Article

Parallel Code Execution as a Tool for Enhancement of the Sustainable Design of Foundation Structures

Jiri Brozovsky [1], Radim Cajka [2] and Zdenka Neuwirthova [2,*]

1 Department of Structural Mechanics, Faculty of Civil Engineering, VSB-Technical University of Ostrava, 708 00 Ostrava, Czech Republic; jiri.brozovsky@vsb.cz
2 Department of Structures, Faculty of Civil Engineering, VSB-Technical University of Ostrava, 708 00 Ostrava, Czech Republic; radim.cajka@vsb.cz
* Correspondence: zdenka.neuwirthova@vsb.cz

Abstract: Civil engineering structures are always in interaction with the underlying parts of the Earth. This form of interaction results in deformations and stresses that affect the service life of structures. Long and predictable service life is one of important aspects of sustainable design. Thus, good knowledge of the interaction effects is an essential part of sustainable design. However, to obtain this information, the use of complex numerical models is often necessary. In many cases, the creation and analysis of such complex models are not possible with the tools usually available in civil engineering practice. Technically, the necessary software and computer hardware exist, but their use for such tasks is still very infrequent and includes many challenges. The main aim of this article was thus to propose an approach of numerical analysis that utilizes parallel supercomputers and software based on the finite element method. The paper concentrated on the feasibility of the solution and on calculation times, because these aspects are usually the main reasons why engineers in practice tend to reject these approaches. The approach was demonstrated on a model case that was compatible with actual in situ experiments executed by the author's team, and thus the validity of the computed results is verifiable. Limitations of the proposed approach are also discussed.

Keywords: finite element method; supercomputer; soil-structure interaction; ANSYS; parallel processing; parallel code execution

1. Introduction

Design of sustainable civil engineering structures (that is, effective and durable structures with a long service life and with a minimal environmental footprint) is often limited by the available design tools. Today, it is common to use computers and advanced numerical tools like the non-linear finite element method. However, it is hard to use these tools in some areas because their complexity still requires computational power, which is not available with contemporary personal computers nor workstations. One problem area is the soil–structure interaction. This problem is often reduced to the interaction between the foundation structure, or even just a part of the foundation structure, and the soil [1]. Such an approach is common in civil engineering practice, but it does not respect the fact that there is an interaction between the whole structure (definitely not just the foundation) and the area of the Earth under the structure, which can be measured in tens or hundreds of meters (for the sake of simplicity, such areas are addressed in this article as "soil"). Such soil is in reality non-homogeneous space that usually has several layers of materials with different properties (soils, rocks of different types [2]). It is obvious that there is a little space for simplification and that reduction of the problem to the interaction of some structural members with limited Earth volume have to lead the designs that incorporate many inaccuracies [3]. The importance of these inaccuracies increases with time due to time-related changes in all materials and structures involved. Thus, for the design of sustainable structures, it is necessary to use approaches that will make it possible to create

numerical models of structures and their surrounding environments (especially these parts of the Earth volume that are in interaction with the structure). Thus, this paper focuses on the soil–structure interaction problems. Necessary computational [4] and numerical tools exist [5]. However, their actual use requires additional efforts that have to be studied, developed, and then introduced to civil engineering practice.

As it was mentioned above, numerical modelling of building structures is used in every phase of the design process and for every possible material choice to predict various phenomena and independence. Analysis can be performed as linear or nonlinear [6–8]. In both cases, the numerical modelling of building structures requires a large number of simplifications [3]. Material properties are often assumed to be linear elastic or elastic-plastic. The behaviour of many structural elements is idealized and structural supports are often idealized or simplified too. It is still common that ideal models of supports (e.g., hinges, fixed supports) are used. Modern finite element software packages for civil engineering [9,10] often incorporate less or more sophisticated models for soils, which can be used for better representation of soil–structure interaction problem. Some of this software models the compression-only nature of the soil–structure interaction, others do not. Simplified models are often based on the Winkler subsoil model [11,12] or its derivatives like the Pasternak or Kolar models [13]. Models that are based on elastic half-space also exist [14,15] and they are implemented, for example, in the Soilin model of the SCIA software [9,16] or in the academic code by Cajka et al. [17,18].

These models are in general 1D or 2D-oriented and they replace the 3rd dimension (the soil depth and all soil and rock layers) with one or a few parameters. It makes them less suitable for modelling of complex (but very common) situations like the interaction between several buildings with foundations in different depths or even for modelling of a single building that has several levels of foundations. In these cases, the 3D model is required. It can be based on a combination of finite and semi-infinite elements [19,20] or it can be represented as a limited part of 3D half-space. Such a model allows for the inclusion of several material layers with different material properties. These properties can be linear elastic or (preferably) elastic-plastic (the Drucker-Prager or the Mohr-Coulomb models are common).

Using these 3D computational models may achieve more precise results. The calculation must be well prepared including the computing method and all input parameters. Using this detailed computing method will help to design optimized buildings. Optimization will result in less construction material used and therefore in a lower carbon footprint per structure. Making optimized buildings will result in lower production of building materials and have a positive effect on the environment [21].

