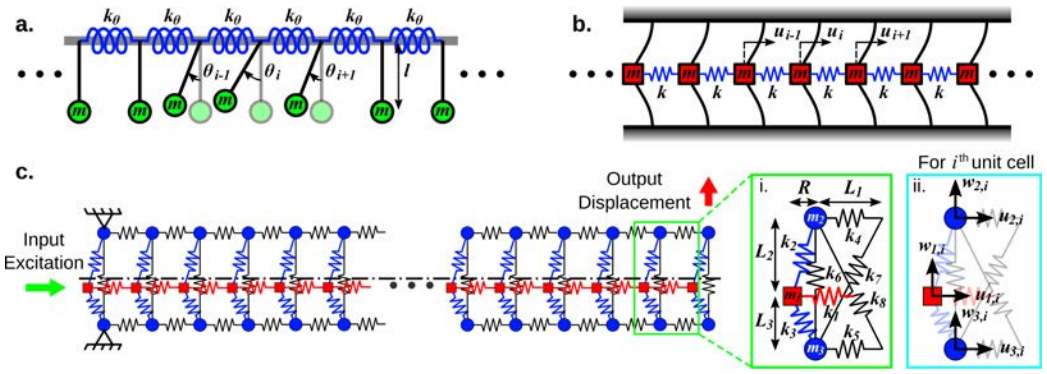

Figure B.9: Schematic representations of (**a**) a 1D multi-stable lattice with coupled pendula, (**b**) a 1D lattice with quartic on-site potentials, and (**c**) a metabeam with a bi-stable microstructure (from Ref. 43). The inset (i) shows the key components of the unit cell. The inset (ii) shows the physical coordinates that describe the motion of each unit cell.

The unit cell of a 1D lattice of quartic on-site potentials (discrete ϕ-4 model) is composed of a bistable element with a mass element connected to the nearest neighbors (Fig. B.9b). The following form of the governing equation is used:

$$m\frac{d^2u_i}{dt^2}+b\frac{du_i}{dt}=k(u_{i+1}-2u_i+u_{i-1})-(C_1+2C_2u_i+3C_3u_i^2+4C_4u_i^3)+p_i, \quad \text{(B.2)}$$

where u_i is the displacement of the i^{th} unit cell, m is the mass of the unit cell, b is the on-site damping coefficient, k is the inter-site spring stiffness, C_j's are the coefficients of the quartic on-site potential, and p_i is the applied force at the i^{th} unit cell. The design parameters for the ϕ-4 lattices used in this paper are listed in Tab. B.1. Although the dynamics of the ϕ-4 lattices do not depend on the lattice spacing, we set the spacing to be 1 m for the numerical calculation of the propagation speed in Section 8.

Lastly, the metabeam with a bi-stable microstructure is shown in Fig. B.9c. Due to the complexity of the constitutive unit, its equations of motion are very lengthy. The full expressions can be found in the supplementary material of Ref. 43. The unit cell of the metabeam is composed of three masses confined in a 2D space, so that its displacement vector can be represented as $\boldsymbol{u}_i = \left[u_{1,i}, w_{1,i}, u_{2,i}, w_{2,i}, u_{3,i}, w_{3,i}\right]^\intercal$, where $u_{j,i}$ and $w_{j,i}$ are the in-plane and out-of-plane displacements of the j^{th} mass of the i^{th} unit cell, respectively. The following single unit cell design in Tab. B.2 is used throughout

Design	k	m	C_1	C_2	C_3	C_4	b
Discrete asymmetric	1	2	-0.036	-0.060	0.012	0.030	0.1
Continuous asymmetric	1	2	-0.0090	-0.0150	0.0030	0.0075	0.1
Discrete symmetric	1	2	0	-0.03	0	0.015	0

Table B.1: Summary of the design parameters for 1D bi-stable lattices. The units of k, m, C_1, C_2, C_3, C_4, and b are N/m, kg, N, N/m, N/m^2, N/m^3, and Ns/m, respectively.

all the simulations of the metabeam problems in this paper. Only the number of the unit cells are varied among the analyses: N_{UC}=30 is used for the code validation and numerical performance analyses; three different values for N_{UC} (128, 1024, 16384) are used for the strong scaling analysis; proportionally scaled sizes to the number of processes for the weak scaling analysis; and, N_{UC}=80,001 is used for the nonlinear energy transmission example. To suppress any undesirable transient effects, we impose a mass proportional damping coefficient γ=9.91 s^{-1} throughout the analyses except for the energy transmission example, where we are interested in the fundamental propagation characteristics.

k_1	k_2	k_3	k_4, k_5	k_6	k_7, k_8	m_1	m_2, m_3	L_1	L_2	L_3	R
1.241	1.076	0.6	100	100	100	2	1	20	40	20	8

Table B.2: Summary of the metabeam design parameters. The units of the stiffness k_j's, mass m_j's, and lengths L_j's and R are N/mm, g, and mm, respectively.

