

# Scope of MPI/OpenMP/CUDA Parallelization of Harmonic Coupled Finite Strip Method Applied on Large Displacement Stability Analysis of Prismatic Shell Structures

Miroslav Hajduković<sup>1</sup>, Dragan D. Milašinović<sup>2</sup>, Miloš Nikolić<sup>1</sup>,  
Predrag Rakić<sup>1</sup>, Žarko Živanov<sup>1</sup>, and Lazar Stričević<sup>1</sup>

<sup>1</sup> University of Novi Sad, Faculty of Technical Sciences,  
Trg D. Obradovića 6, 21000 Novi Sad, Serbia  
{hajduk, losmi83, pec, zzarko, lucky}@uns.ac.rs

<sup>2</sup> University of Novi Sad, Faculty of Civil Engineering,  
24000 Subotica, Serbia  
ddmil@gf.uns.ac.rs

**Abstract.** This paper presents scope of parallelization in large displacement stability analysis of orthotropic prismatic shells with simply supported boundary conditions along the diaphragm-supported edges. A semi-analytical harmonic coupled finite strip method (HCFSM) is used to solve the large deflection and the post-buckling problems. In the HCFSM formulation the coupling of all series terms highly increases computation time when a large number of series terms is used. Fortunately such computation time increase can be compensated by application of MPI, OpenMP and CUDA approaches to parallelization. This paper shows how these approaches can be applied to HCFSM formulation, and what results can be expected.

**Keywords:** Prismatic Shells, Stability analysis, HCFSM, MPI, OpenMP, CUDA.

## 1. Introduction

Typical flat plate structures under consideration here are simply supported by diaphragms and may have arbitrary longitudinal edge conditions. It may be subjected to a transverse loading between its ends (the large-deflection problem) or to an in-plane progressive destabilizing, buckling load (the post-buckling problem) or conceivably to both actions simultaneously. For these structures, the design process should lead to define the optimal morphology of the transversal cross-section, which means its geometry, size, shape and topology.

If a structure undergoes large deformation, the second order terms regarding the strains cannot be ignored. The strain-displacement relations, within the context of a Green-Lagrange strain tensor, represent a sum of

linear and non-linear part. Analysis of plates in the post-buckling range is generally performed on the basis of von Karman equations or by employing an energy approach. Only approximate solutions can be obtained, taking into account the in-plane (membrane) and out-of-plane (bending) boundary conditions.

In this work we present the semi-analytical harmonic coupled finite strip method (HCFSM) for geometric non-linear analysis of thin plate structures under multiple loading conditions. This method takes into account the important influence of the interaction between the buckling modes. In contrast, the finite strip method (FSM), in its usual form, ignores this interaction and therefore cannot be used in analysis of these structures [1].

However, already high computational complexity of FSM becomes even higher in the HCFSM formulation that couples all series terms. The result is a very long calculation time of the existing sequential HCFSM software in cases of a large number of series terms [2]. In order to speed up the lengthy calculations various parallelization techniques can be used. Previous research [3] has considered FSM parallelization based on the “networked workstations” and to our knowledge this is the only publicly available result on this subject. Our research treats a new method (HCFSM) and different computer architectures. We show that not only the multicomputer architecture based on message passing scheme (MPI [4]) is suitable for HCFSM parallelization, but also the multiprocessor architecture based on shared memory (OpenMP [5] and CUDA [6]) offers a good potential for HCFSM parallelization [7, 8, 9]. The main intention of this paper is to present scope of HCFSM parallelization by separate application of each of three different available parallelization techniques.

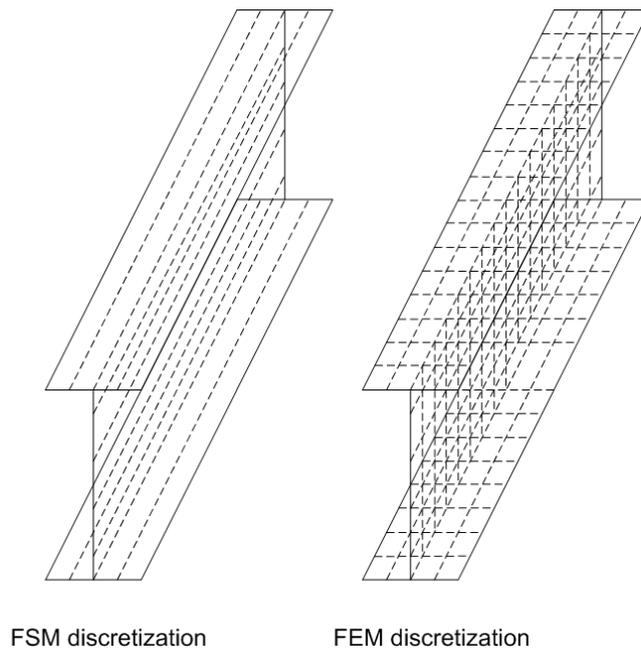
In the second chapter we describe the HCFSM formulation. The third chapter contains a discussion of our hybrid approach to HCFSM parallelization based on application of MPI, OpenMP, and CUDA programming models. This approach results in a noticeable reduction of the execution time and represents a new contribution to the (HCFSM) parallelization. The last two chapters contain acknowledgements and conclusions.

## **2. Harmonic coupled finite strip method for the stability analysis of thin-walled structures**

### **2.1. Introduction to HCFSM**

The contemporary alternatives to perform the stability analysis of thin-walled members are generalized beam theory (GBT) [10], finite element method (FEM) and the FSM [11]. The FSM is derived from the FEM and represents specialization of the FEM to the analysis of engineering structures [2]. The

only difference between FEM and FSM consist of the different discretization of the members, as seen in Fig. 1. The FEM uses a mesh that uses discretization of the member transversally and longitudinally, while the FSM needs only transversal discretization, using currently either harmonic or spline functions in the longitudinal direction of the member.



**Fig. 1.** FSM discretization versus FEM discretization

The FSM was originally developed by Cheung [1] and was widely used by other authors for understanding and predicting the behavior of cold-formed steel members and for bridge decks. There is a reliable harmonic FSM program – CUFSM – for the determination of the critical load parameter of thin-walled prismatic members under a general longitudinal normal stress loading at the extreme cross sections [12].

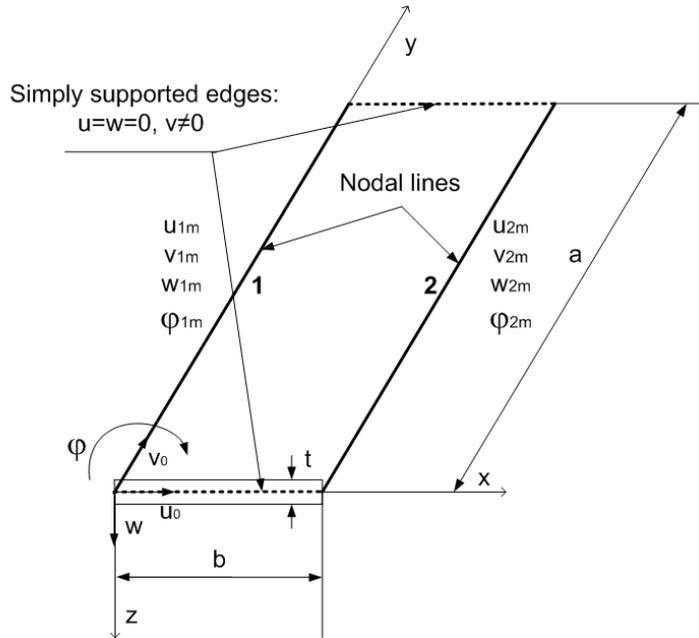
The well known uncoupled FSM formulation, represents a semi-analytical finite element process. As far as linear analysis is concerned, it takes advantage of the orthogonality properties of harmonic functions in the stiffness matrix formulation. However in the case of the geometric nonlinear analysis, and the geometric stiffness matrix calculation, the integral expressions contain the product of trigonometric functions with higher-order exponents, and here the orthogonality characteristics are no longer valid. All harmonics are coupled, and the stiffness matrix order and bandwidth are proportional to the number of harmonics used. This kind of FSM analysis is named the HCFSM [2, 13].

