Jiří Vala; Václav Rek

On a computational approach to multiple contacts / impacts of elastic bodies

In: Jan Chleboun and Pavel Kůs and Jan Papež and Miroslav Rozložník and Karel Segeth and Jakub Šístek (eds.): Programs and Algorithms of Numerical Mathematics, Proceedings of Seminar. Jablonec nad Nisou, June 19-24, 2022. Institute of Mathematics CAS, Prague, 2023. pp. 269–280.

Persistent URL: http://dml.cz/dmlcz/703207

Programs and Algorithms of Numerical Mathematics 21
J. Chleboun, P. Kůs, J. Papež, M. Rozložník, K. Segeth, J. Šístek (Eds.)
Institute of Mathematics CAS, Prague 2023

ON A COMPUTATIONAL APPROACH TO MULTIPLE CONTACTS / IMPACTS OF ELASTIC BODIES

Jiří Vala[1], Václav Rek[2]

[1] Brno University of Technology, Faculty of Civil Engineering
Institute of Mathematics and Descriptive Geometry
602 00 Brno, Veveří 95, Czech Republic
vala.j@fce.vutbr.cz
[2] Czech Technical University in Prague, Faculty of Transportation Sciences
Department of Mechanics and Materials
110 02 Prague, Konviktská 20, Czech Republic
vaclav.rek@cvut.cz

Abstract: The analysis of dynamic contacts / impacts of several deformable bodies belongs to both theoretically and computationally complicated problems, because of the presence of unpleasant nonlinearities and of the need of effective contact detection. This paper sketches how such difficulties can be overcome, at least for a model problem with several elastic bodies, using i) the explicit time-discretization scheme and ii) the finite element technique adopted to contact evaluations together with iii) the distributed computing platform. These considerations are supported by the references to useful generalizations, motivated by significant engineering applications. Illustrative examples demonstrate this approach on structures assembled from a finite number of shells.

Keywords: contact of elastic bodies, finite element method, finite difference method, distributed computing

MSC: 74M15, 74S05, 74S20, 68Q85

1. Introduction

Reliable computational prediction of the behaviour of deformable bodies under mechanical, thermal, etc. loads belongs to the priorities of both civil and mechanical engineering, due to the development of advanced materials, structures and technologies, whose traditional analysis, coming from long-time experience, certified laboratory measurements and heuristic computational formulae, is not available. Such computational prediction should come from the numerical analysis of initial and boundary value problems for systems of partial differential equations of evolution, based on the principles of classical thermomechanics by [3], namely in the form

DOI: 10.21136/panm.2022.25

of conservation of such scalar quantities as mass, (linear and angular) momentum components and energy, supplied by appropriate constitutive relations, whose parameters have to be identified by experiments. A significant task is the modelling and simulation of the rapid movement of several bodies with potential contacts and impacts, accompanied by their deformation: in addition to the incorporation of various geometrical and physical nonlinearities, the design of an effective algorithm needs e. g. some results from the graph theory and the distributed and parallel computing.

After these motivational comments (*Section* 1) we intend to present a model problem of multiple contacts / impacts of elastic (or viscoelastic) deformable bodies. The overview of physical and mathematical background (*Section* 2) will be followed by some details of the computational approach (*Section* 2), with special attention to the advanced search for potential contacts, using a distributed computing platform (*Section* 3). This will be demonstrated on two illustrative examples (*Section* 4) and supplied by brief concluding remarks with future research priorities (*Section* 5).

