SoftwareX 11 (2020) 100485

Contents lists available at ScienceDirect

SoftwareX

journal homepage: www.elsevier.com/locate/softx

Original software publication

CU-BENs: A structural modeling finite element library

Wensi Wu [a,*], Justyna Kosianka [b], Heather Reed [c], Christopher Stull [d], Christopher Earls [a]

[a] Cornell University, Ithaca, NY, United States
[b] Ursa Space Systems, Ithaca, NY, United States
[c] ASAPP, New York, NY, United States
[d] Los Alamos National Laboratory, Los Alamos, NM, United States

ARTICLE INFO

Article history:
Received 23 January 2020
Received in revised form 19 April 2020
Accepted 22 April 2020

Keywords:
Finite element library
Computational structural dynamics
Nonlinear structural analysis

ABSTRACT

The present work discusses capabilities within the finite element library CU-BENs. CU-BENs focuses on applying the finite element method to structural mechanics problems encountered within the context of inverse problems and partitioned fluid–structure interaction; thus element formulations are primarily of a structural type — truss, frame, and triangular discrete Kirchhoff theory shells. CU-BENs defaults to the skyline sparse storage scheme for the system matrix, but also supports other storage schemes when using external libraries such as LAPACK and UMFPACK. CU-BENs includes built-in nonlinear solution strategies, such as the Newton Raphson method and modified spherical arc length method, that are available within static analyses as well within the context of transient dynamic analyses involving a generalized-α implementation of the Newmark implicit time integration scheme.

Code metadata

Current code version	V4.1.2
Permanent link to code/repository used for this code version	https://github.com/ElsevierSoftwareX/SOFTX_2020_13
Legal Code License	GPL 3.0
Code versioning system used	Git
Software code languages, tools, and services used	C
Compilation requirements, operating environments & dependencies	BLAS, LAPACK, UMFPACK
If available Link to developer documentation/manual	https://github.com/nonlinearfun/CU-BENs/blob/master/Introduction%20to%20CU-BEN.pdf
Support email for questions	earls@cornell.edu

1. Motivation and significance

Continuum mechanics is the foundation of scientific discoveries in a rich variety of areas: structural mechanics, aerodynamics, hydrodynamics, biomechanics, *etc.* It uses continuous mathematical descriptions to predict the motion and deformation of a physical system, so that the natural phenomena represented by the system can be better understood. However, the mathematical descriptions are often a set of partial differential equations (PDEs) for which the analytical solutions are difficult to obtain. Owing to this challenge, robust numerical methods are needed in order to further advance discoveries in science and engineering.

The finite element method (FEM) was implemented within CU-BENs for our research purposes because of its solid mathematical foundations for variational initial boundary value problems encountered in nonlinear structural mechanics [1]. The capabilities within CU-BENs enable an analyst to carry out many structural mechanics simulations (*i.e.* calls to a forward solver), as is frequently required when solving model-based inverse problems, or tackling implicit, partitioned coupled analyses within a fluid–structure interaction (FSI) modeling context. One example application from our own work, where the unique capabilities of CU-BENs have been instrumental in enhancing phenomenological understanding, is the context of next generation hull form design under hydrodynamic slamming loading; frequently occurring in high-speed watercraft as they "porpoise" in and out of the water. These loads, occurring when the structure of the watercraft leaves

* Corresponding author.
E-mail address: ww382@cornell.edu (W. Wu).