The nonlinear strain-displacement relations in the finite strip can be predicted by the combination of the plane elasticity and the Kirchhoff plate

theory. Using this assumption in the Green-Lagrange strain tensor (1) for in-plane nonlinear strains gives Green-Lagrange HCFSM formulation. Also that, neglecting lower-order terms in a manner consistent with the usual von Karman assumptions gives HCFSM von Karman formulation.

$$\varepsilon_{ij} = 1/2(u_{i,j} + u_{j,i} + u_{k,i}u_{k,j}). \quad (1)$$

The essential feature of geometric nonlinearity is that equilibrium equations must be written with respect to the deformed geometry – which is not known in advance.



**Fig. 2.** Simply supported flat shell strip

In the FSM, which combines elements of the classical Ritz method and the FEM, the general form of the displacement function can be written as a product of polynomials and trigonometric functions:

$$f = \sum_{m=1}^r Y_m(y) \sum_{k=1}^c \mathbf{N}_k(x) \mathbf{q}_{km} \quad (2)$$

where  $Y_m(y)$  are functions from the Ritz method and  $\mathbf{N}_k(x)$  are interpolation functions from the FEM. We define the local Degrees Of Freedom (DOFs) as the displacements and rotation of a nodal line (DOFs=4), as shown in Fig. 2. The DOFs are also called generalized coordinates.

The following approximate functions are used for the simply supported flat shell strip with edges restrained against in-plane movement:

$$\begin{aligned}
 u_0 &= \sum_{m=1}^r Y_{um}^u \mathbf{N}_u^u \mathbf{q}_{um}^u = \sum_{m=1}^r Y_{um}^u \left[ 1 - \frac{x}{b} \quad \frac{x}{b} \right] \mathbf{q}_{um}^u, \\
 v_0 &= \sum_{m=1}^r \frac{a}{m\pi} Y_{um}^v \mathbf{N}_u^v \mathbf{q}_{um}^v = \sum_{m=1}^r \frac{a}{m\pi} Y_{um}^v \left[ 1 - \frac{x}{b} \quad \frac{x}{b} \right] \mathbf{q}_{um}^v, \\
 w &= \sum_{m=1}^r Y_{wm} \mathbf{N}_w \mathbf{q}_{wm} = \sum_{m=1}^r Y_{wm} [N_1 \quad N_2 \quad N_3 \quad N_4] \mathbf{q}_{wm}, \\
 N_1(x) &= 1 - 3\left(\frac{x}{b}\right)^2 + 2\left(\frac{x}{b}\right)^3, \quad N_2(x) = b \left[ \frac{x}{b} - 2\left(\frac{x}{b}\right)^2 + \left(\frac{x}{b}\right)^3 \right], \\
 N_3(x) &= 3\left(\frac{x}{b}\right)^2 - 2\left(\frac{x}{b}\right)^3, \quad N_4(x) = b \left[ -\left(\frac{x}{b}\right)^2 + \left(\frac{x}{b}\right)^3 \right], \\
 \mathbf{q}_{wm} &= \begin{bmatrix} w_{1m} \\ \varphi_{1m} \\ w_{2m} \\ \varphi_{2m} \end{bmatrix}, \quad \mathbf{q}_{um}^u = \begin{bmatrix} u_{1m} \\ u_{2m} \end{bmatrix}, \quad \mathbf{q}_{um}^v = \begin{bmatrix} v_{1m} \\ v_{2m} \end{bmatrix},
 \end{aligned}
 \tag{3}$$

$$\begin{aligned}
 Y_{um}^u(y) &= \sin\left(m\pi \frac{y}{a}\right) = Y_{wm}, \\
 Y_{um}^v &= Y_{u,ym}^u = \left(\frac{m\pi}{a}\right) \cos\left(\frac{m\pi y}{a}\right), m = 1, 2, \dots
 \end{aligned}$$

## 2.2. HCFSM formulation for stability analysis

### Total potential energy

As a preliminary to tracing the equilibrium paths, it is necessary to determine the total potential energy of the structure as a function of the global DOFs. The steps in the computation are detailed discussed in [2].

The total potential energy of a strip is designated  $\Pi$  and is expressed with respect to the local DOFs by the HCFSM.

$$\begin{aligned}
 \Pi = U + W = (U_m + U_b) + W = & \left( 1/2 \int_A \mathbf{q}_u^T \mathbf{B}_{u1}^T \mathbf{D}_{11} \mathbf{B}_{u1} \mathbf{q}_u dA + 1/2 \int_A \mathbf{q}_w^T \mathbf{B}_{w3}^T \mathbf{D}_{22} \mathbf{B}_{w3} \mathbf{q}_w \right. \\
 & \left. + 1/8 \int_A \mathbf{q}_w^T \mathbf{B}_{w2}^T \mathbf{W}^T \mathbf{B}_{w1}^T \mathbf{D}_{11} \mathbf{B}_{w1} \mathbf{W} \mathbf{B}_{w2} \mathbf{q}_w dA + 1/4 \int_A \mathbf{q}_w^T \mathbf{B}_{w2}^T \mathbf{W}^T \mathbf{B}_{w1}^T \mathbf{D}_{11} \mathbf{B}_{u1} \mathbf{q}_u dA + \right. \\
 & \left. + 1/4 \int_A \mathbf{q}_u^T \mathbf{B}_{u1}^T \mathbf{D}_{11} \mathbf{B}_{w1} \mathbf{W} \mathbf{B}_{w2} \mathbf{q}_w dA \right) + \\
 & \left\{ 1/8 \int_A \mathbf{q}_u^{uT} \mathbf{B}_{u2}^{uT} \mathbf{U}^T \mathbf{B}_{u1}^{uT} \mathbf{D}_{11} \mathbf{B}_{u1}^u \mathbf{U} \mathbf{B}_{u2}^u \mathbf{q}_u^u dA + 1/4 \int_A \mathbf{q}_u^{uT} \mathbf{B}_{u4}^{uT} \mathbf{D}_{11} \mathbf{B}_{u1}^u \mathbf{U} \mathbf{B}_{u2}^u \mathbf{q}_u^u dA + \right. \\
 & \left. + 1/4 \int_A \mathbf{q}_u^{uT} \mathbf{B}_{u2}^{uT} \mathbf{U}^T \mathbf{B}_{u1}^{uT} \mathbf{D}_{11} \mathbf{B}_{u4}^u \mathbf{q}_u^u dA + \right. \\
 & \left. + 1/4 \int_A \mathbf{q}_u^{vT} \mathbf{B}_{u5}^{vT} \mathbf{D}_{11} \mathbf{B}_{u1}^v \mathbf{U} \mathbf{B}_{u2}^v \mathbf{q}_u^v dA + 1/4 \int_A \mathbf{q}_u^{uT} \mathbf{B}_{u2}^{uT} \mathbf{U}^T \mathbf{B}_{u1}^{uT} \mathbf{D}_{11} \mathbf{B}_{u5}^v \mathbf{q}_u^v dA + \right. \\
 & \left. + 1/8 \int_A \mathbf{q}_w^T \mathbf{B}_{w2}^T \mathbf{W}^T \mathbf{B}_{w1}^T \mathbf{D}_{11} \mathbf{B}_{u1}^u \mathbf{U} \mathbf{B}_{u2}^u \mathbf{q}_u^u dA + 1/8 \int_A \mathbf{q}_u^{uT} \mathbf{B}_{u2}^{uT} \mathbf{U}^T \mathbf{B}_{u1}^{uT} \mathbf{D}_{11} \mathbf{B}_{w1} \mathbf{W} \mathbf{B}_{w2} \mathbf{q}_w \right. \\
 & \left. - \int_A \mathbf{q}^T \mathbf{A}^T \mathbf{p} dA \right.
 \end{aligned} \tag{4}$$