Appendix C. Parallel implementation of the conjugate gradient method

During the Newton-Raphson iteration process of the NB method, a linear system $\hat{\boldsymbol{k}}_T \boldsymbol{\Delta u} = \hat{\boldsymbol{R}}$ needs to be solved for $\boldsymbol{\Delta u}$, where $\boldsymbol{\Delta u}$ and $\hat{\boldsymbol{R}}$ are the change in the global displacement vector between the Newton-Raphson iterations and the global residual force vector, respectively [65]. We adopt the conjugate gradient method, an iterative method, in solving the linear system. Since the force vector of each unit cell depends on the displacements of its own unit and the two neighboring unit cells, only $3n_{\mathrm{DoF}}^2 N_{\mathrm{UC}}$ elements (disregarding the boundary elements) of $\hat{\boldsymbol{k}}_T$ can be strictly nonzero, where

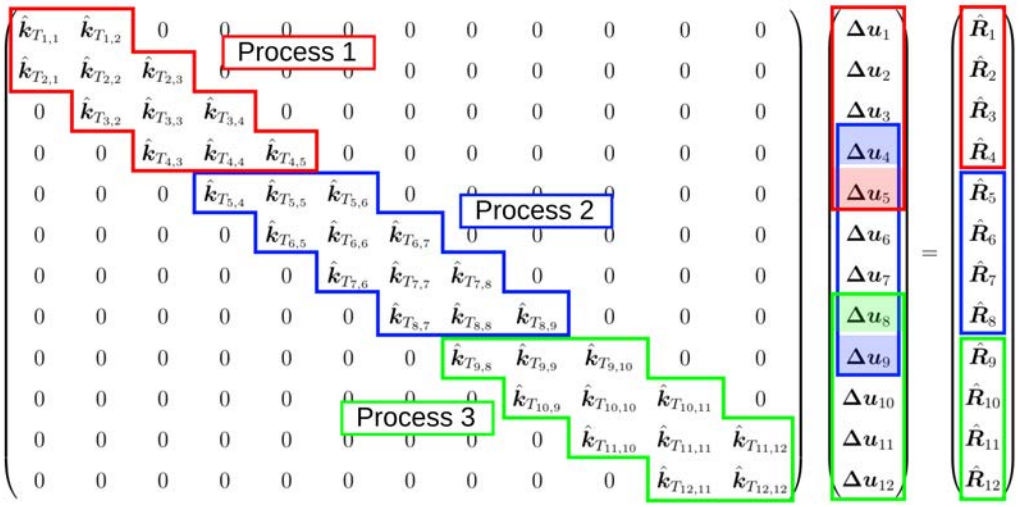

Figure C.10: Example decomposition scheme for the conjugate gradient method in NM^3.

n_{DoF} is the number of degrees of freedom for each unit cell. Accordingly, $\hat{\boldsymbol{k}}_T$ is decomposed such that each process holds $3n_{\mathrm{DoF}}^2 N_{\mathrm{UC}}/P$ elements (assuming N_{UC} is evenly divisible by P). $\hat{\boldsymbol{R}}$ is evenly divided among the processes such that each process holds N_{DoF}/P elements. $\boldsymbol{\Delta u}$ is similarly distributed but contains ghost cells at the boundaries of the subdomain in the same way as the global displacement vector $\boldsymbol{u}$ is distributed. Figure C.10 illustrates an example decomposition for the parallel implementation of the conjugate gradient method in NM^3. In this example, 12 unit cells are decomposed among 3 processes. $\hat{\boldsymbol{k}}_{T_{i,j}}$'s are the components of $\hat{\boldsymbol{k}}_T$. $\boldsymbol{\Delta u}_j$'s and $\hat{\boldsymbol{R}}_j$'s are the change in the displacement vector and the residual vector of the j^{th} unit cell such that $\boldsymbol{\Delta u} = \left[\boldsymbol{\Delta u}_1^{\mathsf{T}}, ..., \boldsymbol{\Delta u}_{N_{\mathrm{UC}}}^{\mathsf{T}}\right]^{\mathsf{T}}$ and $\hat{\boldsymbol{R}} = \left[\hat{\boldsymbol{R}}_1^{\mathsf{T}}, ..., \hat{\boldsymbol{R}}_{N_{\mathrm{UC}}}^{\mathsf{T}}\right]^{\mathsf{T}}$. The shaded regions in $\boldsymbol{\Delta u}$ represent the ghost cells that need to be communicated with the adjacent processes.

Appendix D. Code validation for the 1D lattices

In order to validate the implementation of the discrete sine-Gordon model, a system of four coupled pendula is used. A simple canonical design is used for the unit cell, where the values of I, k_θ, m, g, and l are all set to 1 for the numerical simplicity while b_θ is set to 0. Two qualitatively different behaviors are tested by imposing different initial velocity $d\theta_1/dt|_{t=0}$ at the first element.