2. Physical and mathematical background

As a first model problem, let us consider a deformable body occupying a single domain Ω in the Euclidean space $\mathbb{R}^3$, supplied by a fixed Cartesian coordinate system $x = (x_1, x_2, x_3)$ for simplicity, with the Lipschitz boundary $\partial\Omega$, decomposed to disjoint parts Θ (for homogeneous Dirichlet boundary conditions) and Γ (for Neumann boundary conditions, inhomogeneous in general). The deformation of Ω will be analyzed on a finite time interval $\mathcal{I} = [0, T]$, T being a positive constant, i. e. for any time $t \in \mathcal{I}$. For any appropriate function ϕ we shall write $\phi_{,i}$ instead of $\partial\phi/\partial x_i$ with $i \in \{1, 2, 3\}$ and $\dot{\phi}$ instead of $\partial\phi/\partial t$ for brevity. The unit (formally outward) normal vector $\nu = (\nu_1, \nu_2, \nu_3)$ can be constructed (almost everywhere) on $\partial\Omega$. The standard notation of Lebesgue, Sobolev, Bochner - Sobolev, etc. function spaces following [24, Parts 1 and 7], will be applied here. The basic unknown variable $u(x, t)$, working with $x \in \Omega$ and $t \in \mathcal{I}$, introduced as the displacement of $x \in \Omega$, with possible extensions to Θ and Γ, in time $t \in \mathcal{I}$ related to the initial configuration at $t = 0$, can be considered as an element of $L^p(\mathcal{I}, V)$, with its first time derivative belonging to the same space and the second one (at least) to $L^2(\mathcal{I}, V^*)$. Here $V = \{w \in W^{1,p}(\Omega)^3 : \ w = o \text{ on } \Theta\}$ incorporates all body supports, V^* means the dual to V, o denotes the zero vector from $\mathbb{R}^3$ and $p, q \in [2, \infty)$ are some fixed exponents; satisfying $1/p + 1/q = 1$ (for all linearized formulations always $p = q = 2$). Let us notice that for any $w \in V$ we have (at least) $w \in L^6(\Omega)^3$, thanks to the Sobolev embedding theorem. Let us also introduce $X = L^q(\Omega)^3$ and $Z = L^q(\Gamma)^3$. To avoid technical difficulties, we shall make use of the results of [21, Parts 1.2 and 6.7], for elliptic (purely static) problems, referring to their natural generalization to hyperbolic (general dynamic) problems, thanks to the properties of Rothe sequences by [24, Part 7]. All detailed derivations must be left to the curious reader, due to the limited extent of this paper.

Let the pair of Cauchy initial conditions $u(0,.) = o$ and $\dot{u}(0,.) = \widehat{u}$ be introduced on Ω, $\widehat{u} \in V$ being some prescribed initial displacement rate. Let also the body forces $f \in L^q(\mathcal{I}, X)$ and the surface forces $g \in C_L(\mathcal{I}, Z)$ be given a priori, C_L referring to Lipschitz continuous functions on $\mathcal{I}$ (to avoid the difficulties with the properties of traces from V on $\partial\Omega$). Let $i, j, k \in \{1, 2, 3\}$ be the Einstein summation indices. Then the weak formulation of the conservation of linear momentum reads

$$(w_i, \rho\ddot{u}_i) + (w_{k,i}, \tau_{ik} + \alpha\dot{\tau}_{ik}) = (w_i, f_i) + \langle w_i, g_i\rangle \tag{1}$$

on $\mathcal{I}$ for any test function (virtual displacement) $w \in V$, However, the Piola stress tensor $\tau \in L^q(\Omega)^{3\times3}$ in (1) is still undefined and must be evaluated from an appropriate constitutive relation. Most frequently such relation uses the stress - strain dependence between the symmetric Kirchhoff stress tensor σ (its symmetry can be justified from the conservation of angular momentum, under the usual assumptions on Boltzmann continuum) introduced as $\tau_{ik} = \sigma_{ij}(\delta_{kj} + u_{k,j})$, with the help of the Kronecker symbol δ, and the Almansi strain tensor $\varepsilon_{ik}(u) = (u_{i,k} + u_{k,i} + u_{j,i}u_{j,k})/2$. Moreover, for appropriate functions φ and $\widetilde{\varphi}$, $(\varphi, \widetilde{\varphi})$ in (1) means the Lebesgue integral of $\varphi\widetilde{\varphi}$ over Ω and $\langle\varphi, \widetilde{\varphi}\rangle$ the similar Hausdorff integral over Γ; for $p = q = 2$ we can identify $(\varphi_i, \widetilde{\varphi}_i)$ and $\langle\varphi_i, \widetilde{\varphi}_i\rangle$ just with scalar products on X and Z. In (1) new positive material characteristics occur: $\rho \in L^\infty(\Omega)$ is the material density and $\alpha \in L^\infty(\Omega)$ introduces the structural damping factor, taking certain energy dissipation into account (because no closed physical systems occur in real applications).

The crucial choice for the practical implementation of (1) is the evaluation of σ from ε. Here we shall present only the empirical Hooke law for the isotropic case

$$\sigma_{ij} = \partial\Psi(\varepsilon)/\partial\varepsilon_{ij}\,, \qquad \Psi(\varepsilon) = \lambda_1\varepsilon_{kk}^2/2 + \lambda_2\varepsilon_{ij}\varepsilon_{ij}\,, \tag{2}$$

containing just two positive Lamé factors $\lambda_1, \lambda_2 \in L^\infty(\Omega)$ (or the Young modulus and the Poisson coefficient, derivable from them easily). Admitting the material anisotropy, most of our considerations with a generalized stored-energy function Ψ could be repeated, but with the duty to work with (up to) 21 independent material characteristics on Ω instead of two Lamé factors; the same can be valid even for a wider class of Ψ, introduced carefully, as discussed by [4]. Clearly the positive values of α on Ω in (1) upgrade this formulation to the parallel viscoelastic Kelvin model.