https://doi.org/10.1016/j.softx.2020.100485

the water and then subsequently impacts upon the free surface as it plunges back into the water, induce high strain rate material effects, along with activating nonlinear structural responses. These structural responses may greatly affect the safety and performance of high-speed watercraft. In order to analyze the nonlinear behaviors within such a computational context, proper numerical methods are required (*e.g.* implicitly coupled partitioned FSI simulations). CU-BENs has been effectively used as the computational structural dynamics (CSD) solver in such cases. Given its structural focus, CU-BENs implements the most commonly used methods in nonlinear structural mechanics, including the Newton Raphson [2], modified spherical arc length [3], and Newmark time integration methods [4,5]. The focus CU-BENs places on higher order, nonlinear structural element formulations, used in conjunction with force-space plasticity models (described in the sequel), permits efficient, repeated CSD analyses to be carried out, as encountered when solving model-based inverse problems (frequently involving 10^6 calls to the CSD forward solver) and within the context of partitioned FSI analyses involving implicit coupling (*e.g.* when added mass effects are important). Additionally, the default, rudimentary input/output system (*i.e.* text based, non-GUI) is organized and tailored for inverse problems, which allows CU-BENs to communicate easily with other codes (though utilities are available in the software repository for using Abaqus CAE for graphical preprocessing and ParaView for graphical post processing). A few examples of this kind of work includes Reed' s damage identification [6,7], as well as structural health monitoring of naval ship hulls by Stull et al. [8]. In the future, CU-BENs will be used to analyze the impact of slamming loads on ship hulls.

The outline of the current paper is as follows: Section 2 discusses the architecture and functionalities within CU-BENs; Section 3 presents a selected, simple verification result illustrating the generalized-α method that is followed by a more involved analysis example of an entire ship hull; and Section 4 discusses the major impacts of CU-BENs.

2. Software description

CU-BENs is a 3D implicit CSD finite element (FE) solver, written with the intent of creating a fast, efficient, and accurate structural modeling software system for linear and nonlinear, static and dynamic FE analysis. It also includes a linear, acoustic fluid–structure interaction (FSI) capability [9]. While many general purpose FEM software such as deal.II [10], FEniCS [11], DUNE [12], *etc.*, are available for use, CU-BENs is an FEM library that specializes within the domain of structural mechanics. From a usability perspective, the specialization of CU-BENs requires fewer dependencies and provides a more coherent user interface compared to the other FEM libraries mentioned; thus making it more accessible for application scientists. From a computational mechanics perspective, the use of structural elements allows for substantial dimensional reduction in solving structural mechanics problems; thus, computational efficiency improves considerably when CU-BENs is called repeatedly: *e.g.* when solving inverse problems. The previously cited, and excellent, FE libraries are sophisticated, general purpose tools that are, however, less capable within the specific context of CSD; thus CU-BENs meets an important need within the computational science and engineering (CSE) community.

CU-BENs exploits higher-order nonlinear elements to permit greater solution accuracy per degree of freedom (DOF), within structural dynamic computations, as compared with more general solid continuum elements. In this spirit, CU-BENs supports truss, frame, and discrete Kirchhoff theory (DKT) shell elements. Additionally, CU-BENs contains a built-in acoustic FSI analysis capability (formulation details can be found in [13]) that is useful, for example, when modeling SONAR effects for predicting acoustic signatures of vessels. All elements, except for the truss, are formulated using the updated Lagrangian referential description [4] under large displacement, small strain assumptions. The truss element is formulated using the total Lagrangian approach [4] under large displacement, large strain assumptions.

The formulation of the system (stiffness) matrix takes advantage of the skyline sparse storage scheme to enhance analysis speed and improve storage efficiency. As opposed to a more common sparse matrix storage scheme (*e.g.* Compressed Sparse Row storage), the skyline structure is preserved during Gaussian elimination when pivoting is not needed [14]. *Importantly, the skyline storage scheme provides a convenient matrix structure for the implementation of the prescribed displacement feature within CU-BENs (to reduce the need for secondary data structures)*. It is pointed out that CU-BENs also supports linear system solution using LAPACK and UMFPACK (the latter being of particular use within acoustic FSI contexts where the monolithic partitioning scheme implemented within CU-BENs results in a sparse, non-symmetric system coefficient matrix).

