11th World Congress on Computational Mechanics (WCCM XI)
5th European Conference on Computational Mechanics (ECCM V)
6th European Conference on Computational Fluid Dynamics (ECFD VI)
July 20–25, 2014, Barcelona, Spain

A DATA EFFICIENT, CQM-BASED BEM APPROACH FOR ELASTODYNAMICS

Bernhard Kager[1], Martin Schanz[2]

[1] Graz University of Technology, Technikerstraße 4, 8010 Graz, Austria, bernhard.kager@tugraz.at, www.mech.tugraz.at
[2] Graz University of Technology, Technikerstraße 4, 8010 Graz, Austria, m.schanz@tugraz.at, www.mech.tugraz.at

Key words: *Boundary Element Method, Fast Algorithms, Elastodynamics, Convolution Quadrature, Data Efficiency.*

The numerical treatment of wave propagation through linear elastic continua with the Boundary Element Method (BEM) inherently involves the proper approximation of temporal convolution integrals. This talk introduces and discusses a new data efficient formulation that is based on Lubich's convolution quadrature method [1].

Convolution Integral and Convolution weights Given a time interval $(0, T) \in \mathbb{R}^+$ and its subdivision into M equidistant timesteps Δt, such that $t_n = n\Delta t$, $n = 0..M$. Consider a domain $\Omega \in \mathbb{R}^3$, two points $\boldsymbol{x}, \boldsymbol{y} \in \partial\Omega$ with $\boldsymbol{r} = \boldsymbol{y} - \boldsymbol{x}$ and a vector-valued field $\boldsymbol{\phi}(\boldsymbol{x}, t)$. With the displacement fundamental solution $\boldsymbol{U}(\boldsymbol{r}, t)$, the occuring convolution integral is consequently defined and approximated by

$$\int_0^{t_n} \boldsymbol{U}(\boldsymbol{r}, t_n - \tau)\, \boldsymbol{\phi}(\boldsymbol{y}, \tau)\, \mathrm{d}\,\tau \approx \sum_{m=1}^{n} \boldsymbol{\omega}^{n-m}(\boldsymbol{r})\, \boldsymbol{\phi}^m(\boldsymbol{y}), \quad \boldsymbol{\omega}^n(\boldsymbol{r}) = \frac{\partial^n}{n!\partial\xi^n}\left[\hat{\boldsymbol{U}}\left(\boldsymbol{r}, \frac{\gamma(\xi)}{\Delta t}\right)\right]_{\xi=0}.$$

A crucial prerequisite for this approach is the existence of the fundamental solution $\hat{\boldsymbol{U}}(\boldsymbol{r}, s)$ in Laplace domain and the selection of an appropriate multistep method with characteristic polynomial $\gamma(\xi)$. Commonly, Cauchy's Integral Formula is used to compute the n-th partial derivative with respect to ξ. Contrary to that, as is proposed by [2] for acoustics, we present a direct, recursive evaluation of the weights for the elastodnamic case. $\hat{\boldsymbol{U}}(\boldsymbol{r}, s)$ is a linear combination of expressions

$$P_\alpha^n(r, s) := s^{-n} \exp\left(-\frac{s}{r c_\alpha}\right), \quad n = 0, 1, 2 \text{ and } \alpha = 1, 2$$

with c_α being the wave velocities. Based on the BDF-2 scheme, Hackbusch, Kress and Sauter [2] provide exact expressions for the n-th partial derivative of $P_\alpha^0(r, \gamma(\xi)/\Delta t)$ involving Hermite polynomials. Monegato [3] extends this to higher BDF schemes and provides recurrence relations. In this presentation, we derive similar expressions for the computation of the n-th partial derivatives of the two remaining expressions $P^1_\alpha\left(r, {}^{\gamma(\xi)}/_{\Delta t}\right)$ and $P^2_\alpha\left(r, {}^{\gamma(\xi)}/_{\Delta t}\right)$ for BDF-2. This finally results in a direct evaluation of the convolution weights for the single and regularized double layer potential.

Weight Interpolation Scheme Since the spatial discretization is inevitably bounded with $r \leq r_{\max}$, a Hermite interpolation scheme is employed for the weight functions in $[0, r_{\max}]$ to speed up the kernel evaluation.

Data Efficiency The weight functions that exhibit local support behaviour in r act as kernel functions for the spatial integration. To utilize the local support information, a Principal Component Analysis is performed to compute subdomains of Γ. Subsequently, utilizing a numerical support detection, zero matrix blocks are determined prior to any matrix evaluation.

Results A rod of dimensions $3\,\mathrm{m} \times 1\,\mathrm{m} \times 1\,\mathrm{m}$ is investigated (Figure 1). The rod is clamped on one side and on the opposing side a heaviside traction load is applied. Parameters are set to $\beta = \frac{c_1 \Delta_t}{r_e} = 1$, $c_1 = 1\frac{\mathrm{m}}{\mathrm{s}}$, $c_2 = \sqrt{0.5}\frac{\mathrm{m}}{\mathrm{s}}$ and the interval detection accuracy $\varepsilon_{\text{supp}} = 10^{-6}$. The results are shown in Table 1.

$\mathrm{m}_{to.}$ ... total memory [GB]

$\mathrm{m}_{th.}$ ... theoretical memory [GB]

lvl	# elem.	n_t	$\mathrm{mem}_{to.}$	$\mathrm{mem}_{th.}$	ratio
0	112	38	-	-	0.81
1	448	56	0.18	0.23	0.78
2	1792	87	4.21	5.91	0.71
3	7168	143	89.68	157.16	0.57

Table 1: Memory Consumption results

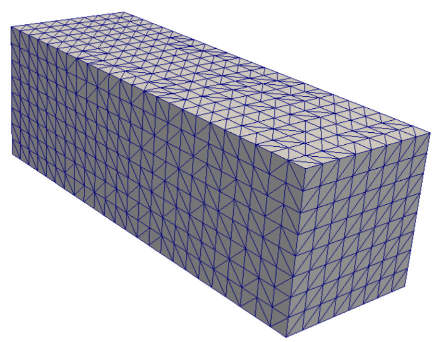

Figure 1: Surface mesh (1792 elements)

REFERENCES

[1] C. Lubich. Convolution Quadrature and Discretized Operational Calculus. I. *Numerische Mathematik*, Vol. **52**, 129–145, 1988

[2] W. Hackbusch, W. Kress, S. Sauter, Sparse convolution quadrature for time domain boundary integral formulations of the wave equation by cutoff and panel-clustering. In *Boundary Element Analysis*, M. Schanz, O. Steinbach (eds). Lecture Notes in Applied and Computational Mechanics, Vol. **29**. Springer: Berlin, Heidelberg, 113–134, 2007.

[3] G. Monegato, L. Scuderi and M. Stanić. Lubich convolution quadratures and their application to problems described by space-time BIEs. *Numerical Algorithms*, Vol. **56**, 1017405–436, 2010.