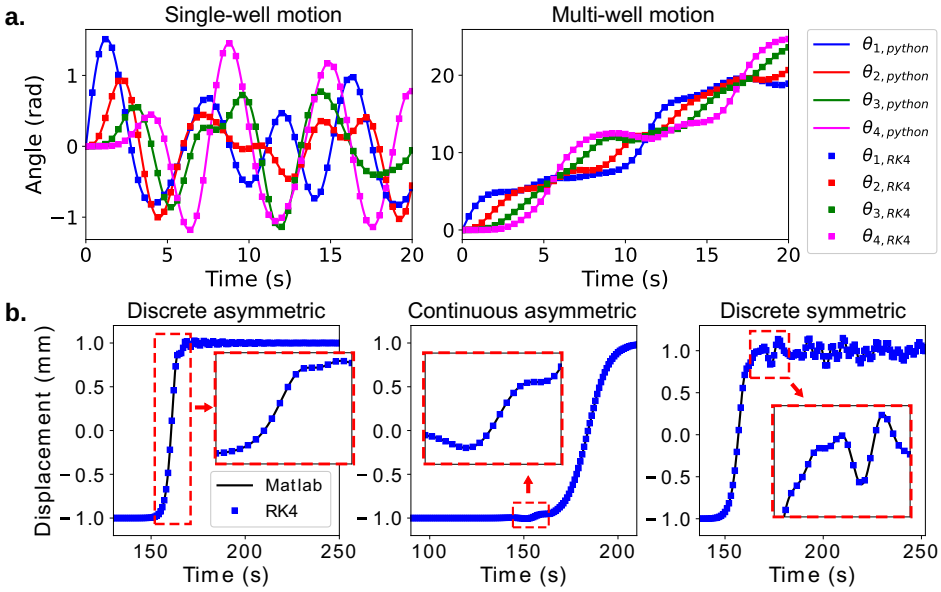

Figure D.11: (**a**) Code validation for the 1D multi-stable lattice with coupled pendula. The response angles of the 3$^{\text{rd}}$ element from a built-in Python function and NM^3's RK4 method are compared for two qualitatively different motions (one bounded in a single potential well, the other traversing through multiple wells). (**b**) Code validation for the discrete 1D lattice with quartic on-site potentials. The displacements of the 100$^{\text{th}}$ element from NM^3's RK4 method are plotted for discrete asymmetric, continuous asymmetric, and discrete symmetric designs and compared with those obtained from the Matlab code in the previous study [13].

The simulations are performed for 20 s with Δt=10^{-3} s. The pendula motions are confined in a single potential well for $d\theta_1/dt|_{t=0}$= 2.0, while the motions traverse multiple wells (i.e., a transition wave is generated) for $d\theta_1/dt|_{t=0}$= 4.6. The reference solution sets are obtained from odeint method in SciPy's (a Python library) integrate module. The responses obtained from NM^3's RK4 method are plotted together with the reference sets in Fig. D.11a, showing almost exact agreement with each other. For example, the RMS errors for the time response of the 3$^{\text{rd}}$ element between the two methods are 5.80×10^{-4} rad and 1.394×10^{-3} rad, respectively, which are small compared to the ranges of its motion.

The exact solution for the discrete 1D lattice with quartic on-site potentials is not available. Hence, its correct implementation is validated by comparing the solutions obtained from our RK4 code with those from the Matlab code used in the previous studies [13], which uses 2$^{\text{nd}}$-order accurate

central difference method. Three example cases are considered: 1) a bi-stable lattice with asymmetric on-site potentials, where the ratio between the inter-site and on-site stiffness is small, representing a typical discrete lattice; 2) a similar lattice with asymmetric on-site potentials but with a larger inter-site to on-site stiffness ratio, effectively closer to a continuum-limit model; 3) a bi-stable lattice with symmetric on-site potentials. All three designs are composed of 500 unit cells, and the corresponding unit cell design parameters are listed in Tab. B.1. Transition waves are triggered by prescribing initial velocities to the first unit cells, the velocities of which are 2 m/s for both of the asymmetric cases and 1 m/s for the symmetric case. The responses are simulated for 400 s with Δt=0.01 s. Figure D.11b shows the displacements of the 100^{th} unit cell for each design, obtained from both codes. For all of the example cases, the solutions match closely, and the RMS errors between the solutions from the two codes are 1.721×10^{-4} m, 3.39×10^{-4} m, and 4.23×10^{-4} m, respectively.

Appendix E. Computational performance for a linear-like problem

To see the effects of system nonlinearity on the computational performance, we plotted the strong and weak scalings for the unit cell design used for the low-amplitude case in Section 6 in Fig. E.12. Each simulation is performed for 1 s at $\Delta t=10^{-4}$ s with file I/O disabled just as the strongly nonlinear (solitonic resonance) example. The wallclock times for the strongly nonlinear case are also plotted in dashed lines for comparison. Both performance measures follow similar trends to those of the strongly nonlinear case. However, the problems involving stronger nonlinearity typically require more iterations for the implicit methods to converge since it induces larger perturbations during the iterative processes (i.e., the matrix inversion and Newton-Raphson method). This added computational cost is manifested by the increased wallclock time for both Abaqus and NB solvers to solve the solitonic resonance case compared to the nearly linear case. In contrast, the fully explicit RK solver involves only a fixed number of operations for a given problem size. Hence, introducing nonlinearity to a problem does not yield additional computational cost, as demonstrated by the near overlaps of the solid (linear-like problem) and dashed (strongly nonlinear problem) time curves for the RK solver. This insensitivity to the problem nonlinearity makes NM^3's RK solver especially shines in the analyses of strongly