Unfortunately, the full procedure of verification of the existence and uniqueness of u satisfying (1) including (2), due to both Cauchy initial conditions, is not straightforward. For the time steps $t = sh$ with $s \in \{1, \ldots, m\}$, $h = T/m$, with the aim $m \to \infty$ in all convergence considerations, we are allowed to search for some $u_s \in V$ instead of the unknown $u(., sh)$ understanding $\tau(u_s)$ as the approximation of $\tau(u(., sh))$ by (2). Replacing $\dot{u}$ and $\ddot{u}$ by the first and second relative differences $\mathcal{D}u_s = (u_s - u_{s-1})/h$ and $\mathcal{D}^2u_s = (\mathcal{D}u_s - \mathcal{D}u_{s-1})/h$, (1) can be rewritten in the form

$$(w_i, \rho\mathcal{D}^2u_{is}) + (w_{k,i}, \tau_{iks} + \alpha\mathcal{D}\tau_{iks}) = (w_i, f_{is}) + \langle w_i, g_{is}\rangle \tag{3}$$

for any $w \in V$ again; f_{is} and g_{is} can be taken e. g. as the Clément quasi-interpolations of the components of f and g by [24, Part 7], together with $u_0 = o$ and $u_{-1} = -h\widehat{u}$. Thus we come, step by step, to some particular nonlinear elliptic equations, which should be solved iteratively, generating several types of Rothe sequences constructed i) as linear Lagrange splines on $\mathcal{I}$ using the values u_{is} and ii) as simple (piecewise constant) abstract functions using the same values, and iii) as time-retarded modifications of i) and ii) (to cover semi-linearization in iterative processes), whose convergence to u in a reasonable sense can be expected. However, the theoretical analysis of these equations needs some assumptions on polyconvexity (or quasiconvexity, etc.) for Ψ on Ω, together with the guarantee of mutual impenetrability of parts of Γ, which must be seen as nontrivial problems beyond the scope of this paper. Since all Rothe sequences i), ii), iii) are defined in infinite-dimensional function spaces, a finite element (or similar) technique is needed for most numerical evaluations.

As a second model problem, let us consider Ω as a union of a finite number of deformable bodies, whose frictionless contact is allowed now. Therefore three parts of $\partial\Omega$ must be distinguished in any time $t \in \mathcal{I}$, namely Θ, Γ and Λ where $\Lambda \subset \Gamma$ refers to all internal, adaptively activated interfaces; the lower index $_*$ will identify the integration over Λ (instead of Γ), the square brackets will be used for interface jumps of function values on Λ. Consequently (1) gets the form

$$(w_i, \rho\ddot{u}_i) + (w_{k,i}, \tau_{ik} + \alpha\dot{\tau}_{ik}) = (w_i, f_i) + \langle w_i, g_i\rangle + \langle [w_i\nu_i], \mathcal{T}\rangle_* \tag{4}$$

on $\mathcal{I}$ for any $v \in V$, containing the interface tractions $\mathcal{T} \in L^q(\Lambda)^3$ (not prescribed explicitly) replacing g_i on Γ by $g_i + \mathcal{T}\nu_i$ on Λ where $[u_i\nu_i] = 0$ is required. The activation and deactivation of Λ can be explained as the conversion of (4) to certain variational inequality of the Hertz - Signorini - Moreau type, as demonstrated by [33]. In some more details: in practical calculations, working with $u_\nu = u_i\nu_i$, we need $[u_i\nu_i]\,\mathcal{T} \le 0$ for all potential contacts (including both Λ and some adjacent parts of Γ) where always i) $[u_\nu] = 0$ and $\mathcal{T} \le 0$ (on Λ) or ii) $\mathcal{T} = 0$ and $[u_\nu] \le 0$ (outside Λ).

Following still [33], such formulation can be handled without the application of explicit inequalities, using the penalty approach. This approach admits some (sufficiently small) impacts, characterized by a positive part of $[u_\nu]_+$, suppressed by an artificial stiffness $\mathcal{K} \to \infty$ (constant frequently), occurring in one additional constitutive equation $\tau = \mathcal{K}[u_\nu]_+$. However, namely the searching for potential couples for all evaluations $[u_\nu]$ in arbitrary time $t \in \mathcal{I}$ can be seen as a serious numerical problem, exceeding the set of usual numerical methods for the analysis of differential equations, tending to the implementation of an appropriated distributed computing platform. Nevertheless, repeating the approach for a first model problem formally, (2) can be applied without any change and (3) has to be enriched by one right-hand-side additive term $\langle [w_{is}\nu_i], \mathcal{T}_s\rangle_*$ only.