CU-BENs consists of multiple, well commented source files that contain all element description functions and supporting functions for FEM analyses, as well as a typical header file containing all the necessary declarations. The element description functions and supporting functions are called in a main driver function that is organized based on a user defined input analysis type and solution algorithm type. Differing from most available FE libraries that address solid and structural mechanics problems, the treatment of material nonlinearity during static nonlinear transient dynamic analyses within CU-BENs employs the force-space plasticity method [1,15]. Meaning, instead of defining a failure surface parametrically (*e.g.* von Mises) using principal stresses (*i.e.* stress-space plasticity), the force-space plasticity method defines a failure surface within stress-resultant space — using forces and moments (*i.e.* the directly modeled actions within the analysis). Axial force, moment about the major principal centroidal axis, and moment about the minor principal centroidal axis are the independent variables used to construct the force-space parametric structure of the failure surface employed. Just as in an associated flow, stress-space plasticity model, the force-space model employs an analogous framework for plastic loading, elastic unloading, and constrained plastic flow [1,15]. The implemented force-space approach leads to a considerable speed up over stress-space approaches; thus resulting in a much more efficient forward solver within the context for inverse problems involving structures, *etc.*

CU-BENs is hosted on GitHub and can be downloaded at, https://github.com/nonlinearfun/CU-BENs/releases. A detailed description of the analysis types, solution methods, and element types as well as the theory behind the finite element formulation can be found in the tutorial and theory manual, *Introduction to CU-BEN* and *CUBENs_theory_manual*, under the CU-BENs repository.

2.1. Software architecture

The sources files for CU-BENs are divided up into three categories: pre/post-processing, element descriptions, and supporting functions. A schematic of the architecture for CU-BENs is provided in Fig. 1. CU-BENs is extensively documented inline — every block of code includes detailed description of the theory and logic behind the implementation. In addition, one could refer to the header file, prototypes.h, to explore the intended use of each function within CU-BENs.

The input file, model_def.txt, provides CU-BENs with necessary element information and analysis instructions for execution. Detailed instructions on how to create an input file and compile CU-BENs can be found in Sections 4 and 5 of the *Introduction to CU-BEN* (available within the software repository).

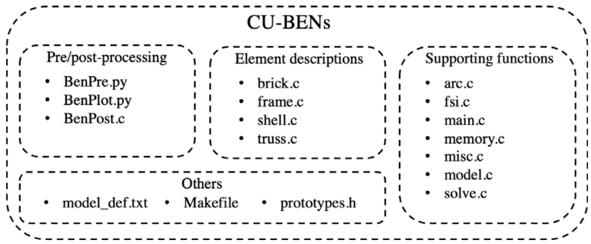

Fig. 1. Schematic of the CU-BENs architecture.

2.2. Software functionalities

CU-BENs supports both static and transient dynamic analyses. The solution of the resulting algebraic system is limited to direct solvers: a built-in LDL^T solver, along with capability to interface with LAPACK and UMFPACK (when the systems are sparse and non-symmetrical, such as within the acoustic FSI contexts). For dynamic analysis, CU-BENs supports Newmark implicit time integration, generalized-α numerical dissipation, automatic time stepping, and prescribed, rigid body motion. Aside from the numerical methods mentioned, CU-BENs also has built-in functions such as the pre/post processing and checkpoint/restart subroutines to improve usability and computational efficiency. The motivation behind the implementation of generalized-α numerical dissipation is now briefly discussed.

2.2.1. Generalized-α numerical dissipation method

Although the Newmark implicit time integration scheme is provably unconditionally stable within linear elastic transient dynamic systems, this behavior is often not enjoyed within nonlinear dynamic systems due to unresolved high frequency responses, and complex interactions between fluid and structural interfaces involving added mass effects. This motivates the implementation of the generalized-α numerical dissipation method [16]. This method relaxes the exact conservation of energy in the dynamic system by introducing numerical dissipation (*i.e.* artificial damping) in high frequency structural modes within the solution, such that *numerical stability* is achieved. The intensity of the dissipation is controlled by the spectral radius of the amplification matrix; a smaller spectral radius values correspond to higher numerical dissipation. One may refer to Chung [16] for the theory and implementation details.

3. Illustrative examples

3.1. Cantilever beam subjected to uniformly distributed pressure pulse

A cantilever beam test problem proposed by Bathe [17] is subjected to a uniformly distributed pressure pulse spanning 0.04 s as shown in Fig. 2. The cantilever beam is 0.4 m long and 0.1 cm thick with a Young's modulus of 70 GPa, Poisson ratio of 0.33, and mass density of 2700 kg/m^3. Numerical results from CU-BENs are compared against ADINA [18].