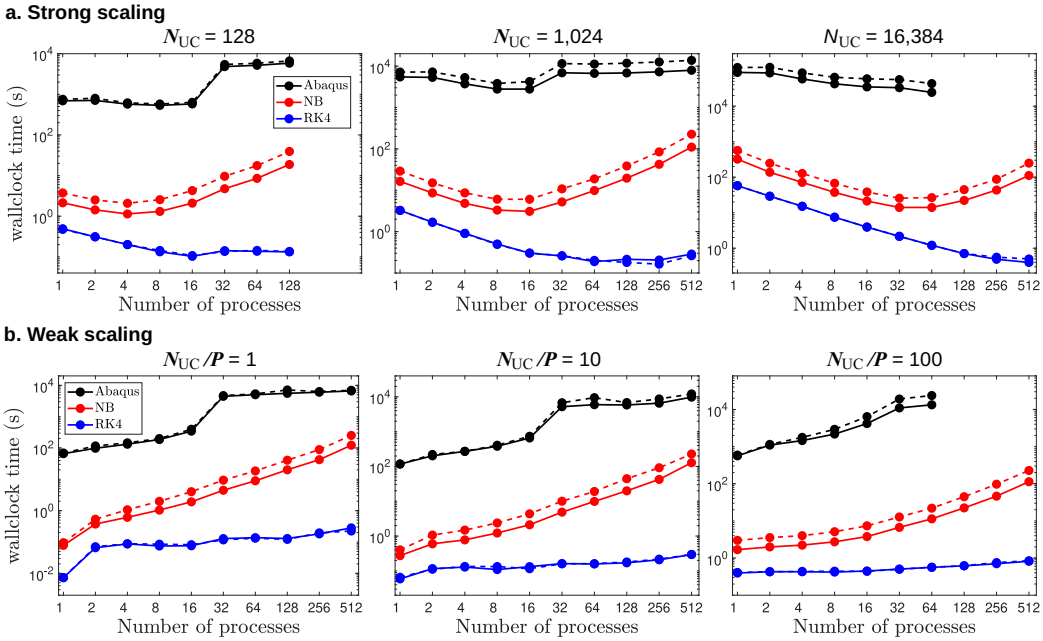

Figure E.12: (a) Strong and (b) weak scalings of Abaqus Implicit solver and NM^3 for the solution of a linear-like problem.

nonlinear problems as long as the solver runs within the numerical stability limit.

Appendix F. Linear regression

The approximate slopes of the wallclock time curves in Fig. 5 are obtained by finding the linear fits of the first four data points. Since the plots are in log-log scales, the approximate relationship is in the form $\log t_{\mathrm{WC},p} = a_0 + a_1 \log p$ (equivalent to $t_{\mathrm{WC},p} = 10^{a_0} p^{a_1}$), where $t_{\mathrm{WC},p}$ is the wallclock time corresponding to the number of the processes p. To solve for a set of the approximate values for constants a_0 and a_1, an overdetermined system of equations can be formed as follows:

$$\underbrace{\begin{pmatrix} 1 & \log 1 \\ 1 & \log 2 \\ 1 & \log 4 \\ 1 & \log 8 \end{pmatrix}}_{\boldsymbol{A}} \begin{pmatrix} a_0 \\ a_1 \end{pmatrix} = \begin{pmatrix} \log t_{\mathrm{WC},1} \\ \log t_{\mathrm{WC},2} \\ \log t_{\mathrm{WC},4} \\ \log t_{\mathrm{WC},8} \end{pmatrix}. \tag{F.1}$$

By premultiplying both sides of Eq. (F.1) by $\boldsymbol{A}^{\intercal}$, we can find a_0 and a_1 that best fit the measured time data.

Appendix G. Calculating energy transmissibility

The energy transmissibility in this paper is defined as the ratio between the transmitted energy to a distant site and the input energy to the system; that is, $\mathrm{TR} = E_{\mathrm{tr}}/E_{\mathrm{in}}$. For the example in Section 7, only the vector component in the in-plane direction is considered to investigate the pass/stop band characteristics in the in-plane direction. Hence, the input energy E_{in} is obtained from the time integration of the instantaneous power supplied to the excitation site in the in-plane direction: $E_{\mathrm{in}} = \int_0^\infty F_{\mathrm{in}} \dot{u}_{1,\mathrm{in}} dt$, where F_{in} is the input force, $\dot{u}_{1,\mathrm{in}}$ is the velocity of the excited mass in the in-plane direction, and the integrand is assumed to be well-behaved.

Similarly, the transmitted energy to the mass of interest is obtained by integrating the power supplied to the mass of the interest from the left (for a right-propagating wave). The force transmitted through the inter-site spring left of the mass of interest is

$$F_{\mathrm{left}} = \frac{\partial}{\partial u_{1,\mathrm{out}}} \left(\frac{1}{2} k_1 \Delta_{1,\mathrm{out}}^2 \right), \tag{G.1}$$

where $u_{1,\mathrm{out}}$ is the in-plane displacement of the center mass at the measuring site, and $\Delta_{1,\mathrm{out}}$ is the deflection of the inter-site spring (with the stiffness k_1) connected to the left of the mass—the full expression for the spring deflection can be found in the supplementary material of Ref. 43. Thus, the transmitted energy can be calculated as $E_{\mathrm{tr}} = \int_0^\infty F_{\mathrm{left}} \dot{u}_{1,\mathrm{out}} dt$.