3. Computational approach

Unfortunately the computational algorithm induced by the generalized version of (1) (including its above sketched generalization), applied to the study of convergence of Rothe sequences successfully, is not optimal for practical calculations. Thus we will sketch the derivation of a simple (nearly) explicit concurrent algorithm, up to the evaluation in certain finite-dimensional space. First, let us notice the unpleasant evaluations of sufficiently large discretized approximate values of u related to the reference configuration of Ω at $t = 0$; in this case the simple remedy is some adaptive setting of a new reference configuration after certain number of time steps, using some a posteriori estimates, relevant for the time development of Ω. The following task is then the full discretization of (1). For simplicity, as usual in the finite element method, let us consider some (at least weakly) regular decomposition of Ω to finite elements, using a set of n basis functions (with a small compact support, as derived e. g. from linear 3-dimensional Lagrange splines) $\{\phi_1, \dots, \phi_n\}$ from an n-dimensional space V^n approximating V (in particular, for conforming finite elements, from such subspace of V). We shall use the notation $\mathfrak{D}$ for a norm of such decomposition, e. g. that introduced as the largest diameter of a ball containing all applied finite elements, too; $\mathfrak{D} \to 0$ with $n \to \infty$ is expected.

Let us try to express $u(., t)$ at $t = sh$ by (3), using one more Einstein summation index $r \in \{1, \dots, n\}$ in its form $u_{is} = \mathbb{U}_{irs}\phi_r$ where $i \in \{1, 2, 3\}$ and $s \in \{1, \dots, m\}$ such that $\mathbb{U}_{irs}$ are, for simplicity, just the values approximating $u_i(x_r, sh)$ in some selected points x_r from Ω and Γ (including Λ); thus $\phi_r = 1$ for $x = x_r$, being zero-valued in all remaining cases. Thus the test functions are allowed to be $w_j = \mathbb{W}_{jr}\phi_r$ where $j \in \{1, 2, 3\}$, just with one non-zero value $\mathbb{W}_{jr}$ equal to 1. As the result we can compose the explicit time integration scheme, inspired by [11], in the form of a system of $3n$ seemingly linear algebraic equations

$$\mathbb{MA}_s = h^2\mathbb{F}_s + (h^2/\mathfrak{D})\mathbb{G}_s + (h^2/\mathfrak{D})\widetilde{\mathbb{G}}_s([\mathbb{U}_s])) - (h/\mathfrak{D}^2)\mathbb{C}(\mathbb{V}_s^\times) - (h^2/\mathfrak{D}^2)\mathbb{K}(\mathbb{U}_s)\,, \quad (5)$$

for $s \in \{0, 1, \dots, m\}$ supplied by the auxiliary formulae

$$\mathbb{V}_{s+1/2} = \mathbb{V}_{s-1/2} + h\mathbb{A}_s\,, \qquad \mathbb{V}_s = (\mathbb{V}_{s-1/2} + \mathbb{V}_{s+1/2})/2, \qquad \mathbb{U}_{s+1} = \mathbb{U}_s + h\mathbb{V}_{s+1/2} \quad (6)$$

(for $s = m$ without the last one) where $\mathbb{M}$ is a positive definite real symmetric sparse matrix of order $3n$ (or even a diagonal one, using the well-known lumped mass trick, working with the replacement of $\{\phi_1, \dots, \phi_n\}$ by simple functions where no differentiation is needed). All other symbols (except h and $\mathfrak{D}$) in (5) and (6) refer to vectors from $\mathbb{R}^{3n}$: $\mathbb{A}_s$, $\mathbb{V}_s$ and $\mathbb{U}_s$ approximate $\ddot{u}(., sh)$, $\dot{u}(., sh)$ and $u(., sh)$, $\mathbb{C}(.)$, $\mathbb{K}(.)$, $\mathbb{F}_s$, $\mathbb{G}_s$ and $\widetilde{\mathbb{G}}_s(.)$ are known a priori, $\mathbb{V}_s^\times \approx \mathbb{V}_s$ should be predicted as $\mathbb{V}_s^\times = \mathbb{V}_{s-1/2} + h(\mathbb{V}_{s-1/2} - \mathbb{V}_{s-1})/2$ for the first guess and corrected by iterations (if needed), $\mathbb{U}_0$ is zero-valued, $\mathbb{V}_0$ can be set using $\widehat{v}(x_r)$, $\mathbb{V}_{1/2} = \mathbb{V}_0 + h\mathbb{A}_0/2$ (to avoid undefined $\mathbb{V}_{-1/2}$ in the second formula of (6)).