The multiplication results of the membrane and bending actions in the first bracket of Eq. (4) are uniquely defined and uncoupled, whilst those in second [von Karman assumptions] and third bracket {Green-Lagrange approach} are functions of the displacements  $u_0$ ,  $v_0$  and  $w$ . Consequently, the membrane and bending actions are coupled in many ways.

The conventional and the geometric stiffness matrices are, respectively:

$$\begin{aligned}
 (\hat{\mathbf{K}}_{uu} = \int_A \mathbf{B}_{u1}^T \mathbf{D}_{11} \mathbf{B}_{u1} dA, \hat{\mathbf{K}}_{ww} = \int_A \mathbf{B}_{w3}^T \mathbf{D}_{22} \mathbf{B}_{w3} dA) & \tag{5} \\
 [\tilde{\mathbf{K}}_{ww} = \int_A \mathbf{B}_{w2}^T \mathbf{W}^T \mathbf{G}_1 \mathbf{W} \mathbf{B}_{w2} dA, \tilde{\mathbf{K}}_{wu} = \int_A \mathbf{B}_{w2}^T \mathbf{W}^T \mathbf{G}_2 dA, \tilde{\mathbf{K}}_{uw} = \int_A \mathbf{G}_2^T \mathbf{W} \mathbf{B}_{w2} dA] \\
 \{ \tilde{\mathbf{K}}_{uu}^{uu} = \int_A \mathbf{B}_{u2}^{uT} \mathbf{U}^T \mathbf{G}_3 \mathbf{U} \mathbf{B}_{u2}^u dA, \tilde{\mathbf{K}}_{uu}^{uu*} = \int_A \mathbf{G}_4 \mathbf{U} \mathbf{B}_{u2}^u dA, \\
 \tilde{\mathbf{K}}_{uu}^{uu**} = \int_A \mathbf{B}_{u2}^{uT} \mathbf{U}^T \mathbf{G}_4^T dA, \tilde{\mathbf{K}}_{uu}^{vu} = \int_A \mathbf{G}_5 \mathbf{U} \mathbf{B}_{u2}^u dA, \\
 \tilde{\mathbf{K}}_{uu}^{uv} = \int_A \mathbf{B}_{u2}^{uT} \mathbf{U}^T \mathbf{G}_5^T dA, \tilde{\mathbf{K}}_{wu}^u = \int_A \mathbf{B}_{w2}^T \mathbf{W}^T \mathbf{G}_6 \mathbf{U} \mathbf{B}_{u2}^u dA, \\
 \tilde{\mathbf{K}}_{uw}^u = \int_A \mathbf{B}_{u2}^{uT} \mathbf{U}^T \mathbf{G}_6^T \mathbf{W} \mathbf{B}_{w2} dA \}.
 \end{aligned}$$

where

$$[ \mathbf{G}_1 = \mathbf{B}_{w1}^T \mathbf{D}_{11} \mathbf{B}_{w1}, \mathbf{G}_2 = \mathbf{B}_{w1}^T \mathbf{D}_{11} \mathbf{B}_{u1} ], \quad (6)$$

$$\{ \mathbf{G}_3 = \mathbf{B}_{u1}^{uT} \mathbf{D}_{11} \mathbf{B}_{u1}^u, \mathbf{G}_4 = \mathbf{B}_{u4}^{uT} \mathbf{D}_{11} \mathbf{B}_{u1}^u, \mathbf{G}_5 = \mathbf{B}_{u5}^{vT} \mathbf{D}_{11} \mathbf{B}_{u1}^u, \mathbf{G}_6 = \mathbf{B}_{w1}^T \mathbf{D}_{11} \mathbf{B}_{u1}^u \}$$

The geometric stiffness matrix of structure is built by summing overlapping terms of the component strip matrices; in the same way that conventional stiffness matrix of structure is built by summing terms of the conventional strip matrices using the transformation matrices between the local and global displacements [2].

### Stability equations

Stability equations are derived from the virtual work principle and the strain energy methods. In order to obtain the stability equations from the variational relations, the principle of the stationary potential energy will be invoked. Since the principle of the stationary potential energy states that the necessary condition of the equilibrium of any given state is that the variation of the total potential energy of the considered structure is equal to zero, we have the following relation:

$$\delta \Pi = 0 . \quad (7)$$

We conclude from Eq. (7) that, if the structure is given the small virtual displacements, the equilibrium still persists if an increment of the total potential energy of the structure  $\delta \Pi$  is equal to zero. Eq. (7) is the basis to derive the variational equation of equilibrium of a structure, and it is correct for both the pre- and post-critical deformation states. Eq. (7) is satisfied for an arbitrary value of the variations of parameters  $\delta \mathbf{q}_m^T$ . Thus we have the following conditions, which must be satisfied for any harmonic  $m$ :

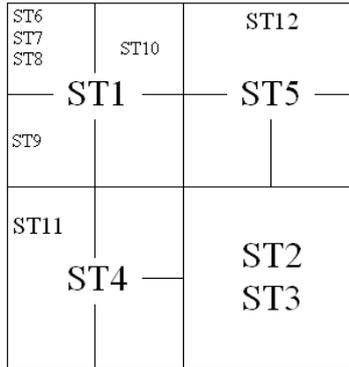
$$\frac{\partial \Pi}{\partial \mathbf{q}_m^T} = \mathbf{0} . \quad (8)$$

Next, we calculate derivatives of the total potential energy of a strip and finally, we get a non-homogeneous and nonlinear set of algebraic equations (9), which are the searched stability equations.

$$\begin{aligned} & (\hat{\mathbf{K}}_{uu} \mathbf{q}_u + \hat{\mathbf{K}}_{ww} \mathbf{q}_w) + [1/2 \tilde{\mathbf{K}}_{ww} \mathbf{q}_w + 1/2 \tilde{\mathbf{K}}_{wu} \mathbf{q}_u + 1/4 \tilde{\mathbf{K}}_{uw} \mathbf{q}_w] + \\ & + \left\{ \begin{aligned} & 1/2 \tilde{\mathbf{K}}_{uu}^{uu} \mathbf{q}_u^u + 3/4 \tilde{\mathbf{K}}_{uu}^{uu*} \mathbf{q}_u^u + 3/4 \tilde{\mathbf{K}}_{uu}^{uu**} \mathbf{q}_u^u + 1/4 \tilde{\mathbf{K}}_{uu}^{vu} \mathbf{q}_u^u + \\ & + 1/2 \tilde{\mathbf{K}}_{uu}^{uv} \mathbf{q}_u^v + 1/4 \tilde{\mathbf{K}}_{wu}^u \mathbf{q}_u^u + 1/4 \tilde{\mathbf{K}}_{uw}^u \mathbf{q}_w \end{aligned} \right\} - \mathbf{Q} = \mathbf{0} \end{aligned} \quad (9)$$