Under this loading condition, the structure undergoes large displacements and subsequent oscillation. A standard Newmark time integration scheme is insufficient to produce a stable solution, as numerical errors accumulate significantly — this leads to physically unrealistic accelerations, velocities, and displacements in the structural response.

Bathe proposed a two sub-step composite scheme [17] to mitigate this issue. Fig. 3 shows a comparison of the structural responses calculated through the generalized-α numerical dissipation method within CU-BENs and Bathe's composite scheme within ADINA. Both methods introduce numerical damping in the structural system. The former introduces numerical damping as a scaling factor within the structural modes, while the latter introduces damping through a second sub-time step computation.

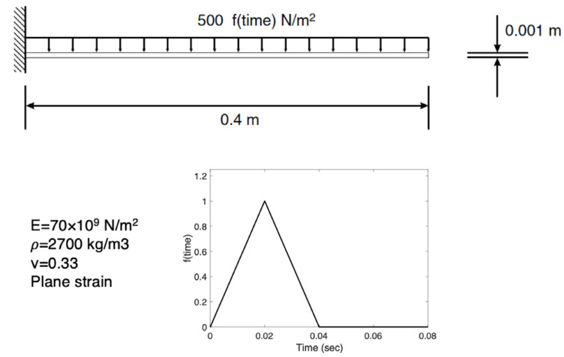

Fig. 2. Cantilever beam subjected to uniformly distributed pulse [17].

The cantilever beam, following the geometry described in Fig. 2, is modeled with 80 DKT shell elements in CU-BENs and a 400 × 1 mesh of nine-node solid elements in ADINA. The total simulation time of the analysis is 2 s with Δt= 0.002 s A spectral radius of 0.95 is applied in CU-BENs to introduce a small amount of numerical damping in the acceleration, velocity, and displacement responses. As shown in Fig. 3(a), the structural responses of the cantilever beam is unbounded when using Newmark's average acceleration scheme. In Fig. 3(b), it is shown that the predicted responses using generalized-α method demonstrates excellent agreement with Bathe's composite scheme.

3.2. JHSS subjected to axial pressure pulse

As a second example, we present the analysis of an entire ship hull (JHSS). In this particular case, we are validating the CU-BENs model using experimental, model scale testing in a model basin. It is pointed out that the complexity of the tow tank model is at least comparable to the complexity encountered within an actual ship structure. A rendering of a solid model fit to the underlying JHSS mesh is illustrated in Fig. 4, along with the pressure pulse loading imposed on the stern (to simulate power plant vibration). The model consists of 45,846 degrees of freedom (DOFs), constrained in rotations as well as y- and z- translations. Nonlinear pressure pulses are applied at the stern. The total simulation time is 0.012 s with Δt= 0.0004 s While the model could not fit into 64GB of RAM with a dense matrix storage scheme, the JHSS model occupies 1GB of RAM using SKYLINE storage, and runs in 2.5 min with 30 implicit Newmark time steps on a MacBook Pro laptop (2.5 GHz Intel Core i7 processor).

4. Impact

CU-BENs is an established computational structural dynamics (CSD) FE software tool that can be used to solve a wide variety of structural engineering problems. CU-BENs is built using structural finite element technology, as opposed to using general continuum solid elements; thus enabling users to access more sophisticated element types — permitting fewer degrees-of-freedom without compromising solution accuracy. The implemented higher order nonlinear element formulations support a force-space plasticity approach to nonlinear material modeling. The resulting structural analysis capability in CU-BENs allows for efficient and repeated calls, as required in model-based inverse problems and partitioned FSI contexts where implicit coupling is important.

