Coupled Systems Mechanics, *Vol. 3, No. 1 (2014) 1-25*
DOI: http://dx.doi.org/10.12989/csm.2014.3.1.001

Formulation, solution and CTL software for coupled thermomechanics systems

R. Niekamp[1], A. Ibrahimbegovic*[1,2] and H.G. Matthies[1]

[1]Technical University Braunschweig, D-38092 Braunschweig, Germany
[2]Ecole Normale Superieure de Cachan, 94235 Cachan Cedex, France

(Received January 12, 2014, Revised March 17, 2014, Accepted March 30, 2014)

Abstract. In this work, we present the theoretical formulation, operator split solution procedure and partitioned software development for the coupled thermomechanical systems. We consider the general case with nonlinear evolution for each sub-system (either mechanical or thermal) with dedicated time integration scheme for each sub-system. We provide the condition that guarantees the stability of such an operator split solution procedure for fully nonlinear evolution of coupled thermomechanical system. We show that the proposed solution procedure can accommodate different evolution time-scale for different sub-systems, and allow for different time steps for the corresponding integration scheme. We also show that such an approach is perfectly suitable for parallel computations. Several numerical simulations are presented in order to illustrate very satisfying performance of the proposed solution procedure and confirm the theoretical speed-up of parallel computations, which follow from the adequate choice of the time step for each sub-problem. This work confirms that one can make the most appropriate selection of the time step with respect to the characteristic time-scale, carry out the separate computations for each sub-system, and then enforce the coupling to preserve the stability of the operator split computations. The software development strategy of direct linking the (existing) codes for each sub-system via Component Template Library (CTL) is shown to be perfectly suitable for the proposed approach.

Keywords: coupled thermomechanical system; operator split procedure; nonlinear stability analysis; multiscale in time; code-coupling via CTL

1. Introduction

Driven by various industry applications, the computational mechanics has expanded into a research area much broader than its early-days focus upon the finite element method development (e.g., see Zienkiewich and Taylor 2000) for summary of early works). Very diverse problems are presently of interest for research and expertise of computational mechanics community, such as renewable energy, biomechanics, nanotechnology, predictive home security (e.g., see Oden *et al*. 2003). Thus, the main strive today is directed towards connecting the knowledge from different disciplines, that were long studied separately. In general, the problems of this kind are very much interdisciplinary, extending beyond a single traditional scientific discipline. In other words, a typical formulation presently of interest is defined in terms of multiphysics problems:

*Corresponding author, Professor, E-mail: adnan.ibrahimbegovic@ens-cachan.fr

http://www.techno-press.org/?journal=csm&subpage=7 ISSN: 2234-2184 (Print), 2234-2192 (Online)

thermomechanical coupling, fluid-structure interaction, coupling of mechanics with physical-chemistry, or yet with probability, optimization, identification or control. Moreover, the vast majority of these problems in mathematical interpretation are highly nonlinear evolution problems with constraints. Thus, in trying to provide the most efficient solution procedure for any such problem, one is often prompted to exploit multiscalemodeling and computations, which consider the most appropriate spatial and temporal scales of evolution for each particular sub-system. Providing the best theoretical formulation for any such problem is one of the currently most important challenges that has been addressed in a number of recent works on multiscalemodeling and computations (e.g., see Feyel and Chaboche 2000, Ibrahimbegovic and Markovic 2003, Hughes 2005, Markovic and Ibrahimbegovic 2004, Ladeveze 2005, Oden and Prudhomme 2005, Hautefeuille *et al*. 2013, among others).

In this work, we address a coupled thermomechanical system, as a typical representative of the whole class of multi-physics problems. Of special interest are complex simulations of nonlinear (constrained) evolution of such a system, with respect to presence of different time scales characterizing each of coupled sub-systems.

Our first goal is to provide the sound theoretical formulation and operator split solution procedure. When placed within the framework of semi-discrete approximation, the coupled thermodynamics results with a set of DAE-Differential Algebraic Equations (e.g., see Dennis and Schnabel 1996, Brenan 1996). In a DAE set, the differential equations define the evolution of state variables, and the algebraic equations represent constraints (e.g., plastic admissibility of stress, incompressibility of strain; (seeIbrahimbegovic 2009). For solving such nonlinear time evolution problems by time integration schemes, the monolithic approach (e.g., see Felippa and Park 2004) can be used where all equations in a coupled system are solved simultaneously, requiring dedicated solver for each new coupled system. An alternative strategy is to use an operator split or partitioned approach, where one sub-system is treated at the time, which allows us to exploit the existing solvers for all sub-systems (Markovic *et al*. 2005, Matthies *et al*. 2006).

In this paper, we focus upon the operator split approach to solving nonlinear evolution problems for coupled thermomechanical systems. We will show that the chosen solution strategy provides the possibility for re-using existing codes and software products dedicated to each specific sub-problem, and that it has a great potential for parallelism. Namely, a coupled multiphysics problem can thus be solved by a distributed time integration, where the time integration of the whole DAE-system is distributed to separate time integration methods with possibly different time step for each sub-system. The main challenge in solving the coupled problems in such a distributed manner, is to keep the coupled solution stable. In other words, we seek a more reliable solution than what is possible to compute by explicit schemes with only conditional stability (e.g., see early works on plasticity (Owen and Hinton 1980). The stability condition is well established for classical monolithic approach with time-integration schemes (e.g., see Gear 1971), but such results no longer apply to operator split solution. Namely, the unconditionally stable time-integration schemes for each sub-system, when partitioned in operator split manner, may result with only conditionally stable scheme. The stability condition of operator split procedure was provided in (Arnold 2001) in the framework of multibody dynamics systems (e.g., seeIbrahimbegovic and Taylor 2003). We here adapt this stability criterion to nonlinear evolution of coupled thermomechanics systems. We note in passing that similar studies have been attempted in (Armero and Simo 1992, Farhat *et al*. 1991), but limited to the same time steps used for time-integration schemes for all sub-systems, which is not optimal for different time-scales.

Our second goal pertains to software engineering issues corresponding to multiphysics problems. This is done re-using the existing codes for each sub-system and coupling them directly at linkage phase with the help of software CTL (Niekamp 2001). The corresponding result is a seamless code-coupling with the single executable code for a particular multiphysics problem. The CTL turns an existing computer code, or component, for each sub-system into a C++ template library, where template meta-programming is used to hide as much as possible technical details of unfamiliar components from the programmer. This tool has been successfully used in fluid-structure interaction (Birkin 2010, Matthies *et al*. 2006, Kassiotis *et al*. 2011a,b), multiscale simulation with mesh-in-element approach (Ibrahimbegovic and Markovic 2003, Markovic and Ibrahimbegovic 2005, Niekamp *et al*. 2009), where the interaction is typically over domain boundary. The present work on thermomechanics systems deals with a novel case of coupling within the volume.

The proposed operator split approach does not only provide advantage in simplifying computational procedure for coupled multiscale problems. Re-using a single sub-system existing code to provide the efficient development of coupled computational tools, or computer software, will have even more benefits for bringing this technology to bear upon the current engineering practice. Namely, as the complexity of multiphysics problems grows, many existing codes are found insufficient to meet the new requirement or even completely obsolete. Thus, a number of new developments have sprung in starting the development of new codes for multiphysics and multiscale problems, where higher level programming languages or existing libraries are used to accelerate the software development. Case in point is codes in Java (e.g., Eyheramendy 2008) or in Smalltalk (e.g., Dubois-Pelerin *et al*. 1992, 1998). However, the software products of this kind are not necessarily the most efficient. In fact, although some tentative to increase the code efficiency are made, many such codes are mostly used in academic research environment, very far from real industrial applications.

Here we propose a very different software development strategy, where the final software product is capable to fully integrate existing computer codes. The proposed strategy not only significantly accelerates the final code development, but even more importantly allows to directly include the existing codes that have been extensively tested previously. Last but not least, the proposed strategy can directly provide a very appealing interface to user who are used to a particular software product making them accept the new code release and other novelties much faster.

We note in passing that the same strategies of coupling of existing codes are currently used by interpreters, such as Python (Langtangen 2008) or Matlab (Bindel 2011). However, contrary to such an approach which sacrifices the efficiency in favor of flexibility while coupling typically the executable versions of existing codes, we target here the code coupling that is done at the compilation and linking time, and thus provides truly the single code as the final software product. Such a code, perhaps needless to say, will be both more robust and more efficient than any other code-coupling alternative.

The outline of the paper is as follows: in Section 2 we review the governing equations of thermomechanics coupling and the stability condition for multiscale operator split integration scheme. In Section 3 we discuss the issues pertinent to software implementation and a set of numerical simulations that allow to illustrate an excellent performance of proposed operator split schemes for coupled thermoelastic problems, their stability and gain for distributed time integration. In Section 4, we state the final remarks and conclusions.

2. Coupledthermomechanics system: theoretical formulation and operator split solution procedure

2.1 Linear and nonlinear evolution problems in thermomechanics

For simplicity, we first start with a linear evolution problem of the thermomechanical coupling in 1D setting. The chosen model is a truss-bar of length *L*. One side of the bar is built-in with the imposed zero displacement conditions $u(0,t)=0$ and zero temperature $\theta(0,t)=0$. The displacement $u(L,t)$ and the temperature $\theta(L,t)$ are imposed values that drive the corresponding evolution problem.

