New High Performance Stiff and Deformable Digital Roads for *FTire*

Gerald Hofmann and Michael Gipser @ cosin scientific software
FTire: Higher frequency, short wavelength tire model

- Structural dynamics based, full 3D nonlinear in-plane and out-of-plane tire model for simulation of belt dynamics, local pressure distribution in the contact patch, rolling resistance, side-wall contact, large camber angles and misuse scenarios.

- Suitable for a frequency range up to 200Hz, excited by short surface wavelength, mass imbalance, non-uniformity or irregular tread patterns.

- Very fast and flexible. Orders of magnitude faster than explicit FE models.

- Simulation of imbalances by inhomogeneous mass distribution and local wear.

- Belt temperature distribution model

- Available in the Altair HyperWorks Partner Alliance
Vehicle / Tire / Road Interfacing

- cosin tire interface (CTI)
- cosin road interface (CRI)
- FTire interface
CRI: Gateway to Road Models and Interfaces (2)

**cosin models**
- **cosin/2Droad**: analytical 2D and 3D rigid road surface description: single obstacles, rotating drum with cleats, hydro-pulse, stochastically uneven roads..
- **cosin/track**: analytical road center line description: straight line, clothoid, circle, mean banking angle, lane number, lane width, animation data..
- **cosin/3Droad**: cosin Regular Grid Roads: efficient storage, evaluation and analysis of high resolution road surfaces, including OpenCL-accelerated implementation
- **cosin/soil**: cosin Soft Soil Model: Cosin’s internal soft soil model. Software structure and interfaces to tire model completed, *physical model under development*
- **cosin/prm**: cosin Particle Road Model: Cosin’s granular material road surface model, making use of massively parallel GPU computations. *Under development*

**User plugins**

**3rd-Party plugins**
CRI: Gateway to Road Models and Interfaces (3)

- cosin models
  - cosin/2Droad
  - cosin/track
  - cosin/3Droad
  - cosin/soil
  - cosin/ prm

- cosin evaluation

- User plugins

- 3rd-Party plugins

implemented in open source software librgr (provided by cosin);
partially making use of new parallelization standard OpenCL
Road Surface / Contact pressure / Spindle Loads

[Image of a graph showing ground pressure in MPa with color coding for different pressure ranges.]
RGR idea: equidistant mesh avoids searching for active triangle
Grid index can be computed directly
- are evaluated with **highest efficiency** and **accuracy**
- no search for active triangles
- are available both in **ASCII** and **binary** format
- have **lossless compression**
- **dynamic patch loading** for virtually unlimited grid sizes
- minimum loading time
- may have a **curvilinear center line**
- can **export center line** for driver models
Maximum Center Line Curvature

center line curvature radius > ½ RGR width

center line curvature radius ≤ ½ RGR width

nearest point on center line becomes ambiguous
RGR Tools

- 2D visualization (single tracks)
- 3D visualization, meshed or rendered
- Generation of RGR roads out of any other supported road model
- Generation of RGR roads out of 2D spectral density formula expression
- Generation of RGR roads out of $z=f(x,y)$ formula expression
- Generation of RGR roads out of image data
- Processing of RGR roads:
  - Smoothing
  - Compression
  - Coarsening
  - Filtering
  - ASCII -> binary
  - Binary -> ASCII
  - Splitting into patches
- Generation of shl-files for road graphics in calling solver
Road surface measurement

www.3D-Mapping.de
Measured Belgian Block Road (2)

resolution 5 x 5 mm
„state-of-the-art“
Predicted Road Load On Belgian Block

Graph showing:
- Predicted road height in millimeters [mm] over time in seconds [s].
- Predicted forces in Newtons [N] over time in seconds [s].

The graphs display variations over time, indicating the dynamic behavior of road load and forces associated with Belgian blocks.
Predicted Road Load On Belgian Block: Road resolution

**red: 10x10 vs. 5x5 mm**
- difference 10x10 to 5x5: 6.7%
- difference 15x15 to 5x5: 11.5%

**blue: 15x15 vs. 5x5 mm**
- difference 10x10 to 5x5: 1.2%
- difference 15x15 to 5x5: 1.9%

- difference 10x10 to 5x5: 9.5%
- difference 15x15 to 5x5: 16.8%
Predicted Road Load On Belgian Block: Tread resolution

- red: 0.75 mm vs. 0.375 mm
  - difference 0.75mm to 0.375mm: 2.1%
  - difference 1.125mm to 0.375mm: 4.8%
- blue: 1.125 mm vs. 0.375 mm
  - difference 0.75mm to 0.375mm: 0.5%
  - difference 1.125mm to 0.375mm: 1.5%
### Efficiency Comparison for Belgian Block Road File Formats

<table>
<thead>
<tr>
<th></th>
<th>Triangulated Road 3D TeimOrbit</th>
<th>RGR ASCII data file</th>
<th>RGR binary data file</th>
<th>RGR binary data file + curved center line</th>
</tr>
</thead>
<tbody>
<tr>
<td># nodes</td>
<td>1.487 mio</td>
<td>1.487 mio</td>
<td>1.487 mio</td>
<td>1.487 mio</td>
</tr>
<tr>
<td># triangles</td>
<td>2.968 mio</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>file size</td>
<td>153.74 MB</td>
<td>12.88 MB</td>
<td>5.80 MB</td>
<td>6.02 MB</td>
</tr>
<tr>
<td>memory amount</td>
<td>237.80 MB</td>
<td>5.80 MB</td>
<td>5.80 MB</td>
<td>5.91 MB</td>
</tr>
<tr>
<td>file loading time</td>
<td>95.75 s</td>
<td>0.73 s</td>
<td>0.21 s</td>
<td>0.28 s</td>
</tr>
</tbody>
</table>
## CPU Time Comparison