We can visualize the construction of a strip stiffness matrix, which is composed of twelve block matrices. Assembling block matrices into conventional/geometric stiffness matrix of each strip is performed according to

the scheme presented in Fig. 3, where:  $ST1=\hat{\mathbf{K}}_{uu}$ ,  $ST2=\hat{\mathbf{K}}_{ww}$ ,  $ST3=\tilde{\mathbf{K}}_{ww}$ ,  $ST4=\tilde{\mathbf{K}}_{wu}$ ,  $ST5=\tilde{\mathbf{K}}_{uw}$ ,  $ST6=\tilde{\mathbf{K}}_{uu}$ ,  $ST7=\tilde{\mathbf{K}}_{uu}^*$ ,  $ST8=\tilde{\mathbf{K}}_{uu}^{**}$ ,  $ST9=\tilde{\mathbf{K}}_{uu}^{vu}$ ,  $ST10=\tilde{\mathbf{K}}_{uu}^{uv}$ ,  $ST11=\tilde{\mathbf{K}}_{wu}^u$  and  $ST12=\tilde{\mathbf{K}}_{uw}^u$  ( $ST5=ST4^T$ ,  $ST8=ST7^T$ ,  $ST10=ST9^T$ ,  $ST12=ST11^T$ ).



**Fig. 3.** Strip stiffness matrix assembling

**Solution of nonlinear equations**

For equilibrium, the principle of stationary potential energy requires that:

$$\mathbf{R} = \partial\Pi/\partial\mathbf{q}^T = [\hat{\mathbf{K}} + \tilde{\mathbf{K}}]\mathbf{q} - \mathbf{Q} = \mathbf{K}\mathbf{q} - \mathbf{Q} = \mathbf{0} . \tag{10}$$

where  $\Pi$  is a function of the displacements  $\mathbf{q}$ , and  $\mathbf{R}$  represent the gradient or residual force vector, which is generally nonzero for some approximate displacement vector  $\mathbf{q}_0$  (the subscript 0 denotes an old value). It is assumed that a better approximation is given by:

$$\mathbf{q}_n = \mathbf{q}_0 + \delta_0 . \tag{11}$$

where subscript  $n$  denotes a new value.

Taylor's expansion of Eq. (10) yields:

$$\mathbf{R}_n = \mathbf{R}(\mathbf{q}_0 + \delta_0) = \mathbf{R}(\mathbf{q}_0) + \bar{\mathbf{K}}_0\delta_0 + \dots = \mathbf{R}_0 + \bar{\mathbf{K}}_0\delta_0 + \dots . \tag{12}$$

where  $\bar{\mathbf{K}}_0 = \partial\mathbf{R}/\partial\mathbf{q}$  is the matrix of second partial derivatives of  $\Pi$  calculated at  $\mathbf{q}_0$  (i.e. the tangent stiffness matrix or Hessian matrix). Setting Eq. (12) to zero and considering only linear terms in  $\delta_0$  gives the standard expression for Newton-Raphson iteration:

$$\delta_0 = -\bar{\mathbf{K}}_0^{-1}\mathbf{R}_0 . \tag{13}$$

Using this approach, a further iteration yields:

$$\delta_n = \bar{\mathbf{K}}_n^{-1} \mathbf{R}_n \quad (14)$$

where  $\bar{\mathbf{K}}_n = \partial \mathbf{R} / \partial \mathbf{q}$  at  $\mathbf{q}_n$ .

In addition, the blocks of the conventional and geometric tangent stiffness matrix of each strip are:

$$\hat{\mathbf{K}} = \begin{bmatrix} \hat{\mathbf{K}}_{uu} & \mathbf{0} \\ \mathbf{0} & \hat{\mathbf{K}}_{ww} \end{bmatrix}. \quad (15)$$

$$\begin{aligned} \tilde{\mathbf{K}} = & \begin{bmatrix} \mathbf{0} & 1/2 \tilde{\mathbf{K}}_{uw} \\ 1/2 \tilde{\mathbf{K}}_{wu} & 3/2 \tilde{\mathbf{K}}_{ww} \end{bmatrix} + \\ & + \begin{bmatrix} 3/2 \tilde{\mathbf{K}}_{uu}^{uu} + 3/2 \tilde{\mathbf{K}}_{uu}^{uu*} + 3/2 \tilde{\mathbf{K}}_{uu}^{uu**} & 1/2 \tilde{\mathbf{K}}_{uu}^{uv} \\ & \mathbf{0} \end{bmatrix} + \begin{bmatrix} \mathbf{0} & 1/2 \tilde{\mathbf{K}}_{uw}^u \\ 1/2 \tilde{\mathbf{K}}_{wu}^u & \mathbf{0} \end{bmatrix} \end{aligned} \quad (16)$$

Comparing these expressions with Eq. (9), it is apparent that the conventional stiffness matrix remains unchanged, while the geometric tangent matrix becomes symmetrical.

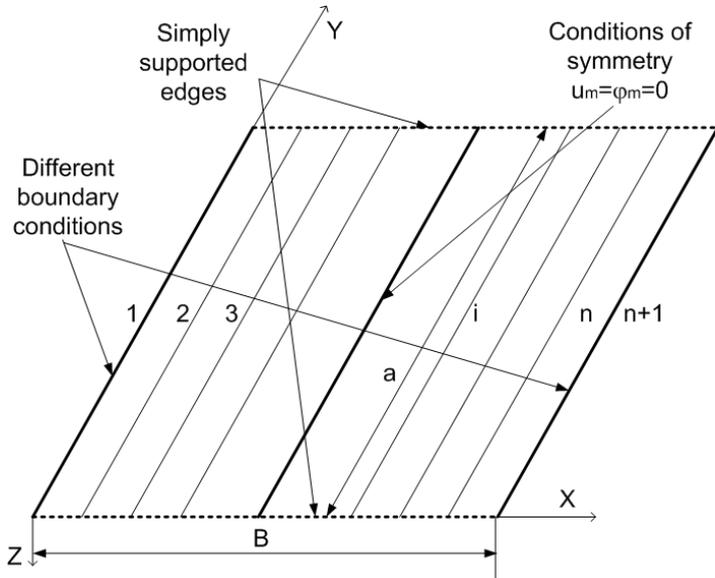
### Static buckling

The loss of stability of static equilibrium states of structures subjected to conservative loads is in general known as static buckling of the structure. For conservative systems, the principle of minimum of the total potential energy can be used to test the stability of a structure (static equilibriums are extremes of the total potential energy). The Hessian with respect to the local DOFs is denoted as the tangent stiffness matrix, of each strip i.e.:

$$\bar{\mathbf{K}} = \left[ \hat{\mathbf{K}} + \tilde{\mathbf{K}} \right]. \quad (17)$$

The (local) stability of equilibrium states of conservative systems by HCFSM can be assessed by looking at the eigenvalues of the tangent stiffness matrix of structure  $\bar{\mathbf{K}}_{(DOFs-r(n+1))(DOFs-r(n+1))}$  with  $(n+1)$  nodal lines, which are all real, since tangent stiffness matrix of the strip is a symmetric matrix.