These 3D models introduce several problems. The issue of obtaining and verifing proper material data is out of the scope of this article. This information is available in other articles that publish results of the experimental program by Cajka et al., for example [1,17,21]. Another important issue is the modelling of boundary conditions. It is well known [22] that the type of boundary condition can importantly influence model behaviour. In this paper, boundary conditions based on study [3] were used. Model behaviour of course also depends on the size of the modelled area: the smaller the area is, the bigger are the unnecessary influences on stress and strain state in the modelled area.

One of the biggest issues is a model's size. Finite element models of buildings can have thousands to millions of unknowns. The subsoil area related to a structure is usually 4–20 times bigger than the structure itself and it extends in all three directions. This size can be determined approximately. For example, the authors of [23] propose an analytical method designed for environments close to the studied one.

Thus, the model of this area may have many times more unknowns than the model of the building itself [24]. For larger cases, it is easily possible to reach the size of tens of millions of unknowns for the whole model. Computation times will be increased if the material model is non-linear or if there are measures for modelling used of a compression-

only nature for the soil–structure interaction (for example, if so-called contact elements [25] are used).

One of the possible approaches to shorten computational time is the use of parallel processing. At the current time, most personal computers and workstations use multi-core central processing units (CPU) [26–28], with 4–32 cores. For a larger number of CPUs or cores, it is necessary to use a supercomputer [4]. This article is going to discuss investigations of the possible use of such supercomputers [29] for soil–structure interaction problems.

Using a supercomputer does not necessarily mean a larger carbon footprint. Tasks computed on a supercomputer wait in the queue and are optimized before their calculation. The supercomputer usually works all the time without delays. Some supercomputers focus on lowering the carbon footprint. The main problem with supercomputers is the large amount of heat that is produced as a side effect of their usage. They usually need a large cooling system. The most popular solution is a connection of the cooling system to the heating system and the use of this energy for building heating, taking a step towards being carbon-neutral [30].

Therefore, it is also important to investigate the optimal size of a model, so the model is detailed enough without consuming too many resources or taking too long [22].

2. Approach Description

The main idea of this article is to use supercomputers to make possible a more detailed numerical analysis of the soil–structure problem. The proposed approach includes these steps:

1. Preliminary analysis (based on analytical formulas or simple numerical models and on field testing data). In this stage, the size of the modelled soil volume should be determined.
2. Initial numerical model based on finite element method. This model may be used for verification of the general behaviour of the numerical model. Size and complexity of this model has to be feasible for a common desktop computer.
3. Preparation of a detailed numerical model for a parallel computer. Such models can be prepared in a similar fashion as that of the usual numerical models. The usual problem is size of the model, which makes it impractical (or even impossible) to create it on a desktop computer. However, some contemporary engineering software allows for a combination of a numerical model of civil engineering structure (which usually can be created interactively with the use of usual engineering software) with the subsoil model which may have to be generated (non-interactively) directly on the supercomputer with support of program commands. For example, the ANSYS [25] software uses the APDL command language, which was used for this work in the example presented below.
4. Numerical analysis of the prepared model on a supercomputer. In this step, linear elastic material properties should be used and the results (at least deformations) should be compared with those from Step 2. This step is necessary to reveal possible critical errors or incompatibilities of the model. In addition, time consumption needs to be assessed to ensure the non-linear solution. It is also necessary to note that parallel code execution offers further challenges, which may have negative influences on the solution. For at least some finite element programs, it is true after a certain problem size, the solution may become unstable, and the maximum possible size of the problem has to be searched [31]. This of course requires re-modelling of the problem (number of finite elements of the soil volume have to be scaled down without changing the building structure model). This task can be eased if a parametric command language like the APDL is used.
5. Non-linear analysis on a supercomputer with all desired non-linearitys (contacts between elements and material non-linearity, for example).

6. Analysis of results. This task may also require processing on a parallel computer. Once again, it is recommended to use program commands to extract the desired result data.

The following text discusses this approach on an example of a structure on soil. The main reason for the selection of such a structure was the availability of results obtained during the long-term experimental program headed by Cajka [1,3,32,33]. The experiment allowed us to study only relatively simple structures in situ (slabs up to 2×2 m) thus numerical models follow their geometry. Steps 1 and 2 are not discussed here because in this particular case, already available data from previous works were used.

The article concentrates on computational times to demonstrate feasibility of the proposed approach. Results of the particular case are not discussed here, just some basic data are shown to verify that the behaviour of the model analysed on parallel computers correlates with the experimental data.

3. Finite Element Modelling

Numerical models were prepared with the use of the finite element method [34]. Both commercial ANSYS [25] and in-house developed finite element packages were used. The material of the subsoil part of the model was assumed to be both linear elastic and elastic-plastic. The material of the structural model was assumed to be linear in most cases.

The model included a sub-volume of subsoil and a model of a reinforced concrete slab, which represented the building structure.

3.1. Used Finite Elements

In the presented work, the brick-shaped, isoparametric, 8-node serendipity family [35,36] finite elements are used. These finite elements are available in the ANSYS software (SOLID45, SOLID65) [25] and in the uFEM finite element code (BRICK9), which was developed at the VSB–Technical University of Ostrava [37].