We follow (Ibrahimbegovic 2009) in postulating the standard hypotheses for the classical linear thermoelasticity, with the Helmholtz free energy function that can be written as

$$\psi(\varepsilon,\theta)= \frac{1}{2}\varepsilon E\varepsilon-(\theta-\theta_0)m\varepsilon-c\left((\theta-\theta_0)+\theta\ln\left(\frac{\theta}{\theta_0}\right)\right) \tag{1}$$

where $\varepsilon=\partial u/\partial x$ is the classical small strain expression defined for a 1D truss-bar, *E* is Young's modulus, *m* is the thermal stress for unit temperature change, and *c* is the specific heat capacity coefficient. We note in passing that *c* can be defined from the free energy according to

$$c:= \frac{\partial}{\partial\theta}(\psi+\theta s)= \frac{\partial\psi}{\partial\theta}+s+\theta\frac{\partial s}{\partial\theta}=-\theta\frac{\partial^2\psi}{\partial\theta^2}. \tag{2}$$

The proposed free energy form could take into account any eventual large temperature variation. However, we will limit our subsequent study to only a small variation of temperature, around the chosen reference value θ_0. The consistent linearization of the last expression around this reference value of temperature leads to a quadratic form of the free energy density pertaining to linear thermoelasticity with small strains and small temperature change

$$\psi(\varepsilon,\theta)= \frac{1}{2}\varepsilon E\varepsilon-\theta m\varepsilon-\frac{1}{2}\frac{c}{\theta_0}\theta^2 \tag{3}$$

where $(\theta-\theta_0)\rightarrow\theta$ is used to simplify the notations. The free energy function in (3) above gives the following linear constitutive equations for stress σ and entropy *s*

$$\sigma:= \frac{\partial\psi}{\partial\varepsilon}=E\frac{\partial u}{\partial x}-m\theta \mathit{ands}:=-\frac{\partial\psi}{\partial\theta}=\frac{c}{\theta_0}\theta+m\varepsilon \tag{4}$$

The local form of the momentum balance for the quasi-static case considered herein, can then be written as

$$\frac{\partial\sigma}{\partial x}+b=0 \tag{5}$$

The local form of the energy balance is also linearized around the reference value of temperature θ_0, and it leads to the following final result

$$\theta_0 \frac{\partial s}{\partial t} :== c \frac{\partial \theta}{\partial t} + \theta_0 m \frac{\partial \varepsilon}{\partial t} = -\frac{dq}{dx} + r \tag{6}$$

We recall again that the reference temperature θ_0 can be considered as a part of the set of the chosen material properties, and not an additional unknown.

Combining last three results we define the Strong Form of 1D thermoelasticity:

Given: the mechanical loading *b* and thermal loading *r*,

Find: displacement *u*(*x*,*t*) and temperature θ(*x*,*t*) such that

$$\begin{cases} \dfrac{\partial \sigma}{\partial x} + b = 0 \quad ; \quad \sigma = E\dfrac{\partial u}{\partial x} - m\theta \\ c\dfrac{\partial \theta}{\partial t} + m\theta_0 \dfrac{\partial^2 u}{\partial t \partial x} + \dfrac{dq}{dx} - r = 0 \quad ; \quad q = -k\dfrac{d\theta}{dx} \end{cases} \tag{7}$$

For the FEM based solution procedure, these equations are rewritten in the weak form (Ibrahimbegovic 2009)

$$\begin{gathered} \int_L \frac{dw}{dx}\sigma\, dx = \int_L wb\, dx + [w\bar{t}]_{\Gamma_\sigma} \\ \int_L \vartheta c \frac{\partial \theta}{\partial t} dx + \int_L \vartheta m \theta_0 \frac{\partial^2 u}{\partial t \partial x} - \int_L \frac{d\vartheta}{dx} q\, dx = \int_L \vartheta r\, dx + [\vartheta \bar{h}]_{\Gamma_q} \end{gathered} \tag{8}$$

In order to derive the simplest model problem illustrating these ideas, we choose the homogeneous load case and a single 2-node finite element approximation and the lumped capacity matrix, which results with the following set of ordinary differential equations to be solved

$$\begin{cases} \dfrac{E}{L}u(t) - \dfrac{m}{2}\theta(t) \qquad = 0 \\ \dfrac{cL}{2}\dot{\theta}(t) + \dfrac{k}{L}\theta(t) + \dfrac{m\theta_0}{2}\dot{u}(t) = 0 \end{cases} \tag{9}$$

We seek to provide the stability condition of the iterative scheme for coupling partitioned solution for the coupled thermoelasticity problem, for any time discretization for each sub-problem that can accommodate any particular time step. Tacit assumption pertains to the need to integrate by an implicit, unconditionally stable time-stepping scheme the both sub-systems, mechanical and thermal part.

We note that an equivalent result will be obtained for a more general model of 2D/3D elasticity applied to coupled thermomechanics systems. We will consider again the case of quasi-static evolution for the mechanical sub-system along with the transient heat transfer. The problem of this kind is the most useful preparation for inelastic problem studied subsequently. We assume that semi-discretization is carried out by FE method (e.g., see Zienkiewicz and Taylor 2000, Bathe 1996 or Ibrahimbegovic 2009). The weak form of 2D/3D thermoelasticity leads to the following set of ordinary differential equations:

$$\begin{cases} \mathbf{K}_M u(t) - \mathbf{F}_M \theta(t) = 0 \\ \mathbf{M}_T \dot{\theta}(t) + \mathbf{K}_T \theta(t) + \mathbf{F}_T \dot{u}(t) = 0 \end{cases} \tag{10}$$

Where $\mathbf{K}_M$ is the stiffness matrix, $\mathbf{M}_T$ is the heat capacity matrix, $\mathbf{K}_T$ is the heat conductivity matrix, whereas $\mathbf{F}_M$ and $\mathbf{F}_T$ are the coupling matrices in semi-discrete approximation of thermomechanical system. Each of these matrices is defined by the standard finite element assembly procedure (e.g., see Zienkiewicz and Taylor 2000, Bathe 1996) to account for different element contributions (each denoted with superscript '*e*') to either mechanical or thermal sub-system, resulting with:

$$\begin{aligned} \mathbf{K}_M &= A_e \mathbf{K}^e_M;\ \mathbf{K}^e_M = \int_{\Omega_e} \mathbf{B}^T \mathbf{E} \mathbf{B} \, d\Omega \\ \mathbf{F}_M &= A_e \mathbf{F}^e_M;\ \mathbf{F}^e_M = \int_{\Omega_e} \mathbf{B}^T m \mathbf{N} \, d\Omega \\ \mathbf{M}_T &= A_e \mathbf{M}^e_T;\ \mathbf{M}^e_T = \int_{\Omega_e} \mathbf{N}^T c \mathbf{N} \, d\Omega \\ \mathbf{K}_T &= A_e \mathbf{K}^e_T;\ \mathbf{K}^e_T = \int_{\Omega_e} \mathbf{B}^T \mathbf{k} \mathbf{B} \, d\Omega \\ \mathbf{F}_T &= A_e \mathbf{F}^e_T\ \mathbf{F}^e_T = \int_{\Omega_e} \mathbf{N}^T m \theta_0 \mathbf{B} \, d\Omega \end{aligned} \tag{11}$$

Here, $\mathbf{E}$ denotes the elasticity tensor, c is specific heat capacity coefficient, $\mathbf{k}$ is the heat conduction tensor, whereas m and θ_0 are the thermal stress and reference temperature, respectively. Moreover, $\mathbf{N}$ denotes the matrix representation of the shape functions chosen for a particular semi-discrete approximation constructed by finite element, and $\mathbf{B}$ is the corresponding strain displacement matrix (e.g., see Ibrahimbegovic 2009).