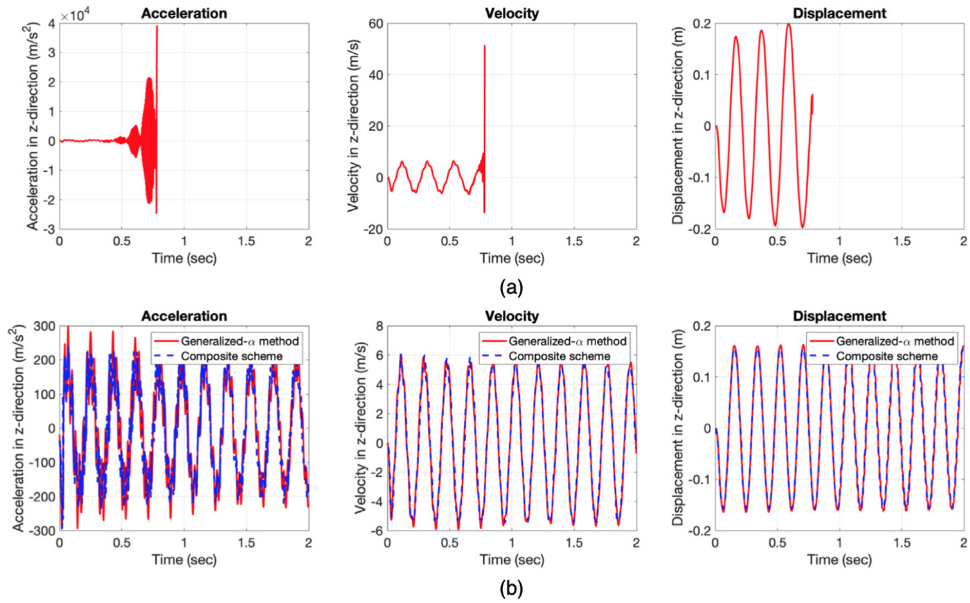

Fig. 3. (a) Tip acceleration, velocity, and displacement of the cantilever beam using standard Newmark time integration (average acceleration) scheme. (b) Tip acceleration, velocity, and displacement of the cantilever beam using generalized-α numerical dissipation method and two sub-step composite scheme.

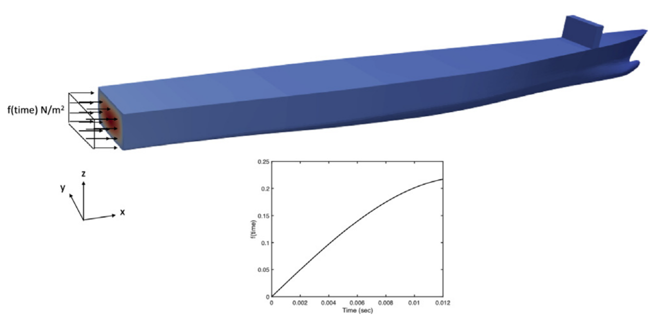

Fig. 4. JHSS subjected to nonlinear pressure pulse.

As interest in solving fluid–structure interaction problems has grown tremendously over the past decade, researchers have been developing FSI software tools using either "monolithic" or "partitioned" approaches [19]. In the latter case, *numerical stability* can be a concern within partitioned approaches where the fluid and structure regions are employing different discretization techniques (*i.e.* finite volumes versus finite elements) and when the density ratio of the solid phase to the liquid phase approaches unity. Therefore, numerical dissipation schemes are required (as well as potential adaptations to spatial discretization) in practice, to assist in ameliorating spurious high frequency responses within implicitly coupled, partitioned FSI simulations where added mass effects are important. The generalized-α scheme implemented within CU-BENs is of important, practical benefit in such cases.

5. Conclusions

CU-BENs is a fast, efficient, and accurate FE package for structural analysis, with built-in numerical algorithms ready for linear and nonlinear, static and transient dynamic FE analyses. All functions implemented in CU-BENs have been rigorously verified against other software, as well as closed-form solutions; in addition to some experimental validations that have also been carried out. CU-BENs has been applied to different engineering problems, such as acoustic FSI and hull structure health modeling, *etc.*, across multiple research projects by the Authors. The computational efficiency of CU-BENs is ideally suited to research projects involving model-based inverse problems involving structures, as well as structural modeling contexts where uncertainty quantification is desired (*e.g.* Markov chain Monte Carlo analysis using structural forward models within the likelihood function). The use of CU-BENs is growing, and the library will be continuously expanded and advanced in accordance with future research needs.

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.

Acknowledgment

The development of CU-BENs was supported under multiple grants by the Office of Naval Research, United States, Division 331, Ship Structural Reliability Program.