Appendix H. Linear Dispersion relation

To check the validity of the pass and stop bands obtained from the energy transmissibility diagram (Fig. 6c), we make a comparison with those of the analytically obtained linear dispersion relation. The governing equations can be linearized by keeping only up to the first-order term in the Taylor series of the governing equations. The linear dispersion relation can be obtained by substituting a traveling wave solution $\boldsymbol{u}_i = \bar{\boldsymbol{u}} e^{qL_1 i - \omega t}$ into the linearized equations, where $\bar{\boldsymbol{u}}$ is a constant vector, and solving for the wavenumber q in terms of the wave frequency ω. The solution process yields six highly

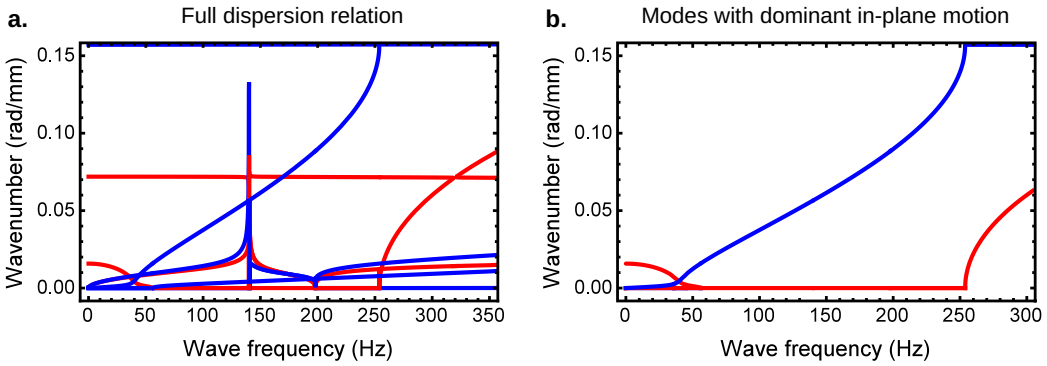

Figure H.13: (**a**) Dispersion relations of the linearized metabeam analyzed in Section 7. The real parts of the wavenumbers are plotted in blue, and the imaginary parts are plotted in red. (**b**) Dispersion relation showing only the modes dominant in the in-plane direction of the center mass.

coupled characteristic equations for the i^{th} unit cell containing many exponential terms for q, the closed-form solution of which is very difficult to obtain if not impossible. Hence, we solve these characteristic equations numerically with the aid of Mathematica. The complete dispersion relation is plotted in Fig. H.13a, which includes every available wave mode in the metabeam. The blue plots represent the real (propagating) parts of the wavenumber, and the red plots represent the imaginary parts, which contribute to the spatial attenuation. Since our output quantity of interest is the in-plane displacements of the center masses, only the modes with the eigenvectors having dominant first entry (corresponding to $u_{1,i}$) are plotted in Fig. H.13b. The upper limit of the pass band is 254 Hz, which closely matches the upper boundary of the energy transmissibility diagram under low forcing amplitude (0.1 N). The lower limit is not definitive since there are multiple modes with the dominant in-plane motions of the center mass. The coupling between these modes contribute to the region with the wavenumbers having both nonzero real and imaginary parts—if it were a strictly 1D lattice (i.e., $w_{1,i}, u_{2,i}, w_{2,i}, u_{3,i}, w_{3,i} = 0$), there would be a sharp boundary at 40 Hz. Depending on the coupling level, a small portion of the incident wave energy is transferred into the other modes, decreasing the transmitted energy. This is manifested by the slight color change near 40 Hz in Fig 6c.

References

[1] N. I. Zheludev, The Road Ahead for Metamaterials, Science 328 (5978) (2010) 582–583. doi:10.1126/science.1186756.

[2] M. I. Hussein, M. J. Leamy, M. Ruzzene, Dynamics of Phononic Materials and Structures: Historical Origins, Recent Progress, and Future Outlook, Applied Mechanics Reviews 66 (4) (2014) 040802. doi:10.1115/1.4026911.

[3] G. Ma, P. Sheng, Acoustic metamaterials: From local resonances to broad horizons, Science Advances 2 (2) (2016) e1501595. doi:10.1126/sciadv.1501595.

[4] N. Nadkarni, C. Daraio, D. M. Kochmann, Dynamics of periodic mechanical structures containing bistable elastic elements: From elastic to solitary wave propagation, Physical Review E 90 (2) (2014) 023204. doi:10.1103/PhysRevE.90.023204.

[5] S. Katz, S. Givli, Solitary waves in a bistable lattice, Extreme Mechanics Letters 22 (2018) 106–111. doi:10.1016/j.eml.2018.06.003.

[6] B. Deng, P. Wang, V. Tournat, K. Bertoldi, Nonlinear transition waves in free-standing bistable chains, Journal of the Mechanics and Physics of Solids 136 (2020) 103661. doi:10.1016/j.jmps.2019.07.004.

[7] S. Katz, S. Givli, Solitary waves in a nonintegrable chain with double-well potentials, Physical Review E 100 (3) (2019) 32209. doi:10.1103/PhysRevE.100.032209.

[8] N. Nadkarni, A. F. Arrieta, C. Chong, D. M. Kochmann, C. Daraio, Unidirectional Transition Waves in Bistable Lattices, Physical Review Letters 116 (24) (2016) 244501. doi:10.1103/PhysRevLett.116.244501.