In order to derive the stability proof in the sense of (Anrold 2001), we recast the thermoelasticity problem in terms of a set of differential-algebraic equations (DAE), which can be written as

$$\begin{cases} \dot{u}(t) = v(t)\ ; \quad 0 = r_M \big(v(t); g(t)\big) := \mathbf{K}_M v(t) - \mathbf{F}_M g(t) \\ \dot{\theta}(t) = g(t)\ ; \ 0 = r_T \big(\theta(t); g(t); v(t)\big) := \mathbf{M}_T \dot{\theta}(t) + \mathbf{K}_T \theta(t) + \mathbf{F}_T \dot{u}(t) \end{cases} \tag{12}$$

The 1D thermoelasticity problem defined with the simplest semi-discrete approximation also belongs to the class of problems, with

$$\mathbf{K}_M = \frac{E}{L}, \quad \mathbf{F}_M = \frac{m}{2}, \quad \mathbf{M}_T = \frac{cL}{2}, \quad \mathbf{K}_T = \frac{k}{L} \quad and \mathbf{F}_T = \frac{m\theta_0}{2} \tag{13}$$

We note that the stability proof remains valid for an inelastic evolution problem of coupled thermomechanics system. The most general form of a nonlinear coupled thermomechanics problem of direct interest for practical applications will consider inelastic behavior described by the internal variables, such as plastic strain or hardening variables (Ibrahimbegovic 2009). The explicit form of thermoechanical residuals depends on a particular choice of inelastic model (Ibrahimbegovic 2009); however, it always counts the equilibrium equations, evolution equations of internal variables and energy balance, resulting with a set of DAEs. In general, we can write: (i) the mechanical residual r_M for quasi-static evolution in terms of displacement $u(t)$ and internal variables $\xi(t)$, gathering the semi-discrete approximation to the weak form of equilibrium equations and plastic constraint on stress admissibility

$$\begin{cases} \dot{u}(t)=v(t) \quad ; \quad \xi(t)=\zeta(t) \\ 0=r_M\left(u(t);v(t);\xi(t);\zeta(t);\theta(t);g(t)\right) \end{cases} \tag{14}$$

The DAE corresponding to thermal residual r_T can be written in terms of the same state variables, also accounting for the inelastic dissipation as an additional source of heating (Ibrahimbegovic 2009)

$$\begin{cases} \dot{\theta}(\mathrm{t})=\mathrm{g}(\mathrm{t}) \\ 0=\mathrm{r}_\mathrm{T}\left(\theta(\mathrm{t});\mathrm{g}(\mathrm{t});\mathrm{u}(\mathrm{t});\mathrm{v}(\mathrm{t});\xi(\mathrm{t});\zeta(\mathrm{t})\right) \end{cases} \tag{15}$$

The thermomechanical coupling takes place throughout the whole domain Ω. The latter is in contrast with the pertinent work on multibody dynamics systems (Anrold 2001), where the coupling conditions are enhanced by an algebraic equation posed on linked boundaries. In order to place these two cases on the same basis, we assume heres here the coupling algebraic equation valid in the whole domain

$$0 = h(u(t), v(t), \xi(\mathrm{t}), \zeta(t), \theta(t), g(t)) \tag{16}$$

In the following, we consider mechanical equilibrium as coupling algebraic condition $h=r_M$, with the temperature as the coupling variable.

2.2 Stability of operator split method for time integration of coupled thermomechanical problems

For studying the stability of the iterative scheme for enforcing the convergence of partitioned solution of the coupled thermomechanics problem, no particular discretization in time ought to be chosen, other than assumption on unconditional stability of the time-integration scheme for each sub-system. Following (Arnold 2001), the convergence proof can be applied directly the product of FEM-based semi-discretization, presented in previous section.

However, in order to illustrate the most general computational procedure, we will assume that in computing solutions in the interval $\left[0,T_e\right]$ for the two sub-systems, we might use independent solvers on a *windows* (or macro time step) $\left[T_n,T_{n+1}\right]$ for each sub-system. The chosen values T_n are synchronization points where the thermomechanical coupling communication is enforced. The iteration counter for corresponding procedure in each window n is denoted as (k), and the best iterative value solutions can thus be denoted

$$\begin{aligned} uT_{[T_n]}\Big|_{[T_n,T_{n+1}]}(t) &\approx \tilde{u}_n^{(k)}(t) \\ \theta T_{[T_n]}\Big|_{[T_n,T_{n+1}]}(t) &\approx \tilde{\theta}_n^{(k)}(t) \\ \xi T_{[T_n]}\Big|_{[T_n,T_{n+1}]}(t) &\approx \xi_n^{(k)}(t) \end{aligned} \tag{17}$$

It is important to note that $\tilde{u}_n^{(k)}$, $\xi_n^{(k)}$ and $\tilde{\theta}_n^{(k)}$ are functions of time t defined on window $\left[T_n,T_{n+1}\right]$. For the case where time step is the same for each sub-system, we do not need to consider any particular interpolation in time. However, for the present case of using multi-scale methods in time, it is important to choose a way to interpolate each function between synchronization points in order to enforce the matching in algebraic equations r_M, r_T and $h$over the whole macro time-step. Thus, the multi-scale in time can be seen in a certain manner as an increase in order of interpolation.

As illustrated in Fig. 1, enforcing the thermomechanical coupling condition in the operator split method for any such constrained evolution problem corresponds to the Gauss-Seidel iterative algorithm; further. This choice can compare favorably to the classical iterative procedure of Jacobi (e.g., see Ciarlet *et al*. 1989, Golub and VanLoan 1989), since it provides a better convergence rate. Further improvement of convergence can be achieved by Newton's iterative method for enforcing the coupling (e.g., see Matthies *et al*. 2006, Matthies and Steindorf 2003), but requires needs an approximation to off-diagonal terms of the Hessian matrix based on mixed derivatives which are not readily available. Hence, the Gauss-Seidel iterative algorithm seems as the best strategy when one seeks to directly reuse the existing solvers (and codes) for all sub-systems. The only serious drawbacks of the Gauss-Seidel method concern only conditional stability (compared to Newton's method), and the lack of easy parallelism (compared to Jacobi's). The former is the subject of further study in this section, and the latter is studied in the subsequent section.

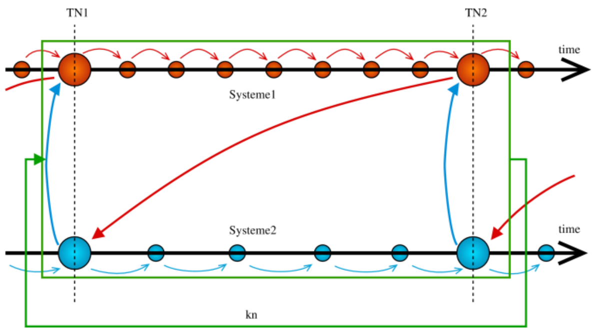

Fig. 1 Coupling algorithm example: Gauss-Seidel method on the window *[T_n,T_{n+1}]*

We study here the stability of the proposed operator split when compute the numerical solution in a typical window for $t \in [T_n, T_{n+1}]$, by analogy with dynamic iteration proposed in (Arnold 2001). We choose to enforce the coupling condition with the first sub-system to be solved, concerning the computation of the mechanical residual or enforcement of the equilibrium equations. Considering a typical iteration sweep (*k*), we can write

$$\begin{aligned} 0 = r_M \left(\tilde{u}_n^{(k)}(t); \tilde{v}\tilde{u}_n^{(k)}(t); \xi\tilde{u}_n^{(k)}(t); \zeta\tilde{u}_n^{(k)}(t); \theta\tilde{u}_n^{(k-1)}(t); \tilde{g}\tilde{u}_n^{(k-1)}(t) \right) \\ 0 = h_M \left(\tilde{u}_n^{(k)}(t); \tilde{v}\tilde{u}_n^{(k)}(t); \xi\tilde{u}_n^{(k)}(t); \zeta\tilde{u}_n^{(k)}(t); \tilde{\theta}\tilde{u}_n^{(k-1)}(t); \tilde{g}\tilde{u}_n^{(k-1)}(t) \right) \end{aligned} \tag{18}$$

The second sub-system is solved with thermal residual

$$0 = r_T \left(\tilde{u}_n^{(k)}(t); \tilde{v}\tilde{u}_n^{(k)}(t); \xi\tilde{u}_n^{(k)}(t); \zeta\tilde{u}_n^{(k)}(t); \tilde{\theta}\tilde{u}_n^{(k)}(t); \tilde{g}\tilde{u}_n^{(k)}(t) \right) \tag{19}$$

Even if the numerical integration of $u(t)$, $\xi(t)$ and$\theta(t)$ for each sub-system is guaranteed to be stable and convergent in each window, the coupled algorithm for the whole system does not necessarily remain stable depending upon the strength of the coupling. In order to guarantee a stable error propagation from one macro-scale window of coupled system time-integration to another, the contractivity condition (Arnold 2001) has to be fulfilled with $\alpha<1$. For the chosen order of the resolution of different sub-systems (Arnold 2001, Matthies *et al*. 2006), the coefficient α reads:

$$\alpha \quad =$$

$$\max_{t \in [0, T_e]} \left\| \left| \left[\frac{\partial r_T^{(k)}}{\partial g_n^{(k)}} \right]^{-1} \frac{\partial r_T^{(k-1)}}{\partial g_n^{(k-1)}} \left[\frac{\partial h^{(k)}}{\partial \dot{u}_n^{(k)}} \left[\frac{\partial r_M^{(k)}}{\partial \dot{u}_n^{(k)}} \right]^{-1} \frac{\partial r_M^{(k-1)}}{\partial g_n^{(k-1)}} \right]^{-1} \frac{\partial h^{(k)}}{\partial g_n^{(k)}} \right\| \right| \tag{20}$$