Fig. 4 shows a rectangular plate divided into  $(n)$  finite strips with  $(n+1)$  nodal lines, where the plate is assumed to be simply supported on edges  $y=0$  and  $y=a$ .



**Fig. 4.** Discretized plate with a pair of simply supported edges

Let  $\lambda_i$  denote the  $i$ th eigenvalue of

$$\bar{\mathbf{K}}_{(DOFs-r(n+1))(DOFs-r(n+1))} \quad (18)$$

Based on theorems of Lagrange-Dirichlet and Lyapunov [14] it can be concluded that an equilibrium state is stable if all  $\lambda_i > 0$ , while an equilibrium state is unstable if one or more  $\lambda_i < 0$ . If along a load-path (IINCS=1, NINCS), at some equilibrium state one or more  $\lambda_i = 0$ , this equilibrium state is denoted as a critical state. Static buckling refers in general to case where, starting from some stable state, a critical state is reached along the load-path.

In general, static buckling is solved by solving Eq. (9) for a varying load  $P$  with for example some sort of numerical path-following routine [2], while simultaneously tracking the eigenvalues of the tangent stiffness matrix of structure given by Eq. (18). Buckling occurs where the matrix becomes singular.

### 3. HCFSM Software Parallelization

#### 3.1. HCFSM Software

The HCFSM software enables large displacement stability analysis of orthotropic prismatic shells with simply supported boundary conditions along

the diaphragm-supported edges. It consists of four modules as shown in Fig. 5.

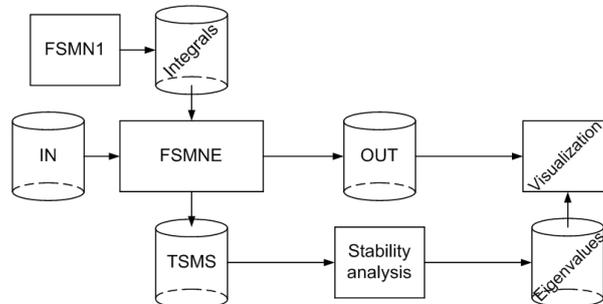


Fig. 5. Overview of HCFSM software

The first module, called FSMN1 program, calculates the values of the integrals using Romberg integration, and stores them in the memory for later computations. The second module, called FSMNE program, calculates displacements and inner forces for each load-part by solving the stability equation (9). The stored integrals are used for the computation of each strip tangent stiffness matrix in FSMNE. The corresponding tangent stiffness matrices of the structure (TSMS) are prepared for the solution of the stability equation and also saved in a file to be analyzed by the third module, called Stability analysis program. The Visualization module provides comparative graphic presentation of the results.

### 3.2. Scope of FSMN1 program parallelization

The number of integral values is proportional to the power of four of the number of harmonics. For a large number of harmonics computation of the integral values can take a substantial amount of time, even with modern hardware resources. However, calculations of the integral values are independent and can be done in parallel.

To avoid repeating computations the integral values are stored in a file once they are calculated. In this way, every time these integral values are needed for the stiffness matrices calculations, they are not calculated but read from the file.

To avoid numerous readings of the file, the integral values can be read from the file just once and placed in the memory (RAM). This approach significantly reduces the execution time (compared to the previous one) because all integral values are used for every strip stiffness matrix calculation in the every iteration and for every load. The drawback of this approach is that it is not generally applicable as the number of integral values outgrows the capacity of available RAM for higher number of harmonics.

The general solution should provide (reasonably fast) stiffness calculation for any (reasonable) number of harmonics. In our vision this can be achieved

by enabling heterogeneous computation where applications utilize the power of both CPU and GPU. These units are treated as separate devices with their own memory spaces allowing for simultaneous computation on both CPU and GPU without contention for memory resources. In particular, using CUDA programming model we are in position to divide independent computations of integral values into different functions (kernels) executed in parallel on general purpose GPUs. Such generated values are used to feed up FSMNE program running on the host processors (CPUs). The globally shared memory located on a GPU device is used to transfer data between the host and device for kernel input and output [6].

For high performance of the system the memory bandwidth between CPU and GPGPU is of crucial importance. To achieve faster communication pipelining we use double buffering technique for the calculation of integral values. Integral values are calculated and placed in two buffers that are used one after the other. At the beginning both buffers are empty. The kernel programs (executing on GPGPU) starts filling the first buffer. When the first buffer is full, FSMNE program (executing on CPU) starts reading integrals from it. At the same time the kernel programs write the next block of integral values to the second buffer. When the first buffer is exhausted, FSMNE program continues to read integrals from the second buffer if it is filled in the mean time. Otherwise, FSMNE program waits for it to be filled. Eventually the kernel programs write the next block of integral values to the second buffer and FSMNE program continues reading them.

The main advantage of using CUDA programming model described above is that it does not require any disk space or memory for storing calculated integral values [8]. Also, it can generate integral values up to 40 times faster than the corresponding sequential approach. However, for efficient implementation of this model we must assure wait-free communication between CPU and the globally shared memory and that represents a major research challenge.

### **3.3. Scope of FSMNE program parallelization**

The essence of FSMNE program can be described as solution of the stability equation (9). The program takes as input parameters the definition of the structure along with the planned load distribution. As a result the program produces the displacement vectors and the vectors of inner forces. The solution for each load-part is obtained using an iterative approach as described in chapter 2.2. Each iteration includes solving the system of the stability equations (SSE) having in mind that TSMS element values depend on the displacements calculated in the previous iteration.

The computational complexity of the problem depend on the number of strips (NELEM) and the number of harmonics (NTERM). Analysis of the program execution reveals that the most time consuming part represents the computation of TSMS, whereas much shorter time is spent on finding the solution of SSE by the method of Gaussian elimination [9]. As the number of

strips grows the time necessary for finding the solution of SSE becomes longer. However, the stiffness matrices computation of separate strips stays dominant.

TSMS is composed of the stiffness matrices for each strip. Computation of these stiffness matrices can be based on the von Karman or the Green-Lagrange prediction. The higher complexity of the Green-Lagrange formulation [9] comes from the fact that each stiffness matrix is formed from the 12 block matrices according to the scheme from Fig. 3. The von Karman prediction requires only the first five block matrices. The block matrices are named by the acronyms ST<sub>i</sub> (i=1, ... , 12).

Computations of the stiffness matrices of separate strips, as well as computations of their parts (different ST<sub>i</sub> block matrices), are mutually independent. Therefore, due to its dominant part in the execution time of FSMNE program, they offer natural bases for FSMNE program parallelization. Computation of the separate strip stiffness matrices is suitable for application of multicomputer – MPI approach [7, 9], while computations of the separate ST<sub>i</sub> block matrices are suitable for application of multiprocessor – OpenMP approach [9].

### MPI parallelization

The multicomputer based computation of the stiffness matrices of separate strips implies engagement of different node computers to calculate the stiffness matrices of different strips. By default, one of the nodes takes the role of “master” while others become “slaves”. Each slave node does the same. It computes the elements of the stiffness matrices for a particular range of strips. The master node initiates these computations, collects the results from the slave nodes, composes TSMS and performs other non-parallelized tasks (e.g. solving the SSE).