In numerous papers written by engineers all considerations start with some discrete formulae like (5) and (6), continuing with their various modifications and al-

ternatives, which leads to the risk of misunderstanding with the language of mathematicians. Namely the common form of (5) is $\mathbb{MA}_s = \mathcal{F}_s^I + \mathcal{F}_s^E + \mathcal{F}_s^C$ where, as inherited from *Section* 2, according to the full discretization above, $\mathbb{MA}_s$ (as the complete left-hand side of (5)) represents the inertia forces, $\mathcal{F}_s^I$ the internal forces, expressed by the fourth and fifth right-hand-side additive terms of (5), $\mathcal{F}_s^E$ the external forces, expressed by its first and second additive terms, and $\mathcal{F}_s^C$ the contact forces, expressed by its third additive term, whose effective evaluation is the most delicate task. The following blocks of comments are motivated by the experience with the development of the prototype of the computational tool for the effective simulation of multiple contact of deformable bodies, applicable e. g. to crash testing in the automotive industry.

General approach. Our numerical approach should ensure all computations regardless of their environment, i. e. in sequential, parallel or hybrid manner, on a computer network. Each cluster node, considered in the hybrid form of computation, as suggested by [23], can be represented by some single workstation, which processes computation of a set of associated macro-entities; it can comprise a multi-core CPU capable of executing computational instructions in a fully parallel form. All computational procedures are activated within a global time loop. These computing cluster nodes are called worker nodes. The parallel and hybrid types of computations require the synchronization of CPU threads between individual dependent parts of the computation on each worker node. The synchronization is performed by means of barriers, supplied by some supportive processes. This mainly concerns the functionality focused on data exchange with the central server (master node) used in the hybrid type of computation compatible with [22] and [8]. The procedures themselves are called from another thread, focused purely on communication within the in scope of a computer network.

Contact analysis. The computational platform is assumed to deal with the node-to-segment type of contact in the sense of [33]. Due to its generality, the algorithm should be applicable to all finite elements of a model to find all finite element (FE) nodes suspected from the penetration of a finite element. A naive way to perform contact detection, checking each body against all other ones, ignoring any available information about the distribution of particular bodies in $\mathbb{R}^3$, has the very expensive time complexity $O(\mathcal{N}^2)$, $\mathcal{N}$ being the number of items in a dataset. A more suitable is offered by the nearest neighbour (NN) search, following [26]. The core of such algorithm is defined as a collection of $\mathcal{N}$ objects (FE element nodes); this builds a data structure which provides objects (FEs, their nodes, etc.) in the time as fast as possible, based on the NN query. Two levels of such analysis can be distinguished: i) search for penetration between bounding box volumes encapsulating individual macro-entities, and ii) search for contacts betweens FE nodes and individual FEs, using the node-to-segment approach. Even i) separately (as presented in both examples of *Section* 4) provides underlying support for the analysis of Macro Entity Interaction Multi-graph (MEIM, see lower) regarding the data redistribution within a computer cluster. The *kd*-tree data structure, utilized e. g. for machine learning

and for composition of graphical (gaming) engines, is able to provide an algorithmic support for both i) and ii).

Nearest neighbour search. Let us consider a given set S_p of points p in some high-dimensional real space. Our aim is to construct, to any query point q, a data structure able to find the point in S_p closest to q. Such NN problem belongs to a larger class of proximity problems investigated in computational geometry. Geometric range-searching data structures are constructed by subdividing $\mathbb{R}^3$ into several regions with some predefined properties and recursive generation of a data structure for each such region. Range queries are answered with such a data structure by performing a depth-first search through the resulting recursive space partition. Such data structure is created only once, until the development of situation (as of the FE-based approximate solutions by (5)) does not force its dynamical changes; this algorithm can be useful also for the MEIM analysis. The data structure used here is the k-dimensional tree (kd-tree), designed by [1] as a powerful extension of one-dimensional trees, i. e. the binary tree where the underlying space is partitioned using the value of just one attribute at each level of the tree, instead of all d attributes, unlike the quad-tree, introduced by [27], making such d-tests at each level. The basic analysis of kd-trees can be found in [20]; for its development see [25] and [32]. To compare other multi-dimensional data structures for spatial databases, cf. [19] for R-trees and their mutations, [2] and [14] for X-trees and their mutations and [36] for PH-tree.

