Peridigm

Peridigm Postproduction Guide

For Peridigm versions $\geq$ 1.4.1

Martin Rädel

DLR Deutsches Zentrum für Luft- und Raumfahrt
German Aerospace Center

DLR German Aerospace Center
Composite Structures and Adaptive Systems
Structural Mechanics
Dr. Tobias Wille

38108 Braunschweig
Germany

Tel: +49 (0)531 295-3701
Fax: +49 (0)531 295-3702
Web: http://www.dlr.de/fa/en

Martin Rädel
Tel: +49 (0)531 295-2048
Fax: +49 (0)531 295-2232
Mail: martin.raedel@dlr.de

Repository
This document is part of the PeriDoX repository.
The complete repository can be found at:
https://github.com/PeriDoX/PeriDoX

Citing
When citing this document, please reference the following:
Martin Rädel, Christian Willberg, Peridigm Users Guide, DLR-IB-FA-BS-2018-23, DLR Report, 2018

Contents

List of Figures

List of Listings

1. About

1.1. Scope

This document is supposed to contain informations about the postproduction of Peridigm results as pictures and charts in LaTeX documents or videos in presentations.

1.2. Contributors

This document represents a cumulative effort. The following list shows the contributors in alphabetic order by last name.

- Anna-Janina Bednarek
- Martin Rädel
- Lasse Wiedemann
- Christian Willberg

2. Animation editing for use in documents

2.1. Windows+Linux: Editing with *VLC Media Player*

The following sections describe the video editing using the tool *VLC Media Player*. The installation for *Windows* and *openSUSE* is described in the Peridigm Installation Guide from this same repository.

OGV- as well as AVI-video files are tested for the conversion. AVI is the common file extension for saved animations in *ParaView* and *Abaqus* under *Windows*. OGV is the only supported build-in extension of animations for *ParaView* in *openSUSE*.

2.1.1. Convert videos for use in LaTeX

For the approach to play videos in LaTeX-documents, either normal documents or beamer presentations, shown in section 3.5 a video in the MP4-format is required. Since MP4 itself is just a container-format and can contain several different codecs for video, audio and images, a working version is presented. This solution is tested but may not necessarily be the only or even best choice. This solution is based on the use of the H.264 MPEG video codec.

The resulting MP4-video can be used in LaTeX according to section 3.5.

GUI

1. Open *VLC Media Player*
2. Click *Media* in the menubar
3. Click *Convert/Save*
4. In the *File* tab:
   - click *Add...*
   - Select the video you want to convert to mp4-format and click *Open*
   - Click *Convert/Save* or just *Convert* via the little selection menu behind the downward facing triangle

5. In the *Convert*-windows:
   - Select *Convert*
   - From *Profile* selector choose *Video - H.265 + MP3 (MP4)*
   - Yes, H.265 instead of H.264
   - Click the profile preferences button  next to the profile selector
6. In the *Profile edition* window:
   - Go to *Video codec* tab
   - In the *Encoding parameters* tab select *H-264* as *Codec*
   - Go to *Audio codec* tab
   - Uncheck *Audio*
   - Click *Save*
7. Select a *Destination file* name and folder if required
8. Click *Start*

Command line

```
vlc --no-repeat --no-loop --rate=3.0 -I dummy RVE_Fatigue_r113360_udam.
    avi :sout=#transcode{vcodec=h264,vb=7750,scale=1,acodec=none}:std{
    access=file{no-overwrite},mux=mp4,dst=test2.mp4} vlc://quit
```

2.1.2. Convert videos for use in *PowerPoint*

Videos in *PowerPoint* work best when they are imported as a windows media video (WMV). Unfortunately, I have found no way to convert a video to a proper WMV-file using the VLC Media Player graphical user interface.

Fortunately, the command line is your friend, even in Windows. You can use the following line of code to convert the video `$INPUTFILE` to `$OUTPUTFILENAME.wmv`. To use this you can either add the path to `vlc.exe` to the `PATH`-variable and use:

```
vlc --no-repeat --no-loop $INPUTFILE.avi :sout=#transcode{vcodec=WMV2,vb
    =1800,scale=1,acodec=wma2,ab=128,channels=2,samplerate=44100}:std{
    access=file,mux=asf,dst=$OUTPUTFILENAME.wmv}
```

or it is possible to use the same command with the absolute path to `vlc.exe`:

```
D:\Programme\VideoLAN\VLC\vlc.exe --no-repeat --no-loop $INPUTFILE.avi :
    sout=#transcode{vcodec=WMV2,vb=1800,scale=1,acodec=wma2,ab=128,
    channels=2,samplerate=44100}:std{access=file,mux=asf,dst=
    $OUTPUTFILENAME.wmv}
```