These finite elements use shape functions of the following form [35]:

$$N_i(\zeta,\eta,\xi) = 0.125\,(1 + \zeta\zeta_i)(1 + \eta\eta_i)(1 + \xi\xi_i)(\zeta\zeta_I + \eta\eta_I + \xi\xi_I - 2) \tag{1}$$

where ζ,η,ξ are natural coordinates of an arbitrary point and ζ_i, η_i, ξ_i are natural coordinates of the ith node of finite elements. The shape functions (N_i) are used for approximation of unknown deformations. These elements are discussed in greater detail in [35].

The 8-node element has 3 unknown parameters in every node (displacements in x, y, and z directions), thus a single element has 24 unknown parameters. The reinforced concrete slab was in most cases [38] represented by the same brick-shaped finite element types as were used for the subsoil model (the SOLID45 and the SOLID65 elements of the ANSYS).

The connection of parts of the model was first assumed as ideal (node-to-node connection) then so-called contact finite elements were used (the TARGE170 and the CONTA175 elements of the ANSYS). These elements were used to simulate the compression-only character of the soil–structure interaction

3.2. Model Description

The designed numerical models were based on an actual experimental setup of the STAND device at the VSB—Technical University of Ostrava [33].

The model included two main parts: a subsoil model that was the shape of a brick and a reinforced concrete slab model that was located in the centre of the top surface of the subsoil model. The reinforced concrete brick was assumed to have dimensions of $2 \times 2 \times 0.15$ m in all cases. The dimension of subsoil was altered. The main computations presented in this paper were done on a model with dimensions $42 \times 42 \times 20$ m.

Dimension of the modelled area and sizes of finite elements were based on previous works [21]. They also depended on computer resources available for this particular research

task. The problem of mode size for this particular class of cases was studied in greater detail and previously published [3,22].

The load was modelled in the form of forces located in the area of 0.2×0.2 m in the centre of the reinforced concrete slab.

Initial model behaviour was studied on a simple model with a small number of finite elements. The reinforced concrete slab was omitted in this case. Thus, this model did not include contact elements. One of the setups for this model is shown in Figure 1. This model used linear elastic material parameters (Young Modulus and Poisson Ratio), listed in Table 1. The initial model had a larger soil volume ($42 \times 42 \times 42$ m) because it was modelled with the use of the uFEM [32] software, which allows for the use of an arbitrary number of processors.
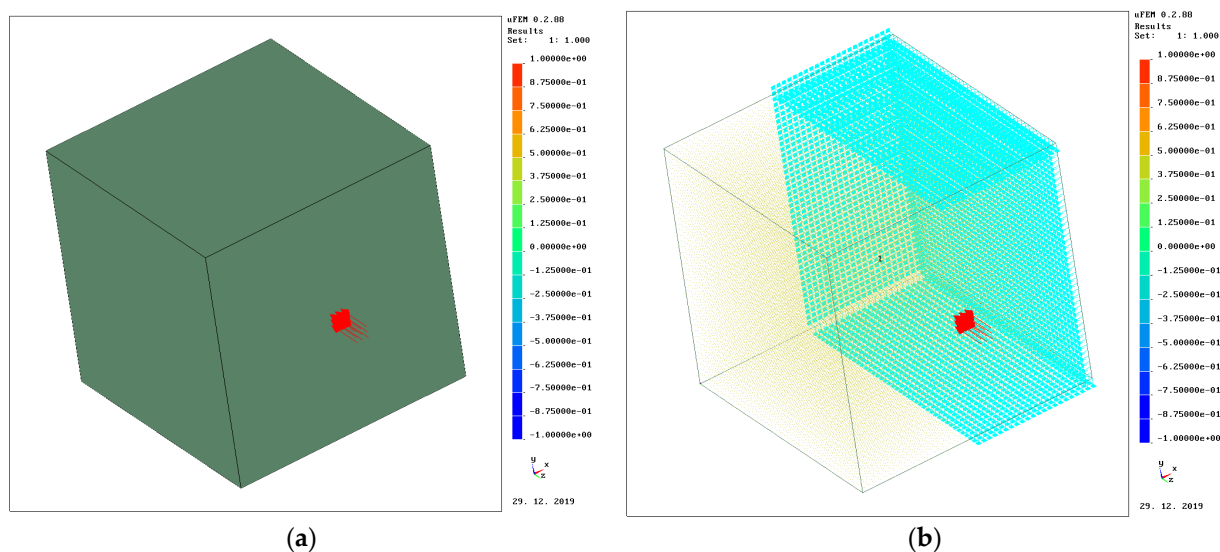

(**a**)　　　　　　　　　　　　　　　　　　　　　　　(**b**)

Figure 1. Initial model geometry (**a**), the orientation of supports (**b**).

Table 1. Material parameters of soil.