[9] J. R. Raney, N. Nadkarni, C. Daraio, D. M. Kochmann, J. A. Lewis, K. Bertoldi, Stable propagation of mechanical signals in soft media using stored elastic energy, Proceedings of the National Academy of Sciences 113 (35) (2016) 201604838. doi:10.1073/pnas.1604838113.

[10] H. Fang, K. W. Wang, S. Li, Asymmetric energy barrier and mechanical diode effect from folding multi-stable stacked-origami, Extreme Mechanics Letters 17 (2017) 7–15. doi:10.1016/j.eml.2017.09.008.

[11] J. Leon, Nonlinear supratransmission as a fundamental instability, Physics Letters, Section A: General, Atomic and Solid State Physics 319 (1-2) (2003) 130–136. arXiv:0310007, doi:10.1016/j.physleta.2003.10.012.

[12] M. J. Frazier, D. M. Kochmann, Band gap transmission in periodic bistable mechanical systems, Journal of Sound and Vibration 388 (2017) 315–326. doi:10.1016/j.jsv.2016.10.041.

[13] M. Hwang, A. F. Arrieta, Input-Independent Energy Harvesting in Bistable Lattices from Transition Waves, Scientific Reports 8 (1) (2018) 3630. doi:10.1038/s41598-018-22003-7.

[14] D. Restrepo, N. D. Mankame, P. D. Zavattieri, Phase transforming cellular materials, Extreme Mechanics Letters 4 (2015) 52–60. doi:10.1016/j.eml.2015.08.001.

[15] S. Shan, S. H. Kang, J. R. Raney, P. Wang, L. Fang, F. Candido, J. A. Lewis, K. Bertoldi, Multistable Architected Materials for Trapping Elastic Strain Energy, Advanced Materials 27 (29) (2015) 4296–4301. doi:10.1002/adma.201501708.

[16] N. Kidambi, R. L. Harne, K. W. Wang, Energy capture and storage in asymmetrically multistable modular structures inspired by skeletal muscle, Smart Materials and Structures 26 (8) (2017) 085011. doi:10.1088/1361-665x/aa721a.

[17] M. Hwang, A. F. Arrieta, Topological wave energy harvesting in bistable lattices, Smart Materials and Structures 31 (1) (2021) 015021. doi:10.1088/1361-665X/ac37ff.

[18] B. Haghpanah, L. Salari-Sharif, P. Pourrajab, J. Hopkins, L. Valdevit, Multistable Shape-Reconfigurable Architected Materials, Advanced Materials 28 (36) (2016) 7915–7920. doi:10.1002/adma.201601650.

[19] J. Meaud, K. Che, Tuning elastic wave propagation in multistable architected materials, International Journal of Solids and Structures 122-123 (2017) 69–80. doi:10.1016/j.ijsolstr.2017.05.042.

[20] L. Jin, R. Khajehtourian, J. Mueller, A. Rafsanjani, V. Tournat, K. Bertoldi, D. M. Kochmann, Guided transition waves in multistable

mechanical metamaterials, Proceedings of the National Academy of Sciences 117 (5) (2020) 2319–2325. doi:10.1073/pnas.1913228117.

[21] H. Yasuda, L. M. Korpas, J. R. Raney, Transition Waves and Formation of Domain Walls in Multistable Mechanical Metamaterials, Physical Review Applied 13 (5) (2020) 1. doi:10.1103/PhysRevApplied.13.054067.

[22] A. Zareei, B. Deng, K. Bertoldi, Harnessing transition waves to realize deployable structures, Proceedings of the National Academy of Sciences of the United States of America 117 (8) (2020) 4015–4020. doi:10.1073/pnas.1917887117.

[23] J. P. Udani, A. F. Arrieta, Programmable mechanical metastructures from locally bistable domes, Extreme Mechanics Letters 42 (2021) 101081. doi:10.1016/j.eml.2020.101081.

[24] Y. Song, R. M. Panas, S. Chizari, L. A. Shaw, J. A. Jackson, J. B. Hopkins, A. J. Pascall, Additively manufacturable micro-mechanical logic gates, Nature Communications 10 (1) (2019) 1–6. doi:10.1038/s41467-019-08678-0.

[25] J. A. Faber, J. P. Udani, K. S. Riley, A. R. Studart, A. F. Arrieta, Dome-Patterned Metamaterial Sheets, Advanced Science 7 (22) (2020) 1–9. doi:10.1002/advs.202001955.

[26] R. K. Narisetti, M. J. Leamy, M. Ruzzene, A perturbation approach for predicting wave propagation in one-dimensional nonlinear periodic structures, Journal of Vibration and Acoustics, Transactions of the ASME 132 (3) (2010) 0310011–03100111. doi:10.1115/1.4000775.

[27] R. Khajehtourian, M. I. Hussein, Dispersion characteristics of a nonlinear elastic metamaterial, AIP Advances 4 (12) (2014). doi:10.1063/1.4905051.

[28] F. Bloch, Über die Quantenmechanik der Elektronen in Kristallgittern, Zeitschrift für Physik (1929). doi:10.1007/BF01339455.

[29] W. T. Thomson, Transmission of elastic waves through a stratified solid medium, Journal of Applied Physics 21 (2) (1950) 89–93. doi:10.1063/1.1699629.