In case the picture is pixelated after conversion, try to adjust the video bit-rate *vb*. To suppress the graphical user interface, add `-I dummy` to the options. To quit the command line version of VLC after completion, add `vlc://quit` at the end:

```
vlc --no-repeat --no-loop -I dummy $INPUTFILE.avi :sout=#transcode{vcodec
    =WMV2,vb=7750,scale=1,acodec=wma2,ab=128,channels=2,samplerate=44100}:
    std{access=file,mux=asf,dst=$OUTPUTFILENAME.wmv} vlc://quit
```

2.1.3. Common options

2.1.3.1. Change video speed

Using the GUI:

1. Open *VLC media player*
2. Click *Media* in the menubar
3. Click *Convert/Save*
4. In the *File* tab:
   - click *Add...*
   - Select the video you want to convert to mp4-format and click *Open*
   - Check *Show more options*
   - Add e.g. `:rate=4.0` to the *Edit options* line for a quadrupled spped
   - Procede like as shown in subsection 2.1.1

On the command line:

```
vlc $INPUTFILE :rate=4.0 :sout=#transcode{vcodec=WMV2,vb=1800,scale=1,
    acodec=wma2,ab=128,channels=2,samplerate=44100}:std{access=file,mux=
    asf,dst=$OUTPUTFILENAME.wmv}
```

2.2. Linux: Editing with FFmpeg library

To-Do

3. Documentation in LaTeX

3.1. Prerequisites

All figures in the following subsections are created with either the TikZ- or the pgfplots-package. Thus, these have to be installed and added to the preamble of the document.

```
\usepackage{pgfplots}
\usepackage{tikz}
```

Both packages use a modular, library-based system for extensions to the base package capabilities. These must be added if certain special features are requested. For this document the following libraries are used:

```
\usepgfplotslibrary{external}
\usepgfplotslibrary{groupplots}
\usetikzlibrary{arrows.meta}
\usetikzlibrary{calc}
\usetikzlibrary{decorations.markings}
\usetikzlibrary{intersections}
\usetikzlibrary{trees}
\usetikzlibrary{positioning}
\usetikzlibrary{spy}
```

In case the externalization of the TikZ-figure creation is used as shown in section 3.7.2, the installation of ImageMagick is required.

3.2. Tool-compliant legends in pgfplots

To create a tool-compliant legend two things are required: the colormap and the colorbar style.

3.2.1. *ParaView*, *Abaqus*, ... colormaps in pgfplots

To use a legend compliant with any free or commercial tool, a contiuous colormap, so the color gradient, has to be defined first. E.g. the definition of the (original?) *ParaView* colormap is specified here:

http://www.paraview.org/Wiki/images/b/be/All_idl_cmaps.xml

It should be possible to convert the exact definition to a `pgfplots` colormap. A rough approximation can be defined in the preamble with the following code-snippet. The colormaps for *ParaView* and *Abaqus* seem to be identical, at least according to color analysis in Gimp.

```
\pgfplotsset{
  colormap={abaqusblueredcolormap}{
    rgb255( 0cm)=(  0,  0,255);
    rgb255( 1cm)=(  0, 93,255);
    rgb255( 2cm)=(  0,185,255);
    rgb255( 3cm)=(  0,255,232);
    rgb255( 4cm)=(  0,255,139);
    rgb255( 5cm)=(  0,255,139);
    rgb255( 6cm)=(  0,255, 46);
    rgb255( 7cm)=( 46,255,  0);
    rgb255( 8cm)=(139,255,  0);
    rgb255( 9cm)=(232,255,  0);
    rgb255(10cm)=(255,185,  0);
    rgb255(11cm)=(255, 93,  0);
    rgb255(12cm)=(255,  0,  0);
  }
}

\pgfplotsset{
  colormap={paraviewblueredcolormap}{
    rgb255( 0cm)=(  0,  0,255);
    rgb255( 1cm)=(  0, 93,255);
    rgb255( 2cm)=(  0,185,255);
    rgb255( 3cm)=(  0,255,232);
    rgb255( 4cm)=(  0,255,139);
    rgb255( 5cm)=(  0,255,139);
    rgb255( 6cm)=(  0,255, 46);
    rgb255( 7cm)=( 46,255,  0);
    rgb255( 8cm)=(139,255,  0);
    rgb255( 9cm)=(232,255,  0);
    rgb255(10cm)=(255,185,  0);
    rgb255(11cm)=(255, 93,  0);
```

```
    rgb255(12cm)=(255,  0,  0);
  }
}
```