Parallel execution of FSMNE program requires communication between the master and the slave nodes. Therefore, the total parallel FSMNE program execution time ( $TP_{total}$ ) includes the communication time ( $TP_c$ ), the stiffness matrices computation time ( $TP_s$ ) and the rest tasks execution time ( $TP_r$ ), while the total sequential FSMNE program execution time ( $TS_{total}$ ) includes only the stiffness matrices computation time ( $TS_s$ ) and the rest tasks execution time ( $TS_r$ ):

$$TP_{total} = TP_c + TP_s + TP_r . \quad (19)$$

$$TS_{total} = TS_s + TS_r . \quad (20)$$

The effects of FSMNE program parallelization are under influence of various factors: the numbers of strips and harmonics on one side and the number of engaged nodes on the other. It is important to analyze the impact of these factors on the different execution times –  $TP_{total}$ ,  $TP_c$ ,  $TP_s$  and  $TP_r$ . The increase of the strips number is followed by the increase of  $TP_s$  and  $TP_r$ .

due to higher dimensions of the total stiffness matrix and the longer time needed to find the solution of SSE. The same trend connects increase of the harmonics number and increase of  $TP_s$ . Increase of the number of nodes causes increase of  $TP_c$  and decrease of  $TP_s$ . The impact of nodes number increase on  $TP_{total}$  depends on the dominant factor: either  $TP_c$  increase or  $TP_s$  decrease.

From the standpoint of the Amdahl's law [15], the maximum speedup of FSMNE program is limited by the following formulas:

$$\frac{n}{1 + f \cdot (n-1)}, \quad (21)$$

$$n = \frac{TS_s}{TP_s}, \quad f = \frac{TS_r}{TS_{total}}.$$

where  $f$  ( $0 < f < 1$ ) is the fraction of time spent in the serial part (before the improvement) and  $n$  represents the speedup of the parallel part of FSMNE program.

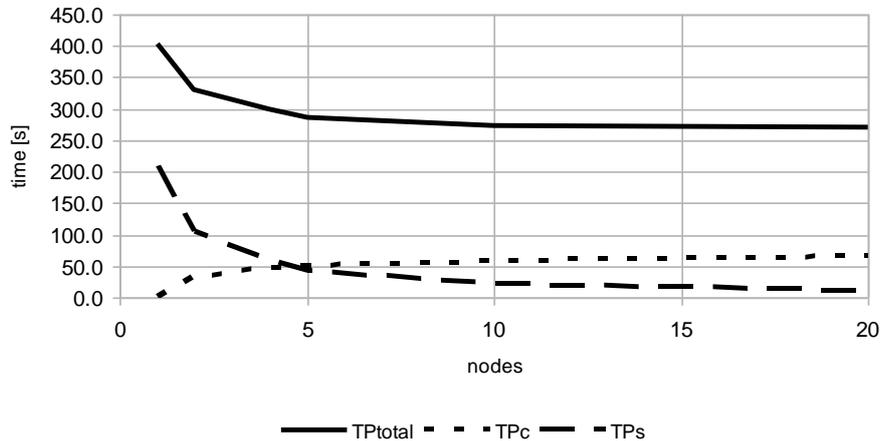
From the standpoint of the parallelization effects it is important to distribute load evenly on multicomputer nodes. The maximum degree of parallelization is achieved when the strip number is divisible by the node number. In contrary, when the node load is not even, the (theoretic) speedup depends on the heaviest loaded node due to the effect of barrier synchronization. For example, in the case with ten strips, the speedup for two nodes is 2, the speedup for three nodes is 2.5, the speedup for four nodes is 3.3, the speedup for five nodes is 5, and the speedup for six to nine nodes is again 5, while the speedup for ten nodes is 10.

Effects of MPI parallelization of FSMNE program for the von Karman and the Green-Lagrange prediction are presented on Fig. 6. The presented results are related to the example of prismatic shell in the form of folded plate cross-section [9]. This shell is divided into 20 strips, and the computation is done for 31 harmonic.

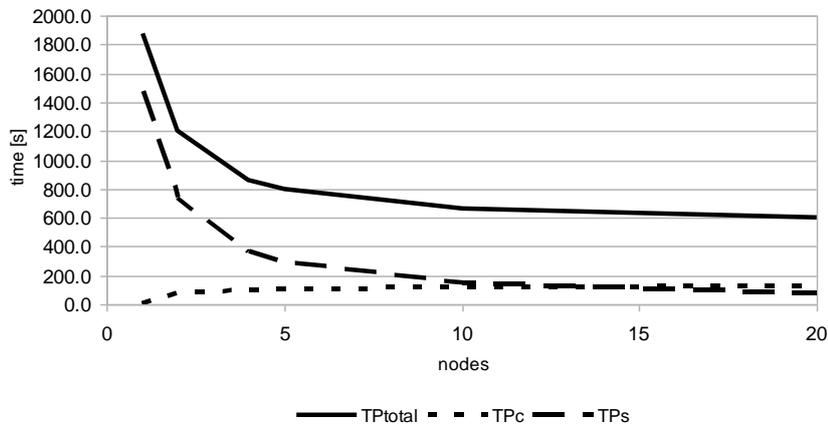
From Fig. 6 follows that in the case of the von Karman prediction  $TP_{total}$  decreases when the node number increases due to the fact that  $TP_s$  represents the significant part of  $TP_{total}$ , so the effects of parallelization of stiffness matrices computation are noticeable. These effects are not canceled by the increase of  $TP_c$  that follows the node number increase. Therefore MPI parallelization of von Karman prediction is valid in the discussed example.

In the case of the Green-Lagrange prediction  $TP_{total}$  significantly decreases when the node number increase, due to the fact that  $TP_s$  is the prevalent part of  $TP_{total}$ . Thus, we can expect that the effects of parallelization of the stiffness matrices computation are very high. These effects are not under substantial influence of the  $TP_c$  increase. Therefore MPI parallelization of the Green-Lagrange prediction is obviously useful in the discussed example.

von Karman



Green-Lagrange



**Fig. 6.**  $TP_{total}$ ,  $TP_c$  and  $TP_s$  for the von Karman and the Green-Lagrange predictions (the presented results are obtained on single CPU core per node multicompiler)

Fig. 6 suggests that in the case of the von Karman, as well as the Green-Lagrange prediction, increase of the node number, when the strip number is high enough, leads to the cancellation of the effect of parallelization of the stiffness matrices computation, due to the  $TP_c$  increase. However, Fig. 7 indicates that the harmonic number increase delays such event, due to increase of the stiffness matrices computation complexity.