Parameter	Value
Young Modulus	24×10^7 Pa
Poisson Ratio	0.35
Cohesion	24×10^7 Pa
Friction Angle	0.45 rad
Dilatancy Angle	0.20 rad

Further models were created in the ANSYS with the use of greater numbers of finite elements and with non-linear material parameters. These models used two different materials: Drucker–Prager elastic-plastic model for soil (Table 1) and the Willam–Warnke model for concrete (Table 2). The Willam-Warnke model implemented in the ANSYS software is based on elastic-plastic behaviour combined with a material softening model for modelling concrete cracking and crushing. The soil model was thus constructed from SOLID45 finite elements and the reinforced concrete slab from the SOLID65 elements. The compression-only connection between soil and slab was envisaged and it was modelled with the use of TARGE170 and CONTA175 contact finite elements. These elements were located on the interface between the slab and the soil volume. The load was applied in the form of pressure located at the 0.2×0.2 m area on the centre of the top of the slab.

Table 2. Material parameters of reinforced concrete.

Parameter	Value
Young Modulus	27×10^7 Pa
Poisson Ratio	0.20
Open Shear Transfer Coefficient	0.40
Closed Shear Transfer Coefficient	0.20
Uniaxial Cracking Stress	640×10^5 Pa
Uniaxial Crushing Stress	320×10^6 Pa
Biaxial Crushing Stress	384×10^6 Pa
Hydrostatic Pressure	554×10^6 Pa
Hydro. Biax. Crush. Stress	464×10^6 Pa
Hydro. Biax. Crack. Stress	557×10^6 Pa
Tensile Crack Factor	0.6

We prepared models of different sizes: the basic model (7000 nodes, Figure 1a) was designed to be used on a desktop workstation, the large model was designed to be executed on a parallel supercomputer (642,330 nodes, Figure 2).
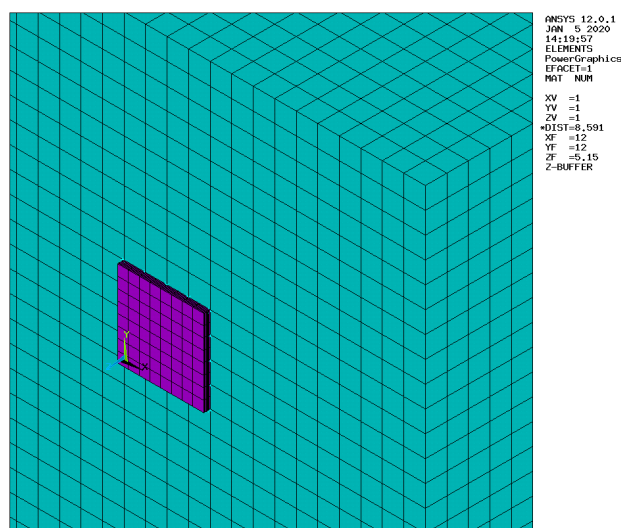
Figure 2. ANSYS-based model of a reinforced concrete slab on the soil.

3.3. Computational Procedures

Computations of the models were done with the use of several environments: The initial model was analysed with the use of the in-house developed uFEM software. The conjugate gradient method [39] with preconditions was used for the solution of the finite element problem. The uFEM software allows parallelization of conjugate gradient solutions by parallelization of key operations (matrix-vector and vector-vector multiplications, problem matrix construction, and results from computations, among others). The software is written in the C language and uses the UNIX threads mechanism to execute parallelization. The computations were executed on a workstation with the POWER9 central processing unit (IBM Corporation, Armonk, U.S.A.) [40] and the Blackbird workstation (Raptor Computing Systems, Belvidere, IL U.S.A.) [41]. This computer allows utilization of up to 16 processes with full performance (4 processes per one CPU core). Due to the available platform (POWER9/Linux), it was not possible to run the ANSYS models on this setup.

The ANSYS models were executed in two different environments. The first was a standard desktop workstation. Unfortunately, the available ANSYS setups did not allow the use of more than 2 CPUs within this environment. Thus, it was not possible to discuss the scalability or limitations of the solution within this environment.

The second setup was the ANSELM supercomputer, which allows running ANSYS with up to 512 CPU cores [31]. The limitation is caused by the limitation of the ANSYS licenses in the National Supercomputing Centre IT4Innovations. The used solver was based on a domain decomposition approach [5,42]. Distributed computing qA implemented with the use of the message passing interface standard (MPI) [43].

4. Results

Computational times are discussed as they are the most important aspects of this paper. The results of this particular study are only briefly mentioned. The computations were compared with available experimental data (available, among others in papers [1,22,33,34]). However, experimental data are limited to deformations (in most cases to slab deformations) and to limited stress sensor data. Thus, only selected deformation data are presented.

4.1. Computational Speed

It is often stated that computational procedures based on parallelization of the solution of linear systems (the parallel conjugate gradients method, for example) can be effectively accelerated with several CPU cores on shared memory computers (on personal computers or traditional desktop workstations) with multiple CPUs or with multi-core CPUs. It is also sometimes stated [25] that the speed increase is negligible on more than four CPU cores or CPU threads. Thus, solutions on a 16-thread POWER9 CPU were conducted and the results are summarized in Table 3.

Table 3. Computational speeds of the uFEM solution.

