STEADY STATE FLOW ALGORITHM FOR MODELING THE IMPACT OF TRUCKS ON ROAD

SEIGFRIED MAÏLINO
CETE de Lyon/ ERA12, Bron
UMR 7649–LMS

HABIBOU MAÏTOURNAM
UMR 7649–LMS
École Polytechnique, France

VÉRONIQUE CEREZO
CETE de Lyon/ ERA12
Bron, France

Abstract
In this paper, we present a powerful tool for studying the impact of trucks on roads. Whereas there is a real need to know the damage made by repeated rolling, FEM methods are not really fitted to compute moving loads and are time consuming. Steady state algorithm is used to integrate movement in FEM computations. This method allows faster computations, more coarse mesh and a more efficient study of time dependent phenomena. This paper presents the method and illustrates it with an example of elastoplastic modeling.

Keywords: Finite element methods, damage, fatigue, pavement, numerical methods.

Résumé

Mots-clés: Éléments finis, endommagement, fatigue, chaussée, méthodes numériques.
1. Introduction

The determination of mechanical fields under a moving load and the evaluation of the residual strains and stresses obtained after repeated passes of loadings are the first step toward the understanding and the prediction of fatigue and damage phenomena occurring under truck traffic. However Finite Element Method is not well adapted to compute moving loads. Methods based on incremental translations of the loading are time consuming and cumbersome. Zarka (1980) and Zarka et al. (1988) proposed a very fast approximate analysis going straight through the stabilized state, but this method did not evaluate the evolution of the plastic strain pass by pass.

\[ n + 1 \quad n \quad n - 1 \]

(a) Classical method

(b) Steady-state flow

Figure 1 – Comparisons between methods

We present a systematic and reliable algorithm for computational analysis of the impact of trucks on roads. It avoids lengthy repeated calculations due to incremental translations of the load. With the basic assumption of a constant velocity, we can consider the reference frame associated to the moving load. In this reference frame, velocities and all physical quantities are time independent, and time derivatives are replaced by spatial derivatives. Geometry, and consequently mesh, don’t change. Mesh will be refined only under the load, and can remain coarse in other areas, thus reducing the computational cost. From a physical point of view, this method allows the integration of the evolution law of quantities in motion (figure 1(b)), instead of considering a succession of singular static states (figure 1(a)). This point is particularly relevant for time-dependent inelasticity.

The steady state method is based on the works of Nguyen and Rahimian (1981) and Dang Van et al. (1985) which described the theoretical framework and first applications of the method. This method had been used to various problem involving moving loads, such as the impact of rolling on rail heads (Maïtournam, 1989, Dang Van and Maïtournam, 1993), beading (Ouakka, 1993), interaction between rock and cutting tool (Geoffroy, 1996), automotive brake disk (Nguyen-Tajan et al., 2002), tunnelling (Corbetta, 1990, Guo, 1995, Maïolino, 2006).

2. Formulation of steady state problems

2.1 General framework

Consider that the load is moving with a velocity \( v : \vec{v} = -ve \vec{X} \). The reference frame is moving with the load, so that in this frame, velocities and all the physical quantities are time-independent. The \( x \) are oriented opposite to the load direction (i.e. \( \vec{e}_x = e \vec{X} \)), so that the initial state of the structure is defined by the state conditions when \( x \) tends toward \(-\infty\). The time derivative of a
tensorial quantity $A$ (equation (1)) can be replaced by a spatial derivative - we consider strains are infinitesimal(equation (2)).

$$\dot{A} = \frac{\partial A}{\partial t} + \vec{v} \cdot \text{grad} A$$  \hspace{1cm} (1)

$$\dot{A} = v \frac{\partial A}{\partial x}$$  \hspace{1cm} (2)

We consider an elastoplastic material, with the following associated flow rule and hardening law ($\eta$ being the hardening parameter) :

$$\begin{cases}
\varepsilon_p = \lambda \frac{\partial f}{\partial \sigma} \\
\eta = -\lambda h(\sigma, \eta)
\end{cases} \hspace{1cm} \text{with $\lambda f = 0, \lambda \dot{f} = 0, \lambda \geq 0$ and $f \leq 0$}$$  \hspace{1cm} (3)

With the steady-state assumption, those laws are expressed as spatial derivatives :

$$\begin{cases}
\frac{\partial \varepsilon_p}{\partial x} = \Lambda \frac{\partial f}{\partial \sigma} \\
\frac{\partial \eta}{\partial x} = -\Lambda h(\sigma, \eta)
\end{cases} \hspace{1cm} \text{with $\Lambda \geq 0$ if $f(\sigma_{n+1}, \eta_{n+1}) = 0$}$$

$$\Lambda = 0 \hspace{1cm} \text{otherwise}$$  \hspace{1cm} (4)

With $\Lambda$ being calculated from the consistency condition. The plastic strain and hardening parameters are obtained by integrating the flow rule and the hardening law along a particle trajectory. This problem is non linear, and will be solved numerically.