[30] B. G. G. Chen, N. Upadhyaya, V. Vitelli, Nonlinear conduction via solitons in a topological mechanical insulator, Proceedings of the National Academy of Sciences of the United States of America 111 (36) (2014) 13004–13009. arXiv:1404.2263, doi:10.1073/pnas.1405969111.

[31] P. Wang, F. Casadei, S. Shan, J. C. Weaver, K. Bertoldi, Harnessing Buckling to Design Tunable Locally Resonant Acoustic Metamaterials, Physical Review Letters 113 (1) (2014) 014301. doi:10.1103/PhysRevLett.113.014301.

[32] K. H. Matlack, A. Bauhofer, S. Krödel, A. Palermo, C. Daraio, Composite 3D-printed metastructures for low-frequency and broadband vibration absorption, Proceedings of the National Academy of Sciences of the United States of America 113 (30) (2016) 8386–8390.

[33] S. Krödel, C. Daraio, Microlattice Metamaterials for Tailoring Ultrasonic Transmission with Elastoacoustic Hybridization, Physical Review Applied 6 (6) (2016) 064005. doi:10.1103/PhysRevApplied.6.064005.

[34] I. Arretche, K. H. Matlack, On the interrelationship between static and vibration mitigation properties of architected metastructures, Frontiers in Materials 5 (November) (2018) 1–16. doi:10.3389/fmats.2018.00068.

[35] B. Deng, P. Wang, Q. He, V. Tournat, K. Bertoldi, Metamaterials with amplitude gaps for elastic solitons, Nature Communications 9 (1) (2018) 3410. doi:10.1038/s41467-018-05908-9.

[36] B. Deng, C. Mo, V. Tournat, K. Bertoldi, J. R. Raney, Focusing and Mode Separation of Elastic Vector Solitons in a 2D Soft Mechanical Metamaterial, Physical Review Letters 123 (2) (2019) 024101. doi:10.1103/PhysRevLett.123.024101.

[37] X. Fang, J. Wen, D. Yu, G. Huang, J. Yin, Wave propagation in a nonlinear acoustic metamaterial beam considering third harmonic generation, New Journal of Physics 20 (12) (2018) 123028. doi:10.1088/1367-2630/aaf65e.

[38] J. Meaud, Nonlinear wave propagation and dynamic reconfiguration in two-dimensional lattices with bistable elements, Journal of Sound and Vibration 473 (2020). doi:10.1016/j.jsv.2020.115239.

[39] R. Chaunsali, E. Kim, J. Yang, Demonstration of accelerating and decelerating nonlinear impulse waves in functionally graded granular chains, Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences (2018). doi:10.1098/rsta.2017.0136.

[40] M. Hwang, A. F. Arrieta, Solitary waves in bistable lattices with stiffness grading: Augmenting propagation control, Physical Review E 98 (4) (2018) 042205. doi:10.1103/PhysRevE.98.042205.

[41] M. Serra-Garcia, M. Molerón, C. Daraio, Tunable, synchronized frequency down-conversion in magnetic lattices with defects, Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences 376 (2127) (2018) 20170137. doi:10.1098/rsta.2017.0137.

[42] V. Ramakrishnan, M. J. Frazier, Transition waves in multi-stable metamaterials with space-time modulated potentials, Applied Physics Letters 117 (15) (2020). doi:10.1063/5.0023472.

[43] M. Hwang, A. F. Arrieta, Extreme Frequency Conversion from Soliton Resonant Interactions, Physical Review Letters 126 (7) (2021) 073902. doi:10.1103/PhysRevLett.126.073902.

[44] D. Wagg, S. Neild (Eds.), Nonlinear Vibration Phenomena, Springer Netherlands, Dordrecht, 2010, pp. 35–80. doi:10.1007/978-90-481-2837-2_2.

[45] A. S. Phani, J. Woodhouse, N. A. Fleck, Wave propagation in two-dimensional periodic lattices, The Journal of the Acoustical Society of America 119 (4) (2006) 1995–2005. doi:10.1121/1.2179748.

[46] M. I. Hussein, Reduced Bloch Mode Expansion for Periodic Media Band Structure Calculations, Proceedings: Mathematical, Physical and Engineering Sciences 465 (2109) (2009) 2825–2848.

[47] A. Cebrecos, D. Krattiger, V. J. Sánchez-Morcillo, V. Romero-García, M. I. Hussein, The finite-element time-domain method for elastic band-structure calculations, Computer Physics Communications 238 (2019) 77–87. doi:10.1016/j.cpc.2018.12.016.

[48] B. Luo, S. Gao, J. Liu, Y. Mao, Y. Li, X. Liu, Non-reciprocal wave propagation in one-dimensional nonlinear periodic structures, AIP Advances 8 (1) (2018). doi:10.1063/1.5010990.

[49] D. Bitar, N. Kacem, N. Bouhaddi, Investigation of modal interactions and their effects on the nonlinear dynamics of a periodic coupled pendulums chain, International Journal of Mechanical Sciences 127 (December 2016) (2017) 130–141. doi:10.1016/j.ijmecsci.2016.11.030.