Number of Threads/Processes (-)	Computational Time (s)	Computational Time (m:s)
1	549	9:09
2	399	6:39
4	259	4:19
6	233	3:53
8	225	3:45
10	212	3:32

The ANSYS solution on the workstation for the model described above required 4:19 on 4 CPUs and 3:53 on 6 CPUs (Figure 3). Thus, it can be concluded that for this particular problem (non-linear model of a given size), there is no noticeable increase in speed based on more than 4 CPUs. This proves the recommendation stated in [25].
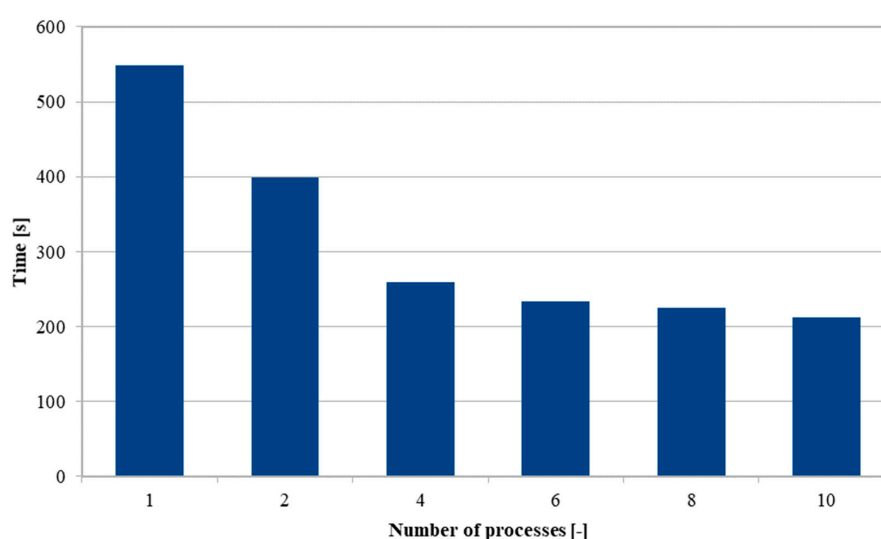

Figure 3. Comparison of solution times using 16-thread Power9 computer processing units (CPUs).

4.2. Computational Speed on HPC

The parallel computing on a supercomputer was performed on the ANSELM supercomputer located in the National Supercomputing Centre IT4Innovations, Ostrava. Detailed information about the submission and limitation of the supercomputer was published before and included the optimization of the resource allocation [31].

We were allocated twice the number of cores than were actually used for computing each case. The unused cores served as a memory buffer. Therefore, the calculation was performed on 256 cores (allocating 512 cores), 128 cores (allocating 256 cores), 64 cores (allocating 128 cores), 32 cores (allocating 64 cores), 16 cores (allocating 32 cores), and 8 cores (allocating 16 cores). Every calculation was performed three times to collect accurate results (Table 4).

Table 4. Computational speeds of the high-performance computing solution.

Nodes	Cores per Node	HPC Licenses	Solution Time (h:min:s)	Average Solution Time (s)	Average Solution Time (h:min:s)
32		256	0:18:59 0:19:04 0:19:22	1148	0:19:08
16		128	0:29:42 0:29:54 0:29:59	1791	0:29:51
8		64	0:52:39 0:52:47 0:53:10	3172	0:52:52
4	16	32	1:41:11 1:41:21 1:40:32	6061	1:41:01
2		16	3:29:26 3:29:24 3:28:25	12545	3:29:05
1		8	6:44:12 6:47:03 6:44:25	24313	6:45:13

Solution times from Table 4 were transferred to Figure 4. A horizontal axis shows solution time in seconds using a logarithmic scale. The ANSYS solution on the supercomputer for the model described showed an exponential increase in speed.
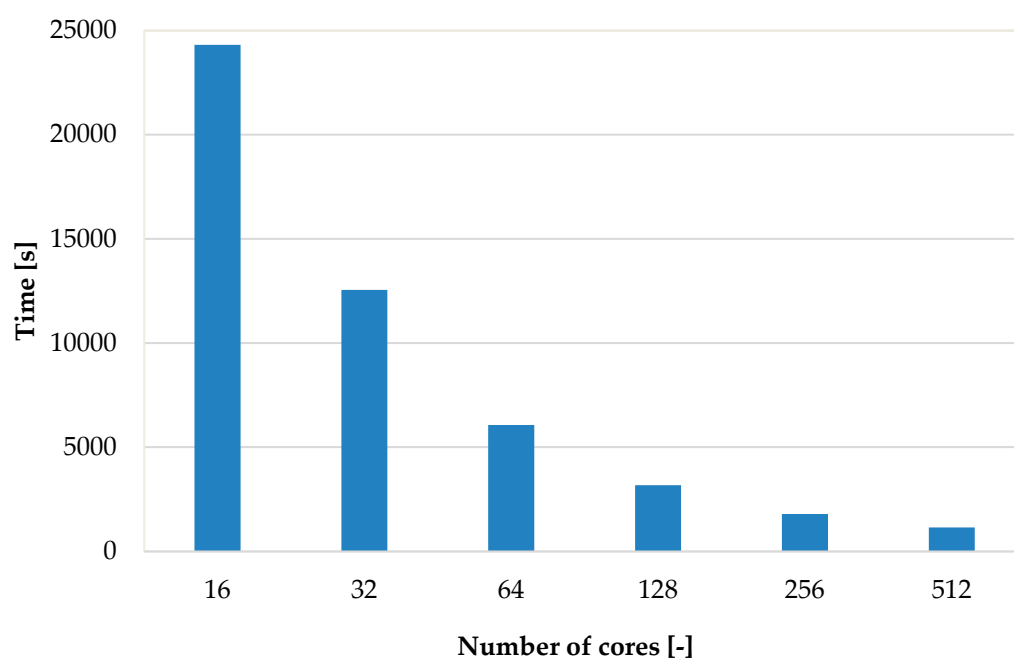

Figure 4. Comparison of the HPC solution times.