Range search. An algorithm working with the kd-tree data structure consist i) of the assembly of a kd-tree map from the appropriate set S_p and ii) of its subsequent usage to obtain a set of nodes falling within the searching range query of any examined node belonging to an appropriate FE. Since the above sketched approach to contact detection can include the topology of the discretized model for the explicit time integration using (5) with (6), no algorithm for deletion of nodes or tree balancing algorithm are needed in i). In ii) we can traverse the kd-tree, but visit only nodes whose region is intersected by the query rectangle. If a region is fully contained in the query rectangle, we can report all the points stored in its sub-tree. When the traversal reaches a leaf, we have to check whether the point stored at the leaf is contained in the query region and, if so, report it.

Explicit integration scheme. Only one special type of FEs will be presented here for simplicity of numerical simulation of a massive impact process, namely the flat shell finite element with co-rotated coordinates of the Reissner - Mindlin type with linear fields for rotations and transverse deflections, developed by [29]. It is very effective in an explicit integration due to a smaller number of operations required for numerical integration (single quadrature point). Their geometrically non-linear behaviour was analyzed in [11] and [34] in details; its rate of convergence is approximately of quadratic order. The concrete form of explicit integration of FE forces $\mathcal{F}_s^I$, $\mathcal{F}_s^E$ and $\mathcal{F}_s^C$ for $s \in \{1, \ldots, m\}$, as required by (5), depends on the implementation of nonlinearities of various types; namely the approach of [34] expects the large rotational kinematics in the small strain regime. Such procedure is performed

by each hardware computational thread of a multi-core CPU; a specific number of FEs is assigned to each thread to optimize the thread load.

Distributed and parallel analysis. Two levels occur in the explicit FE analysis: i) standard process of both parallel integration of all FEs and explicit evaluation of (5), working with parallel integration of internal, external and contact forces, mapping FE ranges on the individual cores of a multi-core CPU, ii) parallel processing of MEIM on computer cluster, representing a distributed computational process able to run on a computer cluster within a cloud environment or some VPN (Virtual Private Network). The TCP/IP protocols enable the interprocess communication within clusters, with difficulties related to the CAP theorem (Consistency, Availability, Partition tolerance) by [5]; for its improvements cf. [6] and [35]. The data distribution for numerical computations is based on the domain decomposition (DD). From this class of methods we need to adopt the FE tearing and interconnecting (FETI), suggested by [10] and developed by [9], [17], [7] and [16] to (5) and (6); for many references (482 items) to particular variants of DD see [30], especially [30, Part 6.3] for the one-level FETI, [30, Part 6.4] for the dual-primal FETI ana [30, Part 8.5] for their applications to elasticity.

Advantages and drawbacks. In our approach each separate discretized domain is able to interact with its surroundings through the contact forces. All domains that come to contact then must be solved together within one worker node in a computer cluster. A large number of moving domains is represented by MEIM, whose edges are related to particular contacts. Such movement of domains is controlled, following [12], by the autonomous character of Lagrangians; this can control even the whole process of data distribution across the computer cluster. Nevertheless, the distributed applications, sketched here, suffer from a number of issues that need to be resolved gradually to reach an optimal model. Serious problems are: i) random application freezing, ii) model data migration between individual worker nodes in a computer cluster at runtime (input structural data, serialized contents of variables), iii) type of transferred data (unstructured vs. structured protocol) and iv) merging of data from individual worker nodes to get the final view on simulation results.

4. Illustrative example

The example presents the announced results for two benchmark problems, referring to the first and second model problem in *Section* 2. These results were obtained from the in-house software at BUT for the type of shells introduced in *Section* 3.

Fig. 1 shows the time development of contacts / impacts of elastic shells in selected time steps: i) for 1 big sphere falling to 1 fixed plane rectangle (3 upper graphs) and ii) for 10 small spheres thrown to 3 moving plane rectangles (3 lower graphs).

5. Conclusions

The aim of this paper was to show the possibility of effective computational analysis of contacts / impacts of deformable bodies for selected model problems, referring