For the chosen multi-scale computations for coupled problem of linear thermoelasticity. with the Gauss-Seidel iterative algorithm in a typical window $[T_n, T_{n+1}]$ and at (*k*)-th iterative sweep we get

$$\left\{ \begin{aligned} 0 = r_M \left(v_n^{(k)}(t); g_n^{(k-1)}(t) \right) \quad &:= \quad \mathbf{K}_M v_n^{(k)}(t) - \mathbf{F}_M g_n^{(k-1)}(t) \\ 0 = h \left(v_n^{(k)}(t); g_n^{(k)}(t) \right) \quad &:= \quad \mathbf{K}_M v_n^{(k)}(t) - \mathbf{F}_M g_n^{(k)}(t) \\ 0 = r_T \left(\theta_n^{(k)}(t); g_n^{(k)}(t); v_n^{(k)}(t) \right) &:= \mathbf{M}_T g_n^{(k)}(t) + \mathbf{K}_T \theta_n^{(k)}(t) + \mathbf{F}_T v_n^{(k)}(t) \end{aligned} \right. \tag{21}$$

We can show explicitly that the proposed isothermal split is only conditionally stable. Namely, from (21) above, we can obtain:

$$\begin{aligned}
&\frac{\partial r_M^{(k)}}{\partial v_n^{(k)}} = \mathbf{K}_M\,; \quad \frac{\partial r_T^{(k)}}{\partial g_n^{(k)}} = \mathbf{M}_T\,; \quad \frac{\partial h^{(k)}}{\partial v_n^{(k)}} = \mathbf{K}_M \\
&\frac{\partial r_M^{(k-1)}}{\partial g_n^{(k-1)}} = -\mathbf{F}_M;\ \frac{\partial r_T^{(k-1)}}{\partial g_n^{(k-1)}} = \mathbf{F}_T\left[\mathbf{K}_M\right]^{-1}\mathbf{F}_M;\ \frac{\partial h^{(k)}}{\partial g_n^{(k)}} = -\mathbf{F}_M
\end{aligned} \tag{22}$$

With these results on hand, the stability condition can be expressed with

$$\begin{aligned}
\alpha &= \left|\left|\left[\mathbf{M}_T\right]^{-1}\mathbf{F}_T\left[\mathbf{K}_M\right]^{-1}\mathbf{F}_M\left[\mathbf{K}_M\left[\mathbf{K}_M\right]^{-1}\mathbf{F}_M\right]^{-1}\mathbf{F}_M\right|\right| \\
&= \left|\left|\left[\mathbf{M}_T\right]^{-1}\mathbf{F}_T\left[\mathbf{K}_M\right]^{-1}\mathbf{F}_M\right|\right| < 1
\end{aligned} \tag{23}$$

For the simplest coupled thermomechanics model we defined in terms of 1D thermoelasticity, the stability criterion in (23) reduces to

$$\alpha := \left|\left[\frac{cL}{2}\right]^{-1}\frac{\theta_0 m}{2}\left[\frac{E}{L}\right]^{-1}\frac{m}{2}\right| < 1 \tag{24}$$

The same criterion can be stated in terms of non-dimensional parameterε, in terms of:

$$\varepsilon := \frac{m^2\theta_0}{Ec} < 2 \tag{25}$$

We note in passing that the same stability condition can be provided for 3D thermoelasticity in terms of the non-dimensional parameter $\varepsilon = (3\lambda+2\mu)^2 m^2\theta_0 / c(\lambda+2\mu)$. It turns out that such a stability criterion is identical to the linearized A-stability criterion (Chorin *et al.* 1978), as proposed in (Armero and Simo 1992) for thermoelasticity and in (Armero and Simo 1993) for thermoplasticity. Besides linear stability, all these works considered the same time steps for integrating both mechanical and thermal sub-systems in computing u_{n+1} and θ_{n+1}, respectively. For the simplest case of discrete approximation of 1D thermoelasticity with a single 2-node element, the isothermal split would lead to

$$\left\{\begin{aligned}
&\frac{E}{h}u_{n+1} - \frac{m}{2}\theta_n = 0 \\
&\frac{ch}{2\Delta t}\left(\theta_{n+1} - \theta_n\right) + \frac{k}{h}\theta_{n+1} + \frac{m\theta_0}{2\Delta t}\left(u_{n+1} - u_n\right) = 0
\end{aligned}\right. \tag{26}$$

We further rewrite this set of equations in matrix notation, which allows us to define the amplification matrix

$$\begin{bmatrix} u_{n+1} \\ \theta_{n+1} \end{bmatrix} = \begin{bmatrix} \frac{E}{h} & 0 \\ \frac{m\theta_0}{2\Delta t} & \frac{ch}{2\Delta t} + \frac{k}{h} \end{bmatrix}^{-1} \begin{bmatrix} 0 & \frac{m}{2} \\ \frac{m\theta_0}{2\Delta t} & \frac{ch}{2\Delta t} \end{bmatrix} \begin{bmatrix} u_n \\ \theta_n \end{bmatrix} \tag{27}$$

The stability condition requires that the determinant of the amplification matrix remains bounded by 1, which can further be expressed as

$$\left[\frac{E}{h} \left(\frac{k}{h} + \frac{ch}{2\Delta t} \right) \right]^{-1} \frac{m^2\theta_0}{4\Delta t} \leq 1 \Leftrightarrow \frac{m^2\theta_0}{Ec} - 2 \leq 4 \, \frac{k}{c} \, \frac{\Delta t}{h^2} \tag{28}$$

It thus follows that such a A-stability criterion is automatically verified if$\varepsilon = \frac{m^2\theta_0}{Ec} < 2$, which confirms our result in (Ibrahimbegovic *et al*. 2005)for the same time step. Still, our result remains more general than the result in (Armero and Simo 1992, 1993), since it also holds for multi-scale problem in time with different time step for any particular sub-system.

The conditional stability requirement can be eliminated by appealing to adiabatic split for thermoelasticity. This is one of possible solutions for conditional stability drawback of the isothermal split, next to more simple equations reordering (Matthies *et al*. 2006) or more costly Newton or quasi-Newton iterative procedure (Matthies and Steindorf 2003, Matthies *et al*. 2006). The adiabatic split (Armero and Simo 1992, Ibrahimbegovic 2009) is fully equivalent as the constraint splitting, as advocated in (Arnold 2001), that can be used to recover the unconditional stability of coupled solution. We propose to use the following

$$G_n^{(k)}(t) = (\mathbf{I}_d - \mathbf{A}(t)) g_n^{(k)}(t) + \mathbf{A}(t) g_n^{(k-1)} \tag{29}$$

The preconditioning of this kind can be seen as an adiabatic split (Armero and Simo 1992, 1993, Ibrahimbegovic 2009), which implies that during the mechanical phase of calculation a fictitious temperature evolution occurs due to difference between$G_n^{(k-1)}(t)$ and $G_n^{(k)}(t)$. The computation in thermal phase is then started from computed value in mechanical phase. An important point to note is that the cost of calculation remains the same as for the isothermal split, with only a slight difference regarding implementation. In that manner, the proposed preconditioning can be considered optimal for this kind of computation.

With such a choice, the coupled problem to be solved reduces to the following set of constraint equations

$$\begin{cases} 0 = r_M \left(v_n^{(k)}; G_n^{(k)}(t) \right) \\ 0 = h \left(v_n^{(k)}; G_n^{(k)}(t) \right) \\ 0 = r_T \left(\theta_n^{(k)}(t); G_n^{(k)}(t); v_n^{(k)}(t) \right) \end{cases} \tag{30}$$

We thus obtain a new value for the contractivity constant α, which is defined as

$$\alpha = \left\| \left(\frac{\partial r_T^{(k)}}{\partial \theta_n^{(k)}} \mathbf{I}_d - \mathbf{A}(t) \right)^{-1} \frac{\partial r_T^{(k)}}{\partial \dot{\theta}_n^{(k)}} \mathbf{A}(t) + \left[\frac{\partial r_T^{(k)}}{\partial \dot{\theta}_n^{(k)}} \right]^{-1} \frac{\partial r_T^{(k-1)}}{\partial \dot{\theta}_n^{(k-1)}} \left[\frac{\partial h^{(k)}}{\partial \dot{u}_n^{(k)}} \left[\frac{\partial r_M^{(k)}}{\partial \dot{u}_n^{(k)}} \right]^{-1} \frac{\partial r_M^{(k-1)}}{\partial \dot{\theta}_n^{(k-1)}} \right]^{-1} \frac{\partial h^{(k)}}{\partial \dot{\theta}_n^{(k)}} \right\| \tag{31}$$

The best choice for pre-conditioning with $\mathbf{A} = -\varepsilon \mathbf{I}_d$ will ensure that the computation for the adiabatic split remains always stable, for any values of material properties and for any value of $\varepsilon = \frac{m^2 \theta_0}{Ec}$.

3. Software development via CTL and numerical simulations

3.1 Software coupling by using CTL for Dirchlet-to-Neumann case

There are several alternatives for enforcing the unconditional stability requirement of coupled mechanics systems computations, starting from simple equations reordering in Gauss-Seidel scheme (Matthies 2006) to more costly Newton or quasi-Newton iterative procedure ((Matthies 2003, 2006). The CTL implementation of these methods is here discussed in the simplest possible context of 1D structure-structure coupled system, as shown in Fig. 2.