References

[1] Earls C. CUBENs_theory_manual. 2016, https://github.com/nonlinearfun/CU-BENs/blob/master/CUBENs_Theory_Manual.pdf.

[2] Chapra S, Canale R. Numerical methods for engineers. Boston: McGraw-Hill Higher Education; 2010.

[3] Riks E. An incremental approach to the solution of snapping and buckling problems. Int J Solids Struct 1979;15(7):529–51. http://dx.doi.org/10.1016/0020-7683(79)90081-7, http://www.sciencedirect.com/science/article/pii/0020768379900817.

[4] Bathe K, Saunders H. Finite element procedures. Prentice Hall International; 1996.

[5] Crisfield M, Remmers J, Verhoosel C. Nonlinear finite element analysis of solids and structures. John Wiley & Sons; 2012.

[6] Reed H, Nichols J, Earls C. A modified differential evolution algorithm for damage identification in submerged shell structures. Mech Syst Signal Process 2013;39(1):396–408. http://dx.doi.org/10.1016/j.ymssp.2013.02.018.

[7] Reed H, Earls C, Nichols J. Stochastic identification of imperfections in a submerged shell structure. Comput Methods Appl Mech Engrg 2014;272:58–82. http://dx.doi.org/10.1016/j.cma.2014.01.003.

[8] Stull CJ, Earls CJ, Koutsourelakis P-S. Model-based structural health monitoring of naval ship hulls. Comput Methods Appl Mech Engrg 2011;200(9):1137–49. http://dx.doi.org/10.1016/j.cma.2010.11.018.

[9] Reed HM, Earls CJ, Nichols JM. Stochastic identification of imperfections in a submerged shell structure. Comput Methods Appl Mech Engrg 2014;272:58–82. http://dx.doi.org/10.1016/j.cma.2014.01.003, http://www.sciencedirect.com/science/article/pii/S0045782514000085.

[10] Arndt D, Bangerth W, Davydov D, Heister T, Heltai L, Kronbichler M, Maier M, Pelteret J-P, Turcksin B, Wells D. The deal. II finite element library: design, features, and insights. Comput Math Appl 2020. http://dx.doi.org/10.1016/j.camwa.2020.02.022, http://www.sciencedirect.com/science/article/pii/S0898122120300894.

[11] Alnæs MS, Blechta J, Hake J, Johansson A, Kehlet B, Logg A, Richardson C, Ring J, Rognes ME, Wells GN. The FEniCS project version 1.5. Arch Numer Softw 2015;3(100). http://dx.doi.org/10.11588/ans.2015.100.20553.

[12] Blatt M, Burchardt A, Dedner A, Engwer C, Fahlke J, Flemisch B, Gersbacher C, Gräser C, Gruber F, Grüninger C, Kempf D, Klöfkorn R, Malkmus T, Müthing S, Nolte M, Piatkowski M, Sander O. The distributed and unified numerics framework. Arch Numer Softw 2016;version 2.4. http://dx.doi.org/10.11588/ans.2016.100.26526.

[13] Everstine G. Finite element formulations of structural acoustics problems. Comput Struct 1997;65(3):307–21.

[14] Barrett R, Berry M, Chan TF, Demmel J, Donato J, Dongarra J, Eijkhout V, Pozo R, Romine C, der Vorst HV. Templates for the solution of linear systems: building blocks for iterative methods. Philadelphia, PA: SIAM; 1994.

[15] McGuire W, Gallagher R, Ziemian R. Matrix structural analysis. John Wiley & Sons; 1979.

[16] Chung J, Hulbert G. A time integration algorithm for structural dynamics with improved numerical dissipation: the generalized-α method. J Appl Mech 1993;60(2):371–5.

[17] Bathe KJ, Baig MMI. On a composite implicit time integration procedure for nonlinear dynamics. Comput Struct 2005;83(31):2513–24. http://dx.doi.org/10.1016/j.compstruc.2005.08.001.

[18] ADINA. version 9.5.1 Watertown (Massachusetts), http://www.adina.com/.

[19] Degroote J. Development of algorithms for the partitioned simulation of strongly coupled fluid-structure interaction problems [Ph.D. thesis], Ghent University; 2010.