### CPU Time for 1 Mio Road Evaluations

<table>
<thead>
<tr>
<th>mean evaluation point distance</th>
<th>Triangulated Road 3D TeimOrbit</th>
<th>RGR ASCII data file</th>
<th>RGR binary data file</th>
<th>RGR binary data file + curved center line</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.001mm</td>
<td>0.826 s</td>
<td>0.203 s</td>
<td>0.203 s</td>
<td>0.556 s</td>
</tr>
<tr>
<td>0.01mm</td>
<td>0.984 s</td>
<td>0.203 s</td>
<td>0.203 s</td>
<td>0.556 s</td>
</tr>
<tr>
<td>0.1mm</td>
<td>2.98 s</td>
<td>0.203 s</td>
<td>0.203 s</td>
<td>0.556 s</td>
</tr>
<tr>
<td>1mm</td>
<td>23.2 s</td>
<td>0.203 s</td>
<td>0.203 s</td>
<td>0.556 s</td>
</tr>
<tr>
<td>10mm</td>
<td>222 s</td>
<td>0.203 s</td>
<td>0.203 s</td>
<td>0.556 s</td>
</tr>
</tbody>
</table>
## CPU Time for Simulation on Belgian Block Road

<table>
<thead>
<tr>
<th>tread blocks per segment</th>
<th>total number of tread blocks</th>
<th>road evaluations per s</th>
<th>total CPU time per s with RGR road</th>
<th>total CPU time per s with TRIA road</th>
</tr>
</thead>
<tbody>
<tr>
<td>20</td>
<td>1600</td>
<td>4.15 mio</td>
<td>6.8 s</td>
<td>551 s</td>
</tr>
<tr>
<td>30</td>
<td>2400</td>
<td>5.80 mio</td>
<td>7.9 s</td>
<td>564 s</td>
</tr>
<tr>
<td>40</td>
<td>3200</td>
<td>7.45 mio</td>
<td>8.9 s</td>
<td>567 s</td>
</tr>
<tr>
<td>50</td>
<td>4000</td>
<td>9.11 mio</td>
<td>10.1 s</td>
<td>574 s</td>
</tr>
<tr>
<td>60</td>
<td>4800</td>
<td>10.76 mio</td>
<td>11.3 s</td>
<td>576 s</td>
</tr>
</tbody>
</table>
2D and 3D visualization

RGR file generation

RGR file processing

SHL and WaveFront file generation
Vehicle / Tire / Road Interfacing

- Rim position and velocity
- Spindle forces and moments
- Distributed contact forces
- Detailed surface geometry

Vehicle model
Tire model
Soil model
void soil_model_interface ( double x0,
                        double dx,
                        uint nx,
                        double y0,
                        double dy,
                        uint ny,
                        double phi,
                        double* Fx,
                        double* Fy,
                        double* Fz,
                        double* z,
                        double* vx,
                        double* vy,
                        double* vz,
                        double* mu,
                        int tire_handle,
                        int calling_mode,
                        double dt,
                        char* soil_model_data_file )
{
    /* calling_mode = 0: initialize soil model instance
    calling_mode = 1: call the soil model which applies the contact forces and updates its
        state variables according to time-step dt; compute and return new
        grid elevations and velocities for soil-model instance
    calling_mode = 99: terminate model instance
    */
    ...
CRI/Tools: RGR File Generation (1)

**Design Parameters**

- **p1**, **p2**, **p3**, **p4** may be used in data files and entry fields within all arithmetic expressions.

- **reset all sliders to default value 1.0**

- \( z(x,y) = \begin{cases} 
  p_1 \cdot 100 \cdot \max(0, \sin(p_2x + p_3y)) & \text{if } (x,y) \in (0.5, 9.5) \times (3, 7) \\
  0 & \text{else} 
\end{cases} \)
\[ z(x, y) = 100 \cdot p_1 \cdot (y > 0 & y < p_2) \cdot \text{mod}(x, p_2) \cdot y \]
\[ S(\Omega_x, \Omega_y) = \frac{10p_1}{\max \left(0.01, p_2 \Omega_x^3 + p_3 \Omega_y^3\right)} \]
CRI/Tools: RGR File Generation (4)

waviness $p_3 = 2.0$

waviness $p_3 = 2.5$

waviness $p_3 = 3.0$

x/y anisotropy factor $p_2 = 3.0$
CRI/Tools: RGR File Generation (5)

### COSIN/roadtools: generate new road data file

<table>
<thead>
<tr>
<th>apply</th>
<th>close</th>
<th>apply &amp; close</th>
<th>apply &amp; exit</th>
</tr>
</thead>
</table>

- **generate RGR file**
  - from other road data file
  - from 2D power spectral density (PSD)
  - from image file
  - from x/y-formula

- **generate RBF center-line file**
  - from COSIN track data

**New RGR file**: `/Users/micha/work/temp.rgr`

**Input image file**: `/Users/micha/MyTest_Pics/Altair_2009.png`

**Pixel distance**: 2.0 mm

- Height values at pure RGB colors: interpolation by RGB superposition
  - Red: 33.33 mm
  - Green: 33.33 mm
  - Blue: 33.33 mm

- Height values at white and black: interpolation using brightness
  - 200 mm
  - 0 mm

- Height values as general function of RGB values
  - \( z(r,g,b) = 33.33(r+g+b) \) for \( 0 \leq r,g,b \leq 1 \), \( z \) in mm
...thank you for your attention!

CRI, CTI, and FTire demo versions, papers, animations, documentation, version updates, and more at: www.cosin.eu