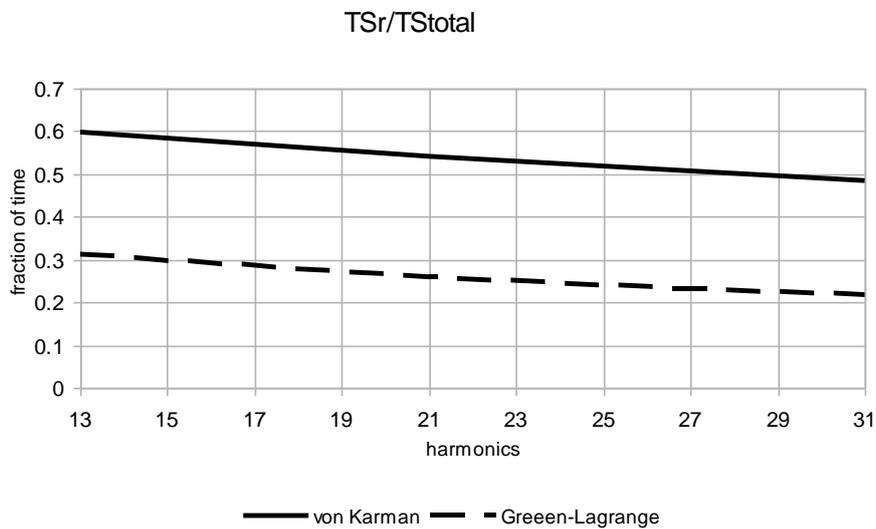
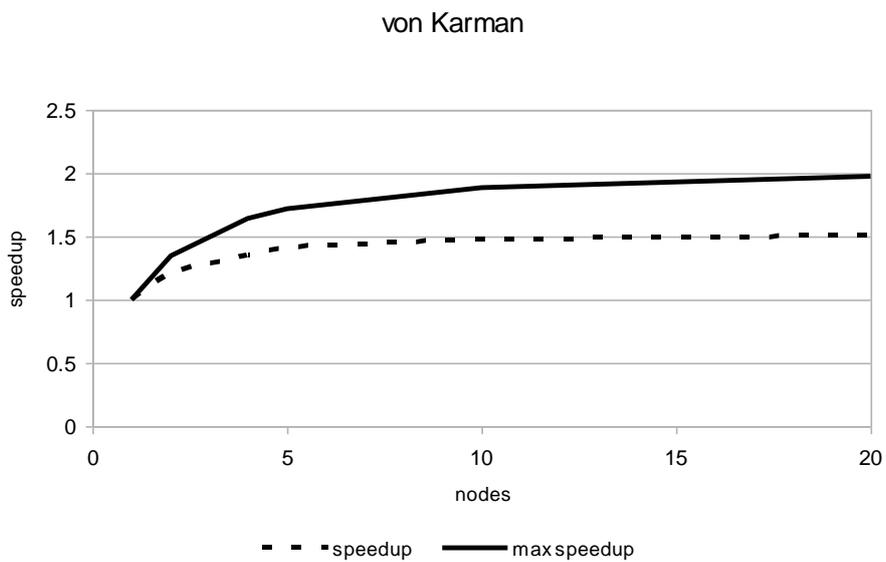
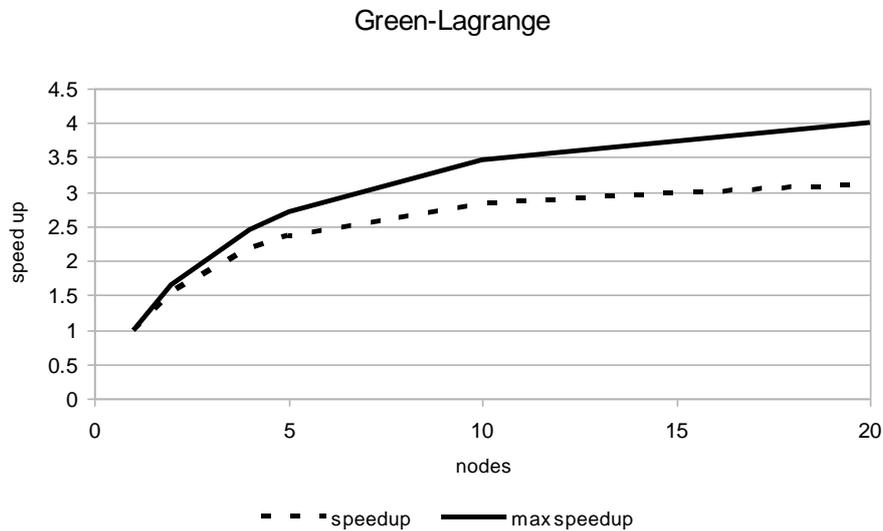


Fig. 7. The impact of the harmonic number increase on the parallelization potential





**Fig. 8.** The speedups of the von Karman and the Green-Lagrange predictions

Fig. 8 suggests that the speedup of the Green-Lagrange prediction is doubled in comparison to the speedup of the von Karman prediction. This is the consequence of the fact that the Green-Lagrange prediction triples the computation time of stiffness matrices in comparison to the von Karman prediction.

### OpenMP parallelization

The multiprocessor based computation of ST<sub>i</sub> block matrices in the Green-Lagrange prediction implies distribution of different ST<sub>i</sub> block matrices computation to different cores of a multi-core processor. The complexity of these block matrices varies which reflects on their execution times. Table 1 contains percentages of the total execution time spent in the ST<sub>i</sub> block matrices.

**Table 1.** Percentage of total execution time spent in the ST<sub>i</sub> block matrices

ST1	ST2	ST3	ST4	ST5	ST6	ST7	ST8	ST9	ST10	ST11	ST12
0.01	0.01	27.38	0.88	0.83	15.15	0.74	0.74	0.74	0.74	24.03	22.56

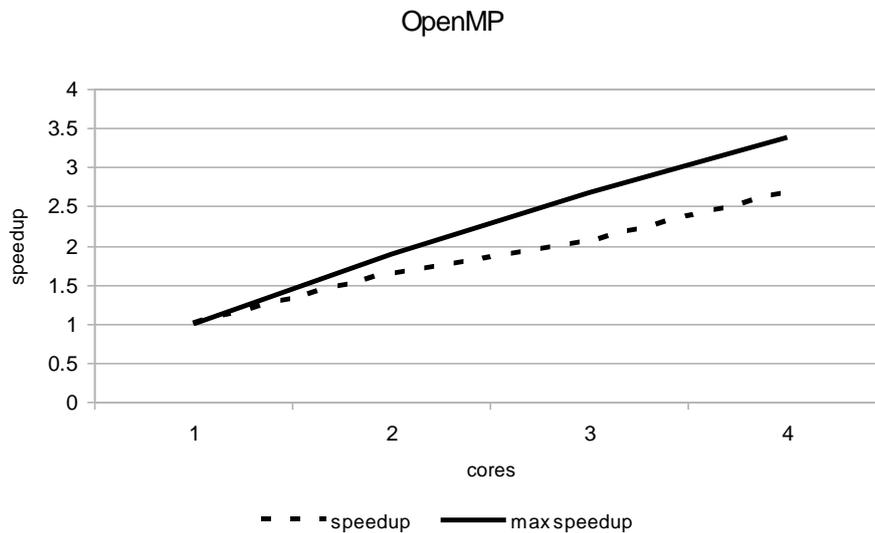
Table 1 suggests that there is no sense to devote a separate core to every ST<sub>i</sub> block matrices, since the computation times of these block matrices are vary disparate and would keep the majority of cores idle. Table 2 presents a possible distribution of the ST<sub>i</sub> block matrices computations to four evenly loaded cores.

**Table 2.** A distribution of the ST<sub>i</sub> block matrices computations to different cores

core	ST <sub>i</sub> block matrices	%
1	ST3	27.38
2	ST1, ST2, ST4, ST5, ST6, ST7, ST8	18.36
3	ST9, ST10, ST12	24.04
4	ST11	24.03

Table 2. implies that almost 94% of the stiffness matrices computation time can be parallelized. The remaining 6% of the time is used for preparation of the ST<sub>i</sub> block matrices computations.

The effects of OpenMP parallelization of FSMNE program for the Green-Lagrange prediction are presented on Fig. 9. The presented results are related to the already mentioned example of prismatic shell.



**Fig. 9.** Results of OpenMP parallelization

The measured speedup follows the Amdahl maximal speedup. Introduction of the third core results in the smaller speedup than expected. The reason is that the data are not evenly distributed among cores anymore, and therefore a good load balancing is not achieved.