2.2 Algorithm for Finite Element Methods

spatial discretization

Discretization points (Gauss points) are aligned in the motion direction and numbered in the direction opposite to the motion (figure 2), so that the evolution of plastic strain will be calculated from $n$ to $n+1$. We will use an implicit scheme so that flow and hardening during motion from point $n$ to $n+1$ are calculated at point $n+1$. The discretized flow rule and hardening law can be written as follow :

$$\begin{cases}
\varepsilon_{p,n+1} - \varepsilon_{p,n} = \Lambda \frac{\partial f}{\partial \sigma}(\sigma_{n+1}, \eta_{n+1}) \\
\eta_{n+1} - \eta_n = -\Lambda h(\sigma_{n+1}, \eta_{n+1})
\end{cases} \hspace{1cm} \text{with $\Lambda \geq 0$ if $f(\sigma_{n+1}, \eta_{n+1}) = 0$}$$

$$\Lambda = 0 \hspace{1cm} \text{if } f(\sigma_{n+1}, \eta_{n+1}) < 0$$  \hspace{1cm} (5)
**general resolution algorithm**

Load application with steady state algorithm is slightly different from usual finite elements. When using the latter method, load is applied incrementally, whereas when using the steady state algorithm the whole load is applied since the first step. However the resolution is incremental as we can see on the flow chart of the figure 3. At first step, the plastic strain is equal to the initial plastic strain: it can be a null tensor or a residual strain \( \mathbf{\varepsilon}_p^0 \) to take into account the state resulting from previous passes of loading (for some materials, like rails, the initial plastic state results from the manufacturing process). At the beginning of each iteration \( i \), the elastic problem with an initial plastic strain \( \mathbf{\varepsilon}_p^i \) is solved. If the resulting stress field \( \mathbf{\sigma}^i \) is statically admissible, the solution is good, otherwise, a new plastic field \( \mathbf{\varepsilon}_p^{i+1} \) is calculated by integrating flow rule and hardening law along the different particle trajectories. The counter is incremented and one more loop is computed.

**plasting strain computation at integration points**

Integration of the flow rule and hardening law is performed sequentially along a particle line. The first point, point 1 is not affected by the load and its state is the initial condition:

\[
\begin{align*}
\mathbf{\varepsilon}_p^{i+1}_1 &= \mathbf{\varepsilon}_p^0 \\
\eta^{i+1}_1 &= \eta^0
\end{align*}
\]

(6)

Assuming the state at point \( n \) has been computed, then the trial state \( \mathbf{\sigma}^* \) at point \( n+1 \) is calculated, assuming there is no variation of plastic strain between the two points:

\[
\mathbf{\sigma}^* = \mathbf{\sigma}^0 + \mathbf{L} \left( \mathbf{\varepsilon}_n^{i+1} - \mathbf{\varepsilon}_n^{p,i+1} \right)
\]

(7)

If \( f(\mathbf{\sigma}^*, \eta^{i+1}_n) < 0 \) there is no plastification between points \( n \) and \( n+1 \), so that: \( \mathbf{\varepsilon}^{p,i+1}_n = \mathbf{\varepsilon}^{p,i+1}_{n+1} \), \( \eta^{i+1}_{n+1} = \eta^{i+1}_n \), and the actual stress is equal to the trial stress: \( \mathbf{\sigma}^{i+1}_{n+1} = \mathbf{\sigma}^* \).
If \( f \left( \sigma^i, \eta^{i+1}_{n} \right) \geq 0 \) then plastification occurs:

1. \( \varepsilon^{p_{i+1}}_{n+1} = \varepsilon^{p_{i+1}}_{n} + \Lambda \frac{\partial f}{\partial \sigma} \left( \sigma^{i+1}_{n+1}, \eta^{i+1}_{n+1} \right) \) 
2. \( \eta^{i+1}_{n+1} = \eta^{i+1}_{n} - \Lambda h \left( \sigma^{i+1}_{n+1}, \eta^{i+1}_{n+1} \right) \)
3. \( f \left( \sigma^{i+1}_{n+1}, \eta^{i+1}_{n+1} \right) = 0 \)
4. \( \sigma^{i+1}_{n+1} = \sigma^{*} + L \left( \varepsilon^{p_{i+1}}_{n+1} - \varepsilon^{p_{i+1}}_{n} \right) \)

The resolution of this system of equations is the same as in classical computational plasticity, as those equations, particularly equation (11), come down to closest point projection problem (figure 4), similar to classical computational elastoplasticity. For circular yield function, the radial return (Wilkins, 1964) reduces the computations to literal expressions (Krieg and Key, 1976). For non-circular criteria, one has to compute return mapping algorithms (Simo and Hughes, 1998), to use the spectral decomposition techniques (Borja et al., 2003, Foster et al., 2005), or the numerical abacuses method (Maïolino et al., 2007).

3. **Numerical example**

We are currently developing specific procedures to implement the steady state method in the finite element software CAST3M, which allows direct access to integration points, and offers tools to introduce new models. On this numerical example, we consider a simplified load, rolling on an elastoplastic material. We use a Von Mises plastic flow rule, with isotropic hardening.

Twenty passes were produced, and we have produced the plastic vertical strain during the first pass.
Figure 6 – Residual plastic strain as a function of depth

figure 5(a) and the last pass figure 5(b). To compare the evolution of the residual state, the plastic vertical strain as a function of depth after loading ($x \to +\infty$) is plotted in the figure 6, for some passes.

4. Prospects

Work is being done to implement more functionalities. The steady state algorithm is particularly well fitted for viscoplastic problem, as the integration scheme between two points refers directly to the speed:

$$\varepsilon_{vp}^{n+1} - \varepsilon_{vp}^n = \frac{\Delta x}{\nu} \frac{\partial \Omega}{\partial \sigma}$$

(12)

With $\Omega$ the viscoplastic potential ($\dot{\varepsilon}_{vp} = \frac{\partial \Omega}{\partial \sigma}$), and $\Delta x$ the distance between the points $n$ and $n + 1$.

We intend to focus on the discrepancies induced by the use of simplified loads: more accurate truck models will be introduced and compared for one pass versus repeated loading, with simplified loadings.

The steady state flow algorithm is very versatile, and can be used with almost any rheologic model. One can notice that there was no restriction on the flow rule, and on the internal variable, in the description of the algorithm: transition from the flow rule to the steady state form is straightforward.

5. References