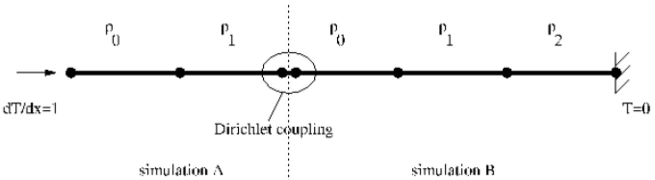

Fig. 2 Simple coupled 1D-system

The CTL-based code coupling is illustrated for the 'standard' coupling case of Dirichlet-to-Neumann coupling of two mechanical sub-systems, which is typical of many coupled systems. The most prominent example of this kind is fluid-structure interaction (Kassiotis 2011a, b), where the Dirichlet condition is enforced on fluid with mesh motion in ALE imposed by structure, and the Neumann condition is imposed on structure with imposed traction forces exerted by fluid. In such context, the stability of the operator split for coupling enforcement depends only upon the ratio of the stiffness between two sub-systems for stationary, or upon the ratio between their densities for non-stationary case. As long as the stiffness (respectively density) of the displacement controlled sub-system with Dirichlet boundary condition is small compared to the other sub-system controlled with Neumann boundary, the standard iterative schemes, such as

Gauss-Seidel or Jacobi-like will converge. In the opposite case, when starting coupled iteration with a soft sub-system, the coupled iterations will diverge. This fact is intuitively clear, since controlling a stiff structure by an inaccurate imposed displacement leads to a force response with an even larger error and on the other hand controlling a flexible structure by an inaccurate force leads to a displacement answer with enlarged error.

In order to implement such a coupling using the CTL, we first have to define software-interfaces which describe the functionality needed for the chosen coupling scheme. Thus, any useful simulation code should at least implement an interface ’stationarySimulationCI’, which can be written as:

```
interfacestationarySimulationCI
{
// initialisation with input-file
stationarySimulationCI (const string /*input-file*/);
void solve();
voidget_state(array<real8>/*state*/) const;
};
```

Using Gauss-Seidel or Jacobi-type solution schemes, the force controlled part of a stationary coupled simulation needs at least to implement the interface:

```
interfacestationaryNeumannCI: extends stationarySimulationCI
{
voidset_load(const array<real8> /*load*/);
voidget_disp(array<real8> /*disp*/);
};
```

The clause extends stationary Simulation CI has the effect, that the methods in stationarySimulationCI get also part of stationary Neumann CI. In this case, the sub-system controlled by Dirichlet-conditions needs at least to implement the following interface:

```
interfacestationaryDirichletCI: extends stationarySimulationCI
{
voidset_disp(const array<real8> /*disp*/);
voidget_load(array<real8> /*load*/);
};
```

The next step is to get the software components for the left and right sub-systems. Here, we use for both sub-systems the FEAP code, as described in detail in (Niekamp *et al*. 2009, Markovic *et al*. 2005, Ibrahimbegovic *et al*. 2014). The only requirement on the simulation codes to be capable for the transfer to software components are:

- its functionality can be split into the functions listed in the interfaces as shown above
- a method invocation has no more side-effects than those given in the interface.

The latter implies that a method declared as ’const’ may not change the state of the simulation, or in other words, the mathematical meaning of a function (same argument, same result) must be ensured.

At the end, we need an implementation of an iterative solution scheme. For example, for the Gauss-Seidel iteration with partially given C++ code, we can write:

```
stationaryDirichletCI left("leftInputFile");
stationaryNeumannCI right("rightInputFile");

vector<double> f, d;
// Gauss-Seidel iteration
while(/*not converged*/)
{
left.solve();
left.get_load(f);
right.set_load(f);
right.solve();
right.get_disp(d);
left.set_disp(d):
}
```

The key point in one master process of the Gauss-Seidel iteration are the calls to two sub-systems of the coupled system as two separate software-components. If we have to enforce the stability by applying other algebraic solver for such a coupled problem, we have to define and implement other interfaces. In order to provide the optimal re-usability these should be arranged, using the extends mechanism, in the order of growing richness of functionality. Having the solve method and those acting on boundary conditions as given above, we can apply Gauss-Seidel-like iterations.

Other solvers would have other pre-assumption on the implemented features of the components that ought to be provided via interfaces. Let us consider for example the case of the Broyden-Fletcher-Goldfarb-Shanno (BFGS) solver. As CTL-interfaces for the simulations we can take for the Dirichlet boundary controlled sub-system:

```
interfacestationaryDirichletwResidualCI: extends
stationaryDirichletCI
{
voidset_state(const array<real8> /*state*/);
voidget_residual(array<real8> /*res*/);
};
```

and for the sub-system with Neumann boundary:

```
interfacestationaryNeumannwResidualCI: extends
stationaryNeumannCI
{
voidset_state(const array<real8> /*state*/);
voidget_residual(array<real8> /*res*/);
};
```

The BFGS algorithm needs access to all, also internal degrees of freedom, of the two simulation sub-systems. Therefore, in order to avoid the communication of the all the coefficients, it is reasonable to have two solver instances each with a local CTL-linkage to the corresponding simulation component.

```
// The Dirichlet part of the BFGS solver
// (running in parallel with  its Neumann part)
...
// create local link using thread linkage to Dirichletsimu part
ctl::linkdirichletSimu("path/to/dirichlet/simulation",
ctl::thread);
stationaryDirichletwResidualCI left(dirichletSimu,
"leftInputFile");
// link to the Neumann part of solver
ctl::linkneumannSolver;

// BFGS iteration
vector<double> r, f, d, u=inertial_guess;
while(/*not converged*/)
{
// compute left part of global residual
left.set_state(u);
 // receive displacement at Dirichlet boundary from Neumann part
neumannSolver>> d;
left.set_disp(d);
left.get_residual(r);
left.get_load(f);
 // send resulting force/pressure at interface boundary of Neumann
part
neumannSolver<< f;

// perform BFGS update for u using residual r
...
}

// The Neumann part of the BFGS solver
// (running in parallel  with its Dirichlet part)
...
// create local link using thread linkage to Neumann simu part
ctl::linkneumannSimu("path/to/neumann/simulation", ctl::thread);
stationaryNeumannwResidualCI right(NeumannSimu,
"rightInputFile");
// link to the Dirichlet part of solver
```

```
ctl::linkdirichletSolver;

// BFGS iteration
vector<double> r, f, d, u=inertial_guess;
while(/*not converged*/)
{
// compute right part of global residual
right.set_state(u);
right.get_disp(d);
 // send displacement at Dirichlet boundary
dirichletSolver<< d;
toDirichlet part
left.get_residual(r);
 // get force/pressure at interface boundary
dirichletSolver>> f;
fromDirichlet part

  r = r + f;

// perform BFGS update for u using residual r
...
}
```

These findings can be summarised within the following list:

- The Gauss-Seidel and Jacobi-iterations need essentially the simulation internal solver (via solve).
- The BFGS method needs essentially the residual function (via residual) which can (and should) be preconditioned.
- The inexact Newton-method needs the residual function together with the directional derivatives (via directionalDerivative) which are needed for the iterative solver for the linearised system.

In the case the method directionalDerivative cannot provide the corresponding values of cross-term derivatives, numerical differentiation can be used instead. However, this typically leads to a limited accuracy of the solution.

3.2 Numerical examples

3.2.1 Simple 1D thermoelastic coupling for validating stability criterion

In the context of code coupling, one of the main goal is to re-use existing specialized components where each one can be, for instance a finite element code. For thermomechanical problems, we split our problem into a mechanical and a thermal sub-problem. Both sub-problems are then solved by two independent sessions of the FEAP computer code (), and each part is linked by the CTL according to the corresponding partitioning scheme shown in Fig. 3 for isothermal split. In such a case, the mechanical sub-problem is solved at fixed temperature which is computed at the previous step as the solution of thermal sub-problem. Only temperature θ and gradient velocity $\dot{e}$ are exchanged between each solver at any synchronization point of a window.

This example is used to validate the stability criterion discussed in the previous section for thermomechanical coupling. For clarity of illustration, we consider a simple one-dimensional thermoelastic truss-bar structure and its FE model that consists of 100 elements.

We impose at the left end displacement time variation in terms of sinusoidal wave (see Fig. 4).

The imposed displacement generates a compressive wave in a bar. Through structural heating, this mechanical wave will generate a local increase in temperature. The reflection on the other end of the bar then generates a traction wave, with the corresponding decrease in temperature. We note that the variations in temperature remain essentially local, since the time scale of non-stationary heat transfer is much larger than the time scale of mechanical wave propagation.