3.2.2. *ParaView*, *Abaqus*, ... colorbars in pgfplots

The colorbar style defines the number of discrete steps in the colorbar. Following are the definition of the default colorbars for *Abaqus* and *ANSYS* as well as an equivalent, but semi-continous, variant for *ParaView*.

```
\pgfplotsset{
  abaqusdiscrete12colorbar style/.style={
    separate axis lines,
    samples=13,                                % Number of steps+1
  }
}

\pgfplotsset{
  ansysdiscrete9colorbar style/.style={
    separate axis lines,
    samples=10,                                % Number of steps+1
  }
}

\pgfplotsset{
  paraviewdiscrete256colorbar style/.style={
    separate axis lines,
    samples=257,                               % Number of steps+1
  }
}
```

3.2.3. Application of colormaps and colorbars in pgfplots

Use the colormap with:

```
\begin{tikzpicture}
  \begin{axis}[
    ...
    colorbar,
    colormap name=paraviewblueredcolormap,
    colorbar style={
      paraviewdiscrete256colorbar style,
```

```
      ...
    },
    ...
  \end{axis}
  ...
\end{tikzpicture}
```

3.3. Create nice plots from screenshots with custom colorbar

3.3.1. Single screenshot

Once there is a nice result screenshot we want to use it in our LaTeX documentation. Therefore, we basically want to create everything besides the actual model picture in LaTeX. E.g. we want to add a colorbar or labels to the plot. One possibility to do that is the `pgfplots` package in LaTeX.

An exemplary result of a damage plot of the *Peridigm* disk impact model looks like figure 3.1.

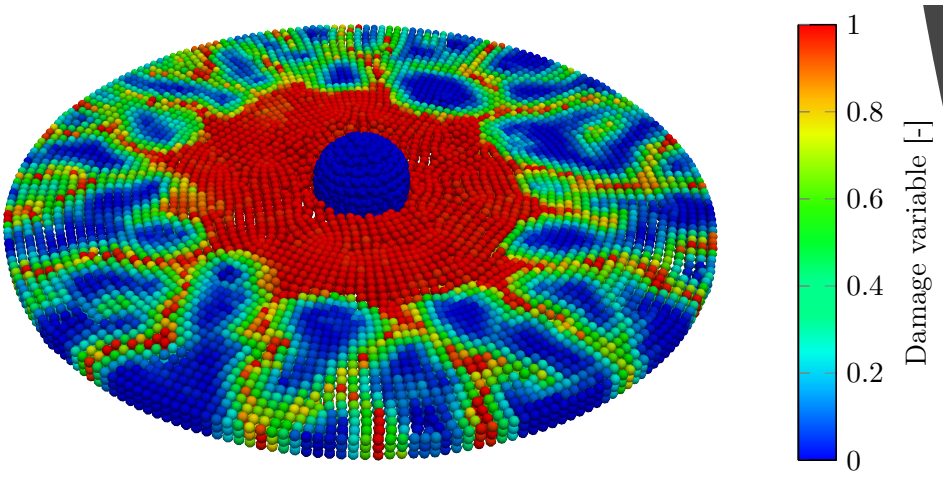
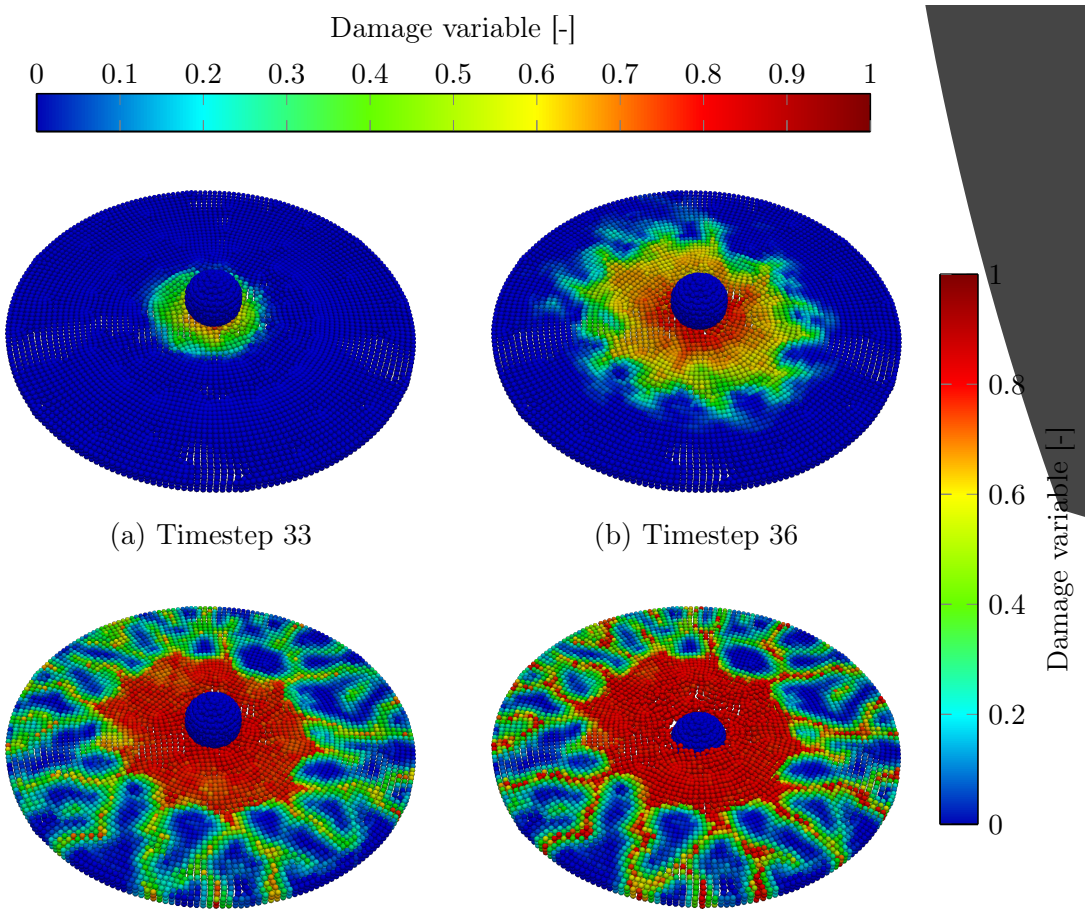