It is worth mentioning that as the number of strips grows, the time necessary for finding the solution of SSE grows too. The fact that SSE are solved by master node and our insight to CUDA programming model leads us to the idea of exploring the potential of applying CUDA programming model on the process of solving SSE as the subject of a future work. But for huge number of strips it might be reasonable to explore usage of MPI solver to the same problem.

**Acknowledgements.** The work reported in this paper is a part of the investigation within the research projects: ON174027 "Computational Mechanics in Structural Engineering" and TR 36017 "Utilization of by-products and recycled waste materials in concrete composites in the scope of sustainable construction development in Serbia: investigation and environmental assessment of possible applications", supported by the Ministry for Science and Technology, Republic of Serbia. This support is gratefully acknowledged.

#### 4. Conclusions

This paper presents a parallelization of finite-strip analysis of geometric nonlinear folded-plate structures applying the von Karman and Green-Lagrange strains, which are both characterized by the coupling of all harmonics. The coupling of all series terms highly increases calculation time in an existing finite-strip sequential program when a large number of series terms are used and justify parallelization efforts.

The HCFSM offers good potential for MPI, OpenMP and CUDA parallelization. Therefore it is not surprise that the MPI/OpenMP/CUDA hybrid approach shows good results in the parallelization that are illustrated on the example folded plate cross-section. The future paper will present cumulative effects of contemporary application of all three discussed parallelization techniques. The further investigation will analyze the negative effects of communication bottlenecks that arise with increase of the number of nodes and will try to answer the question how to postpone such negative effects.

#### References

1. Cheung, Y.K., Tham, L.G.: Finite Strip Method. Boca Raton: CRC Press. (1998)
2. Milašinović, D.D. The Finite Strip Method in Computational Mechanics. Faculty of Civil Engineering: University of Novi Sad, Technical University of Budapest and University of Belgrade: Subotica, Budapest, Belgrade. (1997)
3. Chen, H.C., Byreddy, V.: Solving plate bending problems using finite strips on networked workstations. *Computers & Structures*; 62:227-236. (1997)
4. MPI Forum (2009): MPI: A Message-Passing Interface Standard, Version 2.2. <http://www.mpi-forum.org/docs/mpi-2.2/mpi22-report.pdf>.
5. OpenMP Specifications, <http://www.openmp.org/>.
6. Garland, M., Grand, S.L., Nickolls, J., Anderson, J., Hardwick, J., Morton, S., Phillips, E., Zhang, Y., Volkov, V.: Parallel Computing Experiences with CUDA. *IEEE Micro. IEEE Computer Society*, 28:13-27. (2008)
7. Rakić, P., Milašinović, D.D., Živanov, Ž., Hajduković, M.: MPI-CUDA Parallelisation of the Finite Strip Method for Geometrically Nonlinear Analysis. in B.H.V. Topping, P. Iványi, (Editors), "Proceedings of the First International Conference on Parallel, Distributed and Grid Computing for Engineering", Civil-Comp Press, Stirlingshire, UK, Paper 33. doi:10.4203/ccp.90.33. (2009)
8. Rakić, P.S., Milašinović, D.D., Živanov, Ž., Suvajdžin, Z., Nikolić, M. Hajduković, M.: MPI-CUDA parallelization of a finite-strip program for geometric nonlinear

- analysis: A hybrid approach. *Advances in Engineering Software*; 42(5):273-285. (2011)
9. Nikolić, M., Milašinović, D.D., Živanov, Ž., Marić, P., Hajduković M., Borković A., Milaković I.: MPI/OpenMP Parallelisation of the Harmonic Coupled Finite-Strip Method, in P. Iványi, B.H.V. Topping, (Editors), "Proceedings of the Second International Conference on Parallel, Distributed, Grid and Cloud Computing for Engineering", Civil-Comp Press, Stirlingshire, UK, Paper 94. doi:10.4203/ccp.95.94. (2011)
  10. Davies, J.M.: Generalized Beam Theory (GBT) for Coupled Instability Problems, in: Rondal, J. (Ed.), *Coupled Instabilities in Metal Structures, Theoretical and Design Aspects*, CISM Courses and Lectures N. 379, p.151-223, Wien – NY: Springer – Verlag. (2000)
  11. Rasmussen, K.J.R., Hancock, G.J.: Buckling analysis of thin-walled structures: numerical developments and applications, *Progress in Structural Engineering and Materials*, V.2, pp. 359-368. (2000)
  12. Schafer, B.W.: Progress on the Direct Strength Method, *Proceedings of the 16<sup>th</sup> International Specialty Conference on Cold Formed Steel Structures*, Orlando, FL, U.S.A., pp. 647-662. (2002)
  13. Milašinović, D.D.: Geometric non-linear analysis of thin plate structures using the harmonic coupled finite strip method. *Thin-Walled Structures*; 49(2):280-290. (2011)
  14. Bažant, Z.P., Cedolin, L.: *Stability of Structures: Elastic, Inelastic, Fracture, and Damage Theories*. Oxford University Press, Oxford. (1991)
  15. Amdahl, G.M.: Validity of the single processor approach to achieving large scale computing capabilities. *Proc. AFIPS 1967 Spring Joint Computer Conf.* 30 (April), Atlantic City, N.J, 483-485.

**Dr Dragan D. Milašinović**, born 1954. received his Civil Eng. degree from the Univ. of Sarajevo in 1978 and Ph.D. in 1988. He is currently a faculty member at the Univ. of Novi Sad where he is an Prof. of Plates and Shells Theory and Concepts and Applications of Finite Element Analysis.

**Dr Miroslav P. Hajdukovic** is full professor at Faculty of Technical Sciences, University of Novi Sad, the Chair for Applied Computer Science. He graduated in 1977 at the Faculty of Electrical Engineering in Sarajevo, finished post-graduate studies in 1980 at the Faculty of Electrical Engineering in Sarajevo, and earned PhD degree in 1984 at the Faculty of Electrical Engineering in Sarajevo. His scope of interest includes computer architecture, operating systems, distributed systems and compilers.

**Dr Predrag Rakić** is assistant professor at Department of Applied Computer Science at the University of Novi Sad. He received his M.Sc. degree in the operating systems field and Ph.D. in the area wireless sensor networks, both from the University of Novi Sad. His research interests include parallel and distributed computing, generic and template meta programming, data modeling for high performance computing, etc.

Scope of MPI/OpenMP/CUDA Parallelization of Harmonic Coupled Finite Strip Method  
Applied on Large Displacement Stability Analysis of Prismatic Shell Structures

**Miloš Nikolić** is a PhD student/research assistant at École Polytechnique Fédérale de Lausanne (EPFL) since 2010. Prior joining EPFL, he worked as a research assistant at the Chair for Applied Computer Sciences, Faculty of Technical Sciences, University of Novi Sad. His research interest is in the areas of parallel and distributed systems, cloud computing, databases and compilers.

**Lazar Stričević** has been employed as a teaching assistant at the Faculty of Technical Sciences, University of Novi Sad, the Chair for Applied Computer Science since 2004. His interests include computer architecture, operating systems, computer networks, distributed systems.

**Žarko Živanov** is teaching assistant at Faculty of Technical Sciences, University of Novi Sad, the Chair for Applied Computer Science since 2000. His interests include operating systems, computer architecture, compilers and data visualisation.

*Received: August 22, 2011; Accepted: May 11, 2012.*