4.3. Deformation

Most of the modelled area had the shape of a rectangular brick, which allows for the preparation of a very effective finite element mesh. As an example of the obtained results, the deformation under the centre point of the slab is shown in Figure 5. The maximum computed deformation was −0.653 mm, which correlates well with the data available from experiments (average value of −0.6 mm) for the same point (bottom centre point of the slab).
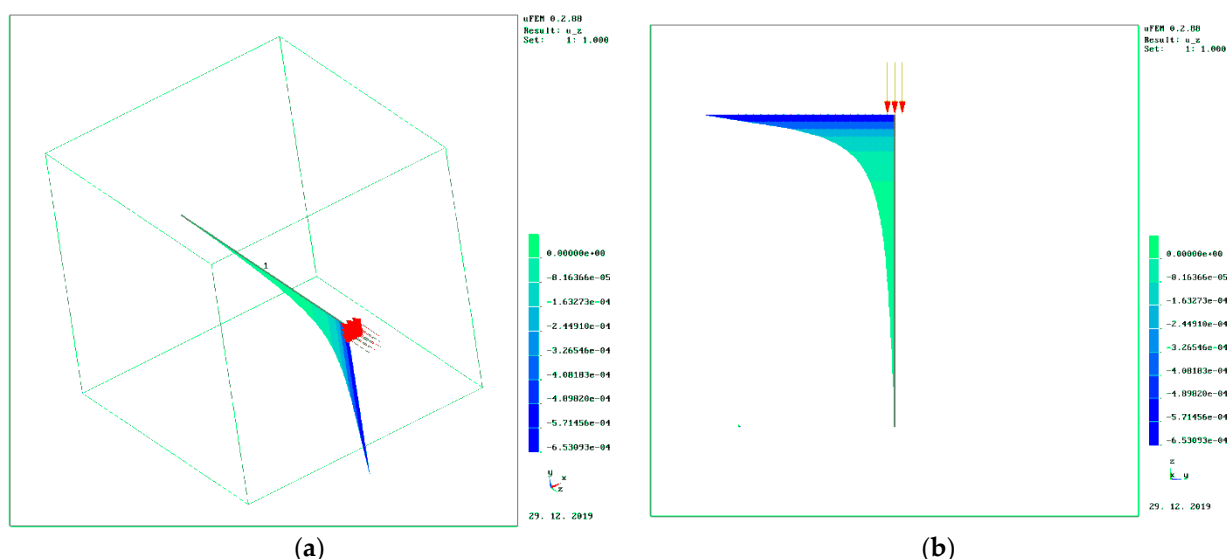

Figure 5. Numerical model results: Deformation in the z-axis in 3D (**a**) and 2D (**b**).

5. Discussion

The solution on the workstation for the model described above required 4:19 on 4 CPUs and 3:53 on 6 CPUs, and it was concluded that there is no noticeable increase in speed based on more than 4 CPUs. A larger problem was executed on a supercomputer using 1×16 to 32×16 cores. The biggest time saving appeared with a change from 1×16 to 2×16 computation, where time was halved. This phenomenon occurred up to 64 cores. Using more than this number seems to be unnecessary for this particular problem. The presented case was relatively simple but it shows that the selected approaches are feasible and can be used for the more complex problems that usually arise from the needs of civil engineering design. However, it is obvious that in the case of supercomputers, there are still available resources to solve more complex cases. Such resources would not be available on much more constrained personal workstations like the one used for the purposes of this article. It has to be noted that a single load case was solved. The actual design of any new civil engineering structure requires solving several load case combinations and also a number of design alternatives. All load cases combinations can be solved concurrently (in parallel) on a supercomputer (provided that sufficient computer memory, CPU cores, and software licenses are available). Software like the ANSYS and uFEM have abilities for non-interactive evaluation of data based on their internal scripting languages, which also makes it possible to execute preliminary evaluations of results (finding the most critical details or stress-based or strain-based comparison of alternative designs) on a supercomputer and thus save additional time.

6. Conclusions

This article proposed an algorithm to increase the speed of the numerical analysis of large soil–structure problems by using parallel supercomputers. The algorithm is based on the use of available software and numerical models. The approach was illustrated on

an example based on an experimental program conducted by one of paper authors. The presented information allows us to define these main conclusions:

- Parallel computations, especially if they are executed on large computers or supercomputers, make it possible to solve large civil engineering problems like soil–structure interactions in much greater detail than is possible on the desktop computers that are usually available to civil engineering professionals.
- The approach proposed in the article is feasible and it has the potential to be used in civil engineering practice.
- Limitations: finite element solution scalability is limited to some extent. When these limits are reached, it is more sound to use the available supercomputing resources for the solution of model (design) alternatives or to speed-up the solution of load case combinations by their concurrent execution. That is, time-savings of a single load step solution or a single model solution are not the only nor the greatest benefits of the use of supercomputers.
- It is always necessary to execute a multi-step computational analysis that will include small-scale computation of less-detailed models and only then perform the large-scale computations on a supercomputer.
- The above mentioned parallel computation approach gives the possibility of designing enhancements that should lead to a more effective design of structures. The ability to provide results of complex and detailed models of complete structures with the foundation makes possible the selection of the best design alternatives in terms of carbon footprint (material saving criteria) and also with the necessary level of durability and service life (by a selection of alternatives with the smallest number of critical structural points as possible).