[50] R. Ganesh, S. Gonella, Nonlinear waves in lattice materials: Adaptively augmented directivity and functionality enhancement by modal mixing, Journal of the Mechanics and Physics of Solids 99 (August 2016) (2017) 272–288. arXiv:1608.01028, doi:10.1016/j.jmps.2016.11.001.

[51] M. Remoissenet, Waves Called Solitons, 3rd Edition, Advanced Texts in Physics, Springer Berlin Heidelberg, Berlin, Heidelberg, 1999. doi:10.1007/978-3-662-03790-4.

[52] T. Dauxois, M. Peyrard, Physics of solitons, Cambridge University Press, 2006.

[53] K. Ito, S. S. Ravindran, A Reduced-Order Method for Simulation and Control of Fluid Flows, Journal of Computational Physics 143 (2) (1998) 403–425. doi:10.1006/jcph.1998.5943.

[54] D. J. Lucia, P. S. Beran, W. A. Silva, Reduced-order modeling: New approaches for computational physics, Progress in Aerospace Sciences 40 (1-2) (2004) 51–117. doi:10.1016/j.paerosci.2003.12.001.

[55] T. Lieu, C. Farhat, M. Lesoinne, Reduced-order fluid/structure modeling of a complete aircraft configuration, Computer Methods in Applied Mechanics and Engineering 195 (41-43) (2006) 5730–5742. doi:10.1016/j.cma.2005.08.026.

[56] J. F. Currie, S. E. Trullinger, A. R. Bishop, J. A. Krumhansl, Numerical simulation of sine-Gordon soliton dynamics in the presence of perturbations, Physical Review B 15 (12) (1977) 5567–5580. doi:10.1103/PhysRevB.15.5567.

[57] M. Peyrard, M. D. Kruskal, Kink dynamics in the highly discrete sine-Gordon system, Physica D: Nonlinear Phenomena 14 (1) (1984) 88–102. doi:10.1016/0167-2789(84)90006-X.

[58] P. Biller, F. Petruccione, Dynamics of sine-Gordon solitons under random perturbations: Weak additive large-scale white noise, Physical Review B 41 (4) (1990) 2139–2144. doi:10.1103/PhysRevB.41.2139.

[59] V. V. Konotop, A. Sánchez, L. Vázquez, Kink dynamics in the weakly stochastic ${\mathrm{\ensuremath{\varphi}}}ˆ{4}$ model, Physical Review B 44 (6) (1991) 2554–2566. doi:10.1103/PhysRevB.44.2554.

[60] Zhang Fei, Y. S. Kivshar, L. Vázquez, Resonant kink-impurity interactions in the sine-Gordon model, Physical Review A 45 (8) (1992) 6019–6030. doi:10.1103/PhysRevA.45.6019.

[61] D. Steinkraus, I. Buck, P. Y. Simard, Using GPUs for machine learning algorithms, Proceedings of the International Conference on Document Analysis and Recognition, ICDAR 2005 (2005) 1115–1120. doi:10.1109/ICDAR.2005.251.

[62] J. D. Owens, M. Houston, D. Luebke, S. Green, J. E. Stone, J. C. Phillips, GPU computing, Proceedings of the IEEE 96 (5) (2008) 879–899. doi:10.1109/JPROC.2008.917757.

[63] J. Nickolls, W. J. Dally, The GPU computing era, IEEE Micro 30 (2) (2010) 56–69. doi:10.1109/MM.2010.41.

[64] J. D. Lambert, Numerical Methods for Ordinary Differential Systems: The Initial Value Problem, John Wiley & Sons, Inc., USA, 1991.

[65] A. Chopra, Dynamics of Structures: Theory and Applications to Earthquake Engineering, 4th Edition, Prentice Hall, 2012.

[66] M. J. Quinn, Parallel Programming in C with MPI and OpenMP, McGraw-Hill Education Group, 2003.

[67] H. M. Hilber, T. J. R. Hughes, R. L. Taylor, Improved numerical dissipation for time integration algorithms in structural dynamics, Earthquake Engineering & Structural Dynamics 5 (3) (1977) 283–292. doi:10.1002/eqe.4290050306.

[68] G. M. Amdahl, Validity of the single processor approach to achieving large scale computing capabilities, in: Proceedings of the April 18-20, 1967, spring joint computer conference on - AFIPS '67 (Spring), ACM Press, New York, New York, USA, 1967, p. 483. doi:10.1145/1465482.1465560.

[69] J. L. Gustafson, Reevaluating Amdahl's law, Communications of the ACM 31 (5) (1988) 532–533. doi:10.1145/42411.42415.

[70] C. Scalo, J. Bodart, S. K. Lele, Compressible turbulent channel flow with impedance boundary conditions, Physics of Fluids 27 (3) (2015) 035107. doi:10.1063/1.4914099.

[71] V. C. B. Sousa, D. Patel, J.-B. Chapelier, V. Wartemann, A. Wagner, C. Scalo, Numerical Investigation of Second-Mode Attenuation over Carbon/Carbon Porous Surfaces, Journal of Spacecraft and Rockets 56 (2) (2019) 319–332. doi:10.2514/1.A34294.

[72] Y. Chen, C. Scalo, Trapped waves in supersonic and hypersonic turbulent channel flow over porous walls, Journal of Fluid Mechanics 920 (Aug. 2021). doi:10.1017/jfm.2021.428.