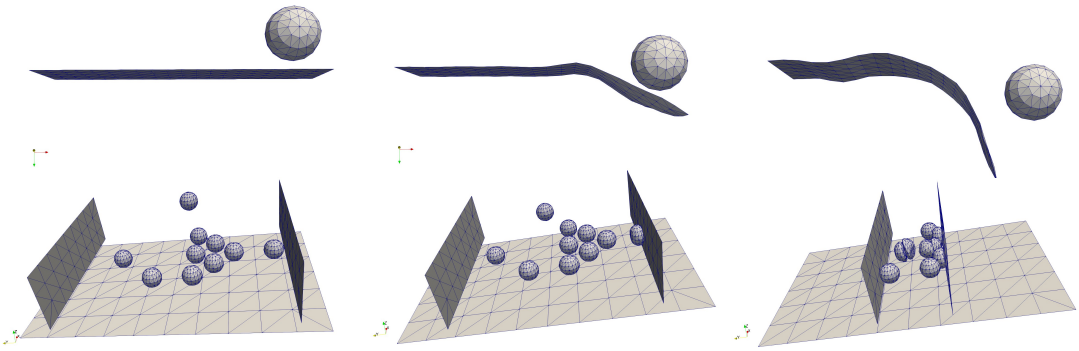

Figure 1: Example of time development of contacts / impacts of some elastic shells.

to still unclosed problems both in mathematical theory and in information science, too. Numerous improvements are required in distributed applications, as summarized at the end of *Section* 3. The upgrade of the explicit calculation scheme, coming from (5) with (6), could be inspired by the recent analyses of [13], [15] and [18].

For real engineering applications the next research step should be the careful revision of physical formulations in the scope of classical thermomechanics, together with the analysis of related mathematical and numerical problems, namely the proper study of energy dissipation on contacts, independently introduced by [28] and [31]. Such dissipation can be accompanied by the formation of plastic or microscopic damage zones, followed by the initiation and development of macroscopic cracks and further phenomena, dangerous for the bearing ability and durability of materials and structures.

Acknowledgements

This work was supported by the project of specific university research No. FAST-S-22-7867 at Brno University of Technology (BUT).

References

[1] Bentley, J. L.: Multidimensional binary search trees used for associative searching. *Commun. ACM* **18** (1975), pp. 509–517.

[2] Berchtold, S., Keim, D. A., and Kreigel, H.-P.: The X-tree: an index structure for high-dimensional data. In: *Proc. 23 VLDB (Int. Conf. on Very Large Databases)* in Bombay (1996), pp. 907–912. VLDB Endowment, Bombay, 1996.

[3] Bermúdez de Castro, A.: *Continuum Thermomechanics*. Birkhäuser, Basel, 2005.

[4] Bonet, J., Gil, A. J., and Ortigosa, R.: A computational framework for polyconvex large strain elasticity. *Comput. Methods Appl. Mech. Eng.* **283** (2015), pp. 1061–1094.

[5] Brewer, E. A.: Towards robust distributed systems. In: *Abstr. 19th PODC (Symposium on Principles of Distributed Computing)* in Portland (Oregon, USA), 2000, p. 7. Assoc. Computing Machinery, New York, 2000.

[6] Brewer, E. A.: CAP twelve years later: how the "rules" have changed. *IEEE Computer Magazine* **45** (2012), pp. 23–29.

[7] Cai, Y., Cui, X., Li, G., and Liu, W.: A parallel finite element procedure for contact-impact problems using edge-based smooth triangular element and GPU. *Int. J. Numer. Methods Eng.* **225** (2018), pp. 47–58.

[8] Celesti, A., Galletta, A., Fazio, M., and Villari, M.: Towards hybrid multi-cloud storage systems: understanding how to perform data transfer. *Big Data Res.* **16** (2019), pp. 1–17.

[9] Dostál, Z., Gomes Neto, F. A. M., and Santos, S. A.: Solution of contact problems by FETI domain decomposition with natural coarse space projections. *Comp. Methods Appl. Mech. Eng.* **190** (2000), pp. 1611–1627.

[10] Farhat, C., and Roux, F. X.: A method of finite element tearing and interconnecting and its parallel solution algorithm. *Int. J. Numer. Methods Eng.* **32** (1991), pp. 1205–1227.

[11] Halquist, J. O., et al.: *LS-DYNA Theoretical Manual*. Livermore Software Technology, Livermore (California), 2006.

[12] Har, J., and Tamma, K. K.: *Advances in Computational Dynamics of Particles, Materials and Structures*. J. Wiley & Sons, Hoboken, 2012.

[13] Jammy, S. P., Jacobs, Ch. T., and Sandham, N. D.: Performance evaluation of explicit finite difference algorithms with varying amounts of computational and memory intensity. *J. Comp. Sci.* **36** (2019), pp. 100565 / 1–7.

[14] Keim, D., Bustos, B., Berchtold, S., and Kriegel, H.-P.: Indexing, X-tree. In: *Encyclopedia of GIS* (Shekhar, S., Xiong, H., and Zhou, X., Eds.), pp. 543–547. Springer, Basel, 2017.