It is obvious from the computation times that are shown in this article that actual optimization approaches (which usually require hundreds, thousands, or even more repeated computations of complete structural models) are still hard to accomplish.

References

1. Cajka, R.; Marcalikova, Z.; Bilek, V.; Sucharda, O. Numerical Modeling and Analysis of Concrete Slabs in Interaction with Subsoil. *Sustainability* **2020**, *12*, 9868. [CrossRef]
2. Todorov, M.; Kostov, V. Prediction of the Development of Long Term Deformations. In Proceedings of the 17th International Multidisciplinary Scientific GeoConference, Vienna, Austria, 27–30 November 2017; Volume 17, pp. 139–146. [CrossRef]
3. Cajka, R.; Labudkova, J. Dependence of Deformation of a Plate on the Subsoil in Relation to the Parameters of the 3D Model. *Int. J. Mech.* **2014**, *8*, 208–215. Available online: http://www.naun.org/main/NAUN/mechanics/2014/a342003-076.pdf (accessed on 25 October 2020).

4.   Our Supercomputers. Available online: https://www.it4i.cz/en/infrastructure/our-supercomputers (accessed on 25 October 2020).

5.   Kozubek, T.; Vondrák, V.; Menší, M.; Horák, D.; Dostál, Z.; Hapla, V.; Kabelíková, P.; Čermák, M. Total FETI domain decomposition method and its massively parallel implementation. *Adv. Eng. Softw.* **2013**, *60*, 14–22. [CrossRef]

6.   Yu, H.; Li, S.; Liu, Y.; Zhang, J. Non-Linear Analysis of Stress and Strain of Concrete-Faced Rockfill Dam for Sequential Impoundment Process. *Math. Comput. Appl.* **2010**, *15*, 796–801. [CrossRef]

7.   Dede, F.T. Nonlinear Finite Element Analysis of an R/C Frame Under Lateral Loading. *Math. Comput. Appl.* **2011**, *16*, 947–958. [CrossRef]

8.   Kantar, E.; Erdem, R.T.; Anıl, Ö. Nonlinear Finite Element Analysis of Impact Behavior of Concrete Beam. *Math. Comput. Appl.* **2011**, *16*, 183–193. [CrossRef]

9.   *ENGINEER S.C.I.A, Version 17.0*; Nemetschek Group Software: Munich, Germany, 2017.

10.   *R.-FEM, Ing.*; Software Dlubal GmbH. Am Zellweg: Tiefenbach, Germany, 2013.

11.   Winkler, E. *Die Lehre von der Elastizität und Festigkeit*; H. Dominicus: Prague, Czech Republic, 1867.

12.   Pasternak, P.L. On a new method of analysis of an elastic foundation by means of two foundation constants. *Gos. Izd. Lit. Po Stroit. Arkhitekture* **1954**, 124773833.

13.   Kolar, V.; Nemec, I. *Modelling of Soil-Structure Interaction*; Elsevier: Amsterdam, The Netherlands, 2012; Volume 58.

14.   *ČSN 73 1004 Geotechnical Design–Foundations–Requirements for Calculation Methods*; Czech Office for Standards, Metrology and Testing: Prague, Czech Republic, 2020.

15.   *CSN EN 1997-1-1 Eurocode 7: Geotechnical Design–Part 1: General Rules*; Czech Office for Standards, Metrology and Testing: Prague, Czech Republic, 2006.

16.   Kolar, V.; Nemec, I. *Modelling of Soil Structure Interaction*; Academia: Prague, Czech Republic, 1989; 336p.

17.   Cajka, R.; Labudkova, J.; Mynarcik, P. Numerical solution of soil-foundation interaction and comparison of results with experimental measurements. *Int. J. Geomate* **2016**, *11*, 2116–2122. Available online: http://www.geomatejournal.com/sites/default/files/articles/2116-2122-1208-lobudkova-July-2016-c1.pdf (accessed on 20 September 2020).

18.   Cajka, R.; Manasek, P. Building structures in danger of flooding. In Proceedings of the IABSE Conference: Role of structural engineers towards reduction of poverty, New Delhi, India, 19–22 February 2005; pp. 551–558.

19.   Brozovsky, J.; Cajka, R.; Koktan, J. Constitutive Models for Design of Sustainable Concrete Structures. In Proceedings of the IOP Conference Series: Earth and Environmental Science, Ho Chi Minh City, Vietnam, 17–19 April 2018; p. 012036.