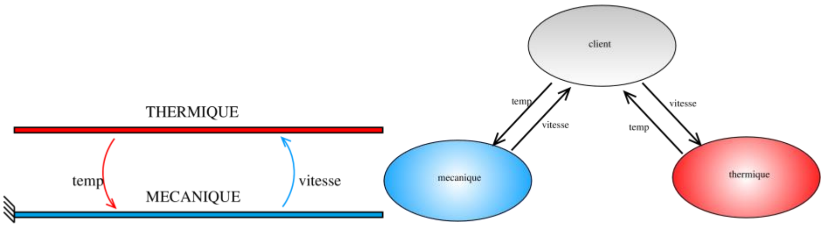

Fig. 3 Software implementation for thermomechanical coupling

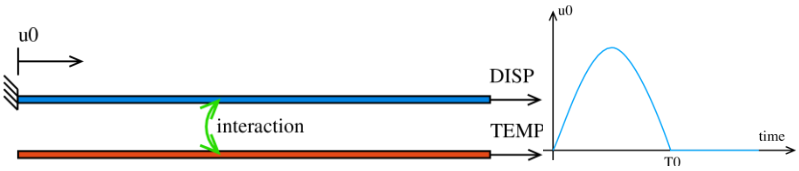

Fig. 4 One-dimensionalthermomechanical test

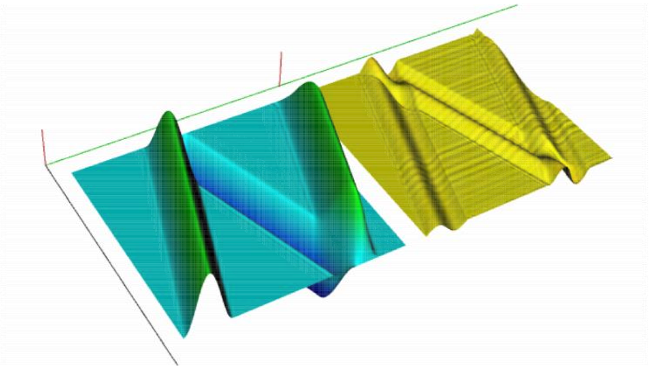

Fig. 5 Global solution of the 1D coupled problem in thermoelasticity computed by isothermal split: i) displacement field *u(x,t)*, ii) temperature field $\theta(x,t)$

Table 1 Materials properties for 1D thermoelastic bar

Young’s modulus	E	$200\,000 N\ mm^{-2}$
Area	A	$1 mm^{2}$
Specific heat capacity	c	$1{,}2 N\ mm^{-2} K^{-1}$
Reference temperature	θ_0	$1 K$
Conductivity	k	$0.15 N\ s^{-1} K^{-1}$
Expansion coefficient	α_0	$1.5\ 10^{-5} K^{-1}$
Thermal stress coupling term	$m_0 = E\,\alpha_0$	$3.00 N\ mm^{-2}\ K^{-1}$
Coupling coefficient	ε	$0.376\ 10^{-4}$

The main goal is to confirm the stability criterion and illustrate instabilities that can occur for strong thermomechanical coupling case. Namely, a weak coupling case corresponding to (see all values for material thermomechanical properties in Table 1), would lead to a very small of $\varepsilon = 0.376\ 10^{-4}$. The results in Figure 6 indeed confirm the expected convergence of isothermal split for small values of ε. However, for the strong coupling with an order-of-magnitude increase in , with the value of $\varepsilon = 0.376$ we run into instability problems with isothermal split for both displacement and temperature, as illustrated in Fig. 6. The same figures show that the adiabatic split preconditioning is capable to deliver the stable even for such strong coupling case. One can only note some spurious oscillations for temperature in Fig. 6 and obviously due to difficulty in representing a propagating sinus function with classical finite element shape functions. These oscillations increase with an increase of .

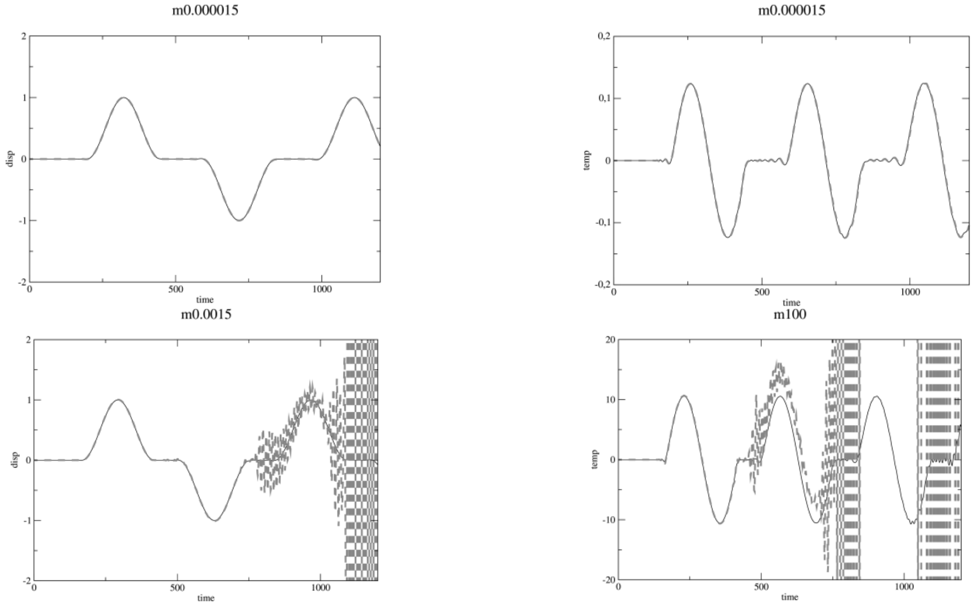

Fig. 6 Computed displacement and temperature for isothermal split (gray dashed line [__ __ __] which exhibits instabilities) and adiabatic split (black continuous line [______________] which always remains stable)

3.2.2 Nonlinear coupled thermomechanical analysis of masonry walls under fire-loading

In this particular example we study a nonlinear constrained evolution problem pertinent to fire resistance and damage of masonry structures under sustained fire. The thermomechanical coupling is brought about by temperature dependence of mechanical properties of the chosen plasticity model for brick material failure, as well as by the failure-induced topological changes to the holes inside the brick-cells where heat radiations effects are accounted for. The chosen material properties and brittle failure of masonry, generating a very small amount of heat, guarantee the stability of isothermal split. Hence, our goal here is not to illustrate the need for pre-conditioning to ensure stability, but to show the advantage of the proposed multi-scale strategy with different time steps for each sub-problem and the resulting gains in computational efficiency. Namely, the multi-scale in time scheme is very appropriate for this case where heat transfer scale is much larger than the mechanics scale of damage phenomena propagation.

The main challenge is to properly account for the thermomechanics coupling within inelastic analysis of masonry walls made from hollow clay bricks (see Fig. 7) exposed to fire. The main difficulty in modelling heat transfer in hollow bricks concerns the proper representation of the heat flow both through the cell-walls by heat conduction in the solid parts and by radiation within the holes inside the cells. The latter is the main source of non-linearity in heat transfer problem, since the inelastic dissipation to brittle failure is negligible. Moreover, the mechanical failure will also influence the thermal problem when the sufficient damage will require to remove the interior partitions between two adjacent holes inside the cell, causing a sudden change in the heat radiation domain. The temperature dependence of mechanical properties, along with the thermal stress produced with temperature induced constrained deformation of the wall, is the way the thermal problem influences the mechanical part.

The computational model, presented in our previous work (Ibrahimbegovic *et al*. 2005), combines folded plates or non-smooth shells for brick-cell partitions and heat radiation element placed inside each hole within a cell. The shell element we use (Ibrahimbegovic *et al*. 1994 a,b) combines the discrete Kirchhoff quadrilateral plate and a membrane element with drilling degrees of freedom (Ibrahimbegovic *et al*. 1990). This allows us to eliminate the lack of compatibility between the membrane and plate degrees of freedom, and corresponding membrane and plate bending deformation modes.

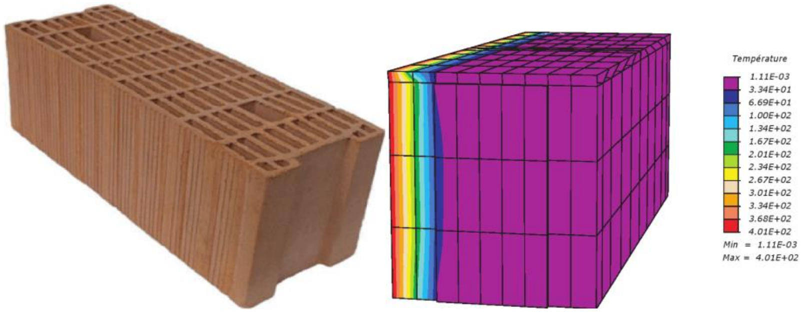

Fig. 7 Clay hollow block and damage contours under fire loading

The mechanical damage of the brick wall is modelled by the plasticity model of Saint-Venant developed in the strain space (Ibrahimbegovic 2009) and further recast in terms of stress resultants (Ibrahimbegovic *et al*. 2005), in the format of multi-surface plasticity (e.g., see Simo 1988). The key idea of this model is to limit the elastic domain by the maximum mechanical strain values, which is in accordance with the experimentally observed behavior of brittle materials. where failure is mainly driven by extensions (positive strains), leading to cracking in the directions perpendicular to the principal tension stress.

The heat transfer model is also developed within the flat shells context, accounting for both conductive and radiative transfers. The corresponding discrete formulation of heat conduction is based upon four degrees-of-freedom per node in-plane interpolations, where we assumed a linear temperature distribution through-the-thickness of the shell elements (Colliat *et al*. 2006). The key point here is that both of these mechanisms are leading to nonlinear behavior. Moreover, due to the particular features of radiative transfers, the tangent operator corresponding to this sub-problem is non-symmetric.