[15] Kim, W., and Reddy, J. N.: Novel explicit time integration schemes for efficient transient analyses of structural problems. *Int. J. Mech. Sci.* **172** (2020), pp. 105429 / 1–16.

[16] Kružík, J., Horák, A., Hapla, V., and Čermák, M.: Comparison of selected FETI coarse space projector implementation strategies. *Parallel Comput.* **93** (2020), pp. 102608 / 1–11.

[17] Kůs, P., and Šístek, J.: Coupling parallel adaptive mesh refinement with a nonoverlapping domain decomposition solver. *Adv. Eng. Softw.* **110** (2014), pp. 34–54.

[18] Liu, W., and Guo, W.: A novel predictor–corrector explicit integration scheme for structural dynamics. *Structures* **34** (2021), pp. 2735–2745.

[19] Manolopoulos, Y., Nanopoulos, A., Papadopoulos, A. N., and Theodoridis, Y.: *R-Trees: Theory and Applications*. Springer, London, 2006.

[20] Mehlhorn, K.: *Data Structures and Algorithms.* Springer, Berlin, 1984.

[21] Nečas, J.: *Introduction to the Theory of Nonlinear Elliptic Equations.* Teubner, Leipzig, 1983.

[22] Pääkkönen, P., and Pakkala, D.: Reference architecture and classification of technologies, products and services for big data systems. *Big Data Res.* **2** (2015), pp. 166–186.

[23] Rek, V., and Vala, J.: On a distributed computing platform for contact / impact of elastic bodies. In: *Proc. SNA (Seminar on Numerical Analysis)* 2021 in Ostrava (virtual), pp. 64–67. Inst. of Geonics CAS, Ostrava, 2021.

[24] Roubíček, T.: *Nonlinear Partial Differential Equations with Applications.* Birkhäuser, Basel, 2005.

[25] Rouquier, J., Alvarez, I., Reuillon, R., and Wuilleminet, P.-H.: A *kd*-tree algorithm to discover the boundary of a black box hypervolume. *Ann. Math. Artif. Intell.* **75** (2015), pp. 335–350.

[26] Sabharwal, Y, Sharma, N., and Sen, S.: Nearest neighbors search using point location in balls with applications to approximate Voronoi decompositions. *J. Comput. Syst. Sci.* **72** (2006), pp. 955–977.

[27] Sperber, M.: Quadtree and octree. In: *Encyclopedia of GIS* (Shekhar, S., Xiong, H., and Zhou, X., Eds.), pp. 1695–1700. Springer, Basel, 2017.

[28] Štekbauer, H., Němec, I., Lang, R., Burkart, D., and Vala, J.: On a new computational algorithm for impacts of elastic bodies. *Appl. Math.* **67** (2022), in print, 28 pp.

[29] Stolarski, H., Belytschko, T., Carpenter, N., and Kennedy, J. M.: A simple triangular curved shell element. *Eng. Comput.* **1** (1984), pp. 210–218.

[30] Toseli, A., and Widlund, O.: *Domain Decomposition Methods – Algorithms and Theory.* Springer, Berlin, 2005.

[31] Wang, G., Liu, C., and Liu, Y.: Energy dissipation analysis for elastoplastic contact and dynamic dashpot models. *Int. J. Mech. Sci.* **221** (2022), pp. 107214 / 1–14.

[32] Wehr, D., and Radkowski, R.: Parallel kd-tree construction on the GPU with an adaptive split and sort strategy. *Int. J. Parallel Program.* **46** (2018), pp. 1139–1156.

[33] Wu, S. R.: A variational principle for dynamic contact with large deformation. *Comput. Methods Appl. Mech. Eng.* **198** (2009), pp. 2009–2015, and **199** (2009), p. 220.

[34] Wu, S. R., and Gu, L.: *Introduction to the Explicit Finite Element Method for Nonlinear Transient Dynamics.* J. Wiley & Sons, Hoboken, 2012.

[35] Yuan, L.-Y., Wu, L., You, J.-H., and Shanghai, Y. Ch.: A demonstration of Rubato DB: a highly scalable NewSQL database system for OLTP and big data applications. In: *Proc. ACM SIGMOD Int. Conf. on Management of Data* in Melbourne (2015), pp. 907–912. Assoc. for Computing Machinery, New York, 2015.

[36] Zäschke, T., Zimmerli, Ch., and Norrie, M. C.: The PH-tree – a space-efficient storage structure and multi-dimensional index. In: *Proc. ACM SIGMOD Int. Conf. on Management of Data* in Snowbird (Utah, 2014), pp. 397–408. Assoc. for Computing Machinery, New York, 2014.