20.   Duris, L.; Hrubesova, E. Numerical Simulation of the Interaction between Fibre Concrete Slab and Subsoil—The Impact of Selected Determining Factors. *Sustainability* **2020**, *12*, 10036. [CrossRef]

21.   Damineli, B.L.; Kemeid, F.M.; Aguiar, P.S.; John, V.M. Measuring the eco-efficiency of cement use. *Cem. Concr. Compos.* **2010**, *32*, 555–562. [CrossRef]

22.   Neuwirthova, Z.; Cajka, R. Parametric study of input parameters of soil-structure interaction based on elastic half-space theory. *WSEAS Trans. Appl. Theor. Mech.* **2019**, *13*, 167–174.

23.   Kuklik, P.; Broucek, M.; Kopackova, M. Fast analytical estimation of the influence zone depth, its numerical verification and FEM accuracy testing. *Struct. Eng. Mech.* **2009**, *33*, 635–647. [CrossRef]

24.   Nepelski, K. A FEM analysis of the settlement of a tall building situated on loess subsoil. *Open Eng.* **2020**, *10*, 519–526. [CrossRef]

25.   Kohnke, P. *ANSYS, Inc. Theory*; SAS IP Press: Canonsburg, PA, USA, 2001.

26.   Smith, R. NVIDIA Announces the GeForce RTX 30 Series: Ampere for Gaming, Starting With RTX 3080 & RTX 3090. 2020. Available online: www.anandtech.com (accessed on 20 September 2020).

27.   Kowaliski, C. Intel Releases 15-Core Xeon E7 v2 Processor. 2014. Available online: https://techreport.com/26056/intel-releases-15-core-xeon-e7-v2-processor/ (accessed on 10 June 2020).

28.   *Sempron™ 3850 APU with Radeon™ R3 Series | AMD*; Advanced Micro Devices, Inc.: Santa Clara, CA, USA, 2019.

29.   Pesatová, K.; Polakova, B.; Cawley, J. *Supercomputing in Science and Engineering, IT4Innovations National Supercomputing Center Czech Republic 2017*; Smetana, M., Smetanová, E., Eds.; VŠB–Technical University of Ostrava: Ostrava-Poruba, Czech Republic, 2017; p. 224. ISBN 978-80-248-4037-6.

30.   Making Supercomputers Carbon Neutral-CSC's Data Centre Will Be Powered with Hydro-Electricity Supplied by Vattenfall. 2020. Available online: http://primeurmagazine.com/weekly/AE-PR-11-20-59.html (accessed on 20 August 2020).

31.   Neuwirthova, Z.; Cajka, R. Parallelizability of Tasks Using Ansys Hpc on Supercomputer to Solve Large Tasks of Civil Engineering—A Case Study. *ARPN J. Eng. Appl. Sci.* **2020**, *15*, 626–633.

32.   Cajka, R.; Marcalikova, Z.; Kozielova, M.; Mateckova, P.; Sucharda, O. Experiments on Fiber Concrete Foundation Slabs in Interaction with the Subsoil. *Sustainability* **2020**, *12*, 3939. [CrossRef]

33.   Hrubesova, E.; Mohlyla, M.; Lahuta, H.; Bui, T.Q.; Nguyen, P.D. Experimental Analysis of Stresses in Subsoil below a Rectangular Fiber Concrete Slab. *Sustainability* **2018**, *10*, 2216. [CrossRef]

34.   Zienkiewicz, O.C.; Taylor, R.L.; Zhu, J.Z. *The Finite Element Method: Its Basis and Fundamentals*; Elsevier: Amsterdam, The Netherlands, 2005.

35.   Irons, B.M. Engineering applications of Numerical Integration in Stiffness Methods. *AIAA J.* **1966**, *4*, 2035–2037. [CrossRef]

36.   Kikuchi, F.; Okabe, M.; Fujio, H. Modification of the 8-node serendipity element. *Comput. Methods Appl. Mech. Eng.* **1999**, *179*, 91–109. [CrossRef]

37.	Brozovsky, J.; Koktan, J. Parallel Execution of Structural Mechanic Tasks with Use of Small Supercomputers. *Procedia Eng.* **2017**, *190*, 296–303. [CrossRef]
38.	Cajka, R.; Krivy, V.; Sekanina, D. Design and Development of a Testing Device for Experimental Measurements of Foundation Slabs on the Subsoil. *Trans. VSB Tech. Univ. Ostrava Civil Eng. Ser.* **2011**, *11*, 1–5. [CrossRef]
39.	Barrett, R. *Templates for the Solution of Linear Systems: Building Blocks for Iterative Methods*; Society for Industrial and Applied Mathematics-SIAM: Philadelphia, PA, USA, 1994.
40.	Power 9. Available online: https://en.wikipedia.org/wiki/POWER9 (accessed on 25 October 2020).
41.	Blackbird. Available online: https://wiki.raptorcs.com/wiki/Blackbird (accessed on 25 October 2020).
42.	Farhat, C.; Roux, F.X. An unconventional domain decomposition method for an efficient parallel solution of large-scale finite element systems. *SIAM J. Sci. Stat. Comput.* **1992**, *13*, 379–396. [CrossRef]
43.	Open MPI. Available online: https://www.open-mpi.org/ (accessed on 25 October 2020).