The computations are performed for the case where both mechanical and thermal loading is applied in the manner representative of fire-resistance testing. The mechanical loading is dead load on the brick wall. The chosen value of 1.3 *MPa* is directly introduced in the initial configuration in terms of compressive pre-stress and kept constant throughout the computation. The thermal loading is then applied, in terms of the uniformly distributed temperature field applied only at the brick facet exposed to fire. The time evolution of this temperature field is imposed according to specified fire duration:

$$\theta(t) = \theta_0 + 345 \log(8t+1) \; ; \; t \in \; 2\,[0, 200] \tag{32}$$

where θ_0 is the initial temperature and t the time in minutes. The thermal and mechanical problems are solved by multi-scale in time independent computations of each sub-problem carried out by separate runs of the FEM-code FEAP, linked through the CTL software, as described in the previous section,

The principal benefit of the chosen multi-scale in time strategy pertains to coupled problems of this kind where the time scale of the different coupled sub-problems are quite different. Here, the time scale for thermal problem is much bigger that the one used the mechanics process pertaining to the evolution of internal variables. Hence, it pays off to use different time steps for each sub-problem in order to gain efficiency. This is confirmed by the results illustrated in Fig. 8, showing the CPU total solution time as measured by the machine clock at a PIV 3.4GHz processor. In different computations we carried out, the time step of mechanical problem is kept fixed with $\Delta t_M = 24s$, and the time step of thermal problem is $\Delta t_T = N \, \Delta t_M$ leading to a window size of a single Δt_T and $N \, \Delta t_M$ for the exchange of data with the Gauss-Seidel algorithm. By assuming that both mechanics and thermal nonlinear evolution problems have the same size and same convergence rate, the theoretical speed-up one can obtain with such a choice of different time steps for each sub-problem in the limit equals to

$$\lim_{N\to\infty} \frac{CPUtime(N)}{CPUtime(N=1)} \approx \lim_{N\to\infty} \frac{N+1}{2N} = \frac{1}{2} \tag{33}$$

Even if the hypothesis on equal convergence rate for each sub-problem is not necessary satisfied in each time step, our results denoted as black squares in Fig. 8 seem to confirm this asymptotic value denoted as dashed line of computational speed-up granted by multi-scale in time computational strategy.

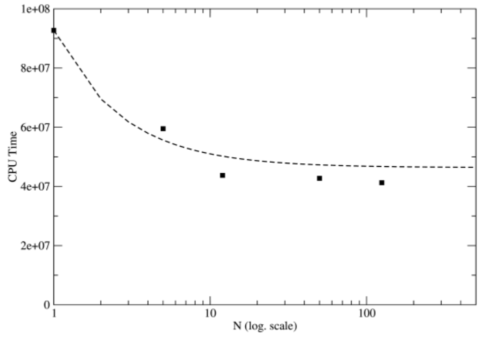

Fig. 8 CPU computation time for different values of ratio $N = \Delta t_T / \Delta t_M$

We note that the main reason for choosing a fixed time step for mechanics was in a more reliable comparison of the CPU time. We note in passing that such a choice is not the same as an optimal value proposed in (Ibrahimbegovic *et al*. 2005), which is obtained by adaptive procedure that ensures the same accuracy of computations of internal variables evolution for the complete duration of fire. Despite a non-optimal choice of time step for mechanics, we show next that our numerical results compare quite well with experimental results.

First, this applies to the results presented for temperature field in Figure, where we show the time-evolution of the temperature in three different cells. The chosen locations (see Fig. 9) concern the two surfaces on both sides of each cell and confirm we are able to capture the temperature evolutions even far from the exposed facet of the wall. This result is further confirmed in Fig. 9, which shows a temperature profile in through wall thickness at 48 minutes after the beginning of fire. One of the main ingredients allows to obtain such a good result is the introduction of radiative exchanges in the heat transfer model.

In Fig. 10 we show the evolution of the total vertical reaction at selected nodes. Each curve corresponds to a line of nodes parallel to the face of the wall exposed to fire, with positive values indicating the compression. We note the variations of this kind of reaction forces with respect to prestressed initial value due to mechanical dead load. Fig. 10 shows the comparison on the horizontal computed displacement with an experimental value measured for the real wall built with ten rows of bricks. This kind of displacement is produced by bending which is due to the temperature gradient through the wall. We can see that the stiffness provided by the analysis is quite correct, even though the displacements are slightly over-estimated due to a poor representation of the true boundary conditions.

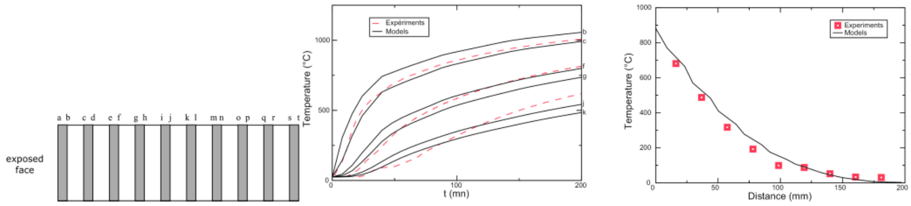

Fig. 9 Brick – (a) locations in brick; (b) temperature evolutions; (c) temperature profile after 48 min;

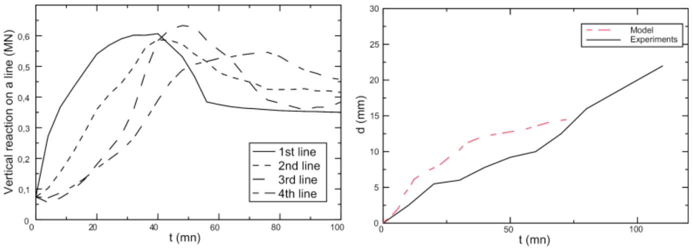

Fig. 10 Brick – total vertical reactions for the facet exposed to fire and horizontal displacement due to bending of the wall

4. Conclusions

The main thrust of this work is directed towards coupled thermomechanics systems, and the solution of corresponding (nonlinear) evolution problems. We have shown that providing the complete answer to such a question pertains to aspects of theoretical formulations, operator split solution procedure and efficient software development by using CTL-Component Template Library that allows to re-integrate the existing codes.

The model problem addressed herein concerns coupled thermomechanical system, which was first studied for the elastic case. We have provided the explicit stability criterion for the operator split solution procedure of coupled problem, where each sub-system (mechanical or thermal) is integrated by its dedicated stable scheme. The proposed stability criterion applies to the case where time-scale of evolution for each sub-system is different from others, and is integrated with its own time step. The stability of the coupled computation is thus enforced on a given window. We have also demonstrated the gain one can obtain when using the different time steps for each sub-system and the corresponding speed-up one can achieve with such a strategy.

The detailed theoretical development of stability criterion is presented for thermoelasticity. However, given that each sub-problem (between mechanics and heat transfer) is handled by an appropriate iterative scheme that is guaranteed to converges, the stability result of coupled problem computations remains valid. In particular, this has been demonstrated in our examples for thermoplasticity problem used to simulate the damage propagation of masonry structure exposed to fire. Other inelastic models are also suitable for the same approach.

We have further exploited this stability criterion to automatically satisfy stability of coupled problem computation. In the present context this amounts to the adiabatic split of the constraint. While this kind of split leads to a stronger interaction of sub-systems within the coupled problem, the benefits of unconditional stability are worthy of it. The corresponding software implementation is shown to increase quite modestly in terms of complexity.

With our previous works on interaction problems (Kassiotis 2011a, b, Ibrahimbegovic *et al.* 2014), the present coupled thermomechanical problem allows to complete the different possibility of interface interaction, either on the sub-domain boundary or within the volume. The future works of interest should expand upon the coupling with non-physical sub-system, such as in stochastic finite element analysis (Krosche and Matthies 2008, Ibrahimbegovic and Matthies 2012), and optimization (Niekamp and Krosche 2003).

Acknowledgements

This work was supported by DFG and Mercator-Fellow program at InstitutfűrStatik, TU Braunschweig. This support is gratefully acknowledged.

References

Armero, F. and Simo, J.C. (1992), “A new unconditionally stable fractional step method for non-linear coupled thermomechanical problems”, *Int. J. Numer.Meth. Eng.*, **35**(4), 737-766.

Armero, F. and Simo, J.C. (1993), “A priori stability estimates and unconditionally stable product formula algoritms for nonlinear coupled thermoplasticity”, *Int. J. Plasticity*, **9**(6), 749-782.

Arnold, M. and Gunther, M. (2001), “Preconditioned dynamic iteration for coupled differential-algebraic systems”, *BIT Numer. Math.*, **41**(1), 1-25.

Bathe, K.J. (1996), *Finite element procedures*, Prentice Hall, Englewood Cliffs.

Bindel, D. (2011), Matfeap, http://www.cs.cornell.edu/ bindel/sw/matfeap/.

Birken, P., Quint, K., Hartmann, S. and Meister, A. (2010), “Choosing norms in adaptive fsi calculations”, *PAMM*, **10**(1), 555-556.

Brenan, K.E., Cambell, S.L. and Petzold, L.R. (1996), *Numerical solution of initial-value problem in differential-algebraic equations*, SIAM - Society for Industrial and Applied Mathematics, Philadelphia.

Chorin, A.J., Hughes, T.J.R., McCracken, M.F. and Marsden, J.E. (1978), “Product formulas and numerical algorithms”, *Commun. Pur. Appl. Math.*, **31**(2), 205-296.

Ciarlet, P.G., Miara, B. and Thomas, J.M. (1989), *Introduction to numerical lnear algebra and optimisation*, Cambridge University Press, Cambridge.

Colliat, J.B., Ibrahimbegovi, A. and Davenne, L. (2006), “Heat conduction and radiative heat exchange in cellular structures using _at shell elements”, *Commun. Numer. Meth. En.*, **22**, 167-180.

Dennis, J.E. and Schnabel, R.B. (1996), *Numerical methods for unconstrained optimization and nonlinear equations*, SIAM - Society for Industrial and Applied Mathematics, Philadelphia.

Dubois-Pelerin, Y. and Pegon, P. (1998), “Object-oriented programming in nonlinear finite element analysis”, *Comput. Struct.*, **67**(4), 225-241.

Dubois-Pelerin, Y., Zimmerman, T. and Bomme, P. (1992), “Object-oriented finite element programming: II prototype programme in smalltalk”, *Comput Method. Appl. M.*, **98**(3), 361-397.

Eyheramendy, D. and Gudin-Dardun, F. (2008), Object-oriented finite elements: from smalltalk to java, *Trends in Eng. Comp. Technology (*Eds., Papadrakakis, M. and Topping, B.H.V.).

Farhat, C., Park, K.C. and Dubois-Pelerin, Y. (1991), “An unconditionally stable staggered algorithm for transient finite element analysis of coupled thermoelastic problem”, *Int. J. Numer. Meth. Eng.*, **85**(3), 349-365.

Felippa, C.A. and Park, K.C. (2004), Synthesis tools for structural dynamics and partitioned analysis of coupled systems. In *Multi-physics and multi-scale computers models in non-linear analysis and optimal design*, pages 50–111, Bled, Slovenia, IOS Press (Eds., Ibrahimbegovic, A. and Brank, B.).

Feyel, F. and Chaboche, J.L. (2000), “Fe2 multiscale approach for modelling the elastoviscoplastic behaviour of long fibre sic/ti composite materials”, *Comput. Method. Appl. M.*, **183**, 309-330.

Gear, G.W. (1971), *Numerical initial value problems in ordinary differential equations*, Prentice-Hall, Englewood Cliffs N.J.

Golub, G.H. and Van Loan, C. (1983), *Matrix computations*, Johns Hopkins University Press.

Hautefeuille, M., Ibrahimbegovic, A., Matthies, H. and Villon, P. (2012), “Multiscale approach to modeling inelastic behavior with softening”, *Comput. Struct.*, **94-95**, 83-95.

Hughes, T.J.R.(2005), “Multiscale phenomena: GreenÃÂÂ™s functions, the dirichlet-to-neumann formulation, subgrid scale models, bubbles and the origins of stabilized methods”, *Comput. Method. Appl. M.*, **127**, 387-401.

Ibrahimbegovic, A. (1994), “Stress resultant geometrically nonlinear shell theory with drilling rotations. Part I: a consistent formulation”, *Appl. Mech. Eng.*, **114**, 311-332.

Ibrahimbegovic, A. (2006), *Nonlinear computational solid mechanics: theoretical formulation and finite element computations*, Hermes-Science Publication, Paris, (in French).

Ibrahimbegovic, A. (2009), *Nonlinear solid mechanics: theoretical formulation and finite element solution methods*, Springer, Berlin.

Ibrahimbegovic, A. and Brank, B. (2005), *Multi-physics and multi-scale computer models in nonlinear analysis and optimal design of engineering structures under extreme conditions*, IOS Press, Amsterdam.

Ibrahimbegovic, A. and Chorfi, L. (2002), “Covariant principal axis formulation of associated coupled thermoplasticity at finite strains and its numerical implementation”, *Int. J. Solids Struct.*, **39**(2), 499-528.

Ibrahimbegovic, A., Colliat, J.B. and Davenne, L. (2005), “Thermomechanical coupling in folded plates and non-smooth shells”, *Comput. Method. Appl. M.*, **194**(21-24), 2686-2707.

Ibrahimbegovic, A. and Frey, F. (1994), “Stress resultant geometrically nonlinear shell theory with drilling rotations. Part III: Linearized kinematics”, *Int. J. Numer. Meth. Eng.*, **37**(21), 3659-3683.

Ibrahimbegovic, A. and Markovic, D. (2003), “Strong coupling methods in multi-phase and multi-scale modeling of inelastic behavior of heterogeneous structures”, *Comput. Method. Appl. M.*, **192**(28-30), 3089-3107.

Ibrahimbegovic, A. and Matthies, H.G. (2012), “Probabilistic multiscale analysis of inelastic localized failure in solid mechanics”, *Comput. Assist. Method. Eng. Sci.*, **19**, 277-304.

Ibrahimbegovic, A., Niekamp, R., Kassiotis, C., Markovic, D. and Matthies, H.G. (2014), “Code-coupling strategy for efficient development of computer software in multiscale and multiphysics nonlinear evolution problems in computational mechanics”, *Adv. Eng. Softw.*, in press.

Ibrahimbegovic, A., Taylor, R.L. and Wilson, E.L. (1990), “A robust quadrilateral menbrane finite-element with drilling degrees of freedom”, *Int. J. Numer. Meth. Eng*., **30**(3), 445-457.

Ibrahimbegovic, A., Taylor, R.L. and Lim, H. (2003), “Non-linear dynamics of flexible multibody systems”, *Comput. Struct.*, **81**(12), 1113-1132.

Kassiotis, C., Ibrahimbegovic, A., Niekamp, R. and Matthies, H. (2001a), “Partitioned solution to nonlinear fluid-structure interaction problems.part i: implicit coupling algorithms and stability proof”, *Comput. Mech.*, **47**, 305-323.

Kassiotis, C., Ibrahimbegovic, A., Niekamp, R. and Matthies, H. (2011b), “Partitioned solution to nonlinear fluid-structure interaction problems.part ii: Ctl based software implementation with nested parallelization”, *Comput. Mech.*, **47**, 335–357.

Krosche, M. and Matthies, H.G. (2008), “Component-based software realisations of Montecarlo and stochastic galerkin methods”, *PAMM*, **8**(1), 10765-10766.

Ladeveze, P. (2005), *Multiscale computational damage modelling of laminate composites*, *CISM course*, (Ed. Sadowski, T.).

Langtangen, H.P. (2008), *Python scripting for computational science*, Springer, Berlin.

Markovic, D., Niekamp, R., Ibrahimbegovic, A., Matthies, H.G. and Taylor, R.L. (2005), “Multi-scale modeling of heterogeneous structures with inelastic constitutive behavior.part i: mathematical and physical aspects”, *Int. J. Eng. Comput.*, **22**, 664-683.

Matthies, H.G., Niekamp, R. and Steindorf, J. (2006), “Algorithms for strong coupling procedures”, *Comput. Method. Appl. M.*, **195**(17-18), 2028-2049.

Matthies, H.G. and Steindorf, J. (2003), “Partitioned strong coupling algorithms for fluid-structure interaction”, *Comput. Struct.*, **81**(8-11), 805-812.

Niekamp, R. (2001), Ctl-homepage, http://www.wire.tu-bs.de/ forschung/ projekte/ ctl/e-ctl.html.

Niekamp, R., Krosche, M. and Matthies, H.G. (2003), *Platon: A problem solving environment for computational steering of evolutionary optimisation on the grid*, (Eds., Bugeda, G. *et al*.), *EUROGEN*.

Niekamp, R., Markovic, D., Ibrahimbegovic, A., Matthies, H.G. and Taylor, R.L. (2009), “Multi-scale modeling of heterogeneous structures with inelastic constitutive behavior.part ii: software coupling and implementation aspects”, *Int. J. Eng. Comput.*, **26**, 6-28.

Oden, J.T., Belytschko, T., Babuska, I. and Hughes, T.J.R. (2003), “Research directions in computational mechanics", *Comput. Method. Appl. M.*, **192**(7-8), 913-922.

Oden, J.T., Prudhomme, S. and Bauman, P. (2005), "On the extension of goal-oriented error estimation and hierarchical modeling to discrete lattice models", *Comput. Method. Appl. M.*, **194**(34-35), 3668-3688.

Owen, D.R.J. and Hinton, E. (1980), .*Finite element method in plasticity*, Pineridge Press, Swansea.

Simo, J.C., Kennedy, J. and Govindjee, S. (1988), "Non-smooth multisurface plasticity and viscoplasticity. Loading/unloading conditions and numerical algorithms", *Int. J. Numer. Meth. Eng.*, **26**(10), 2161-2185.

Zienkiewicz, O.C. and Taylor, R.L. (2000), *The finite element method*, Butterworth-Heinemann, Oxford, 5th Ed.