Figure 3.1.: Disk impact damage variable at timestep 47

The source code to create this figure is shown in section C.1 and can be downloaded from within this document.

3.3.2. Multiple screenshots

It may be necessary to show the development of a variable over multiple time steps sharing a common result data range. An example with colorbars above and right of the groupplot are shown in figure 3.2.



(a) Timestep 33

(b) Timestep 36

(c) Timestep 39

(d) Timestep 60

Figure 3.2.: Disk impact damage development

The source code to create this figure is shown in section C.2 and can be downloaded from within this document.

3.4. Putting labels on external images

The following picture is merely a demonstration of different ways to put labels on an external picture. An auxiliary grid is added to simplify positioning of labels. This grid

has to be switched of for official print publications.

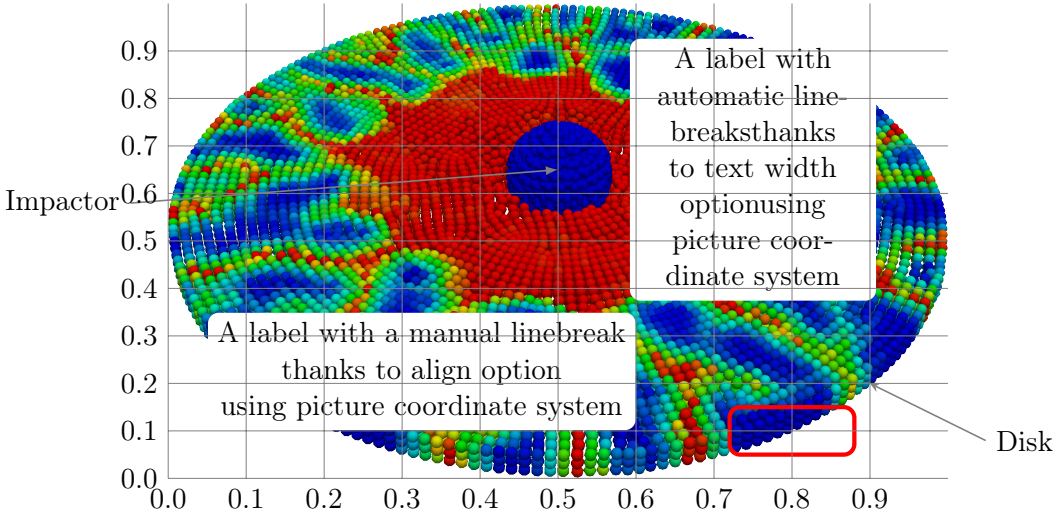


Figure 3.3.: Labels on external picture example

The source code to create this figure is shown in section C.3 and can be downloaded from within this document.

3.5. Use animations in beamer presentations

Prerequisite for the following points is the creation of an animation in *ParaView*. The animation is saved as an avi-video file.

- *ParaView* video animations can not be imported directly to LaTeX beamer presentations using the following approach based on the `media9`-package
- Convert the avi-video to a mp4-video with H.264-video- and mp3-audio-codec, e.g. with VLC player, see section 2.1
- Add `\usepackage{media9}` to the preamble of your LaTeX document
- Embed the video using the script from section C.4.3
- Play the video with Acrobat Reader under Windows

3.6. Create charts from result files

This needs `\usetikzlibrary{spy}` in the preamble of the document.

First, the common `pgfplotsset` for the plots are created. These are shown in section C.4.1. Add the code somewhere in your document. It can be in the preamble or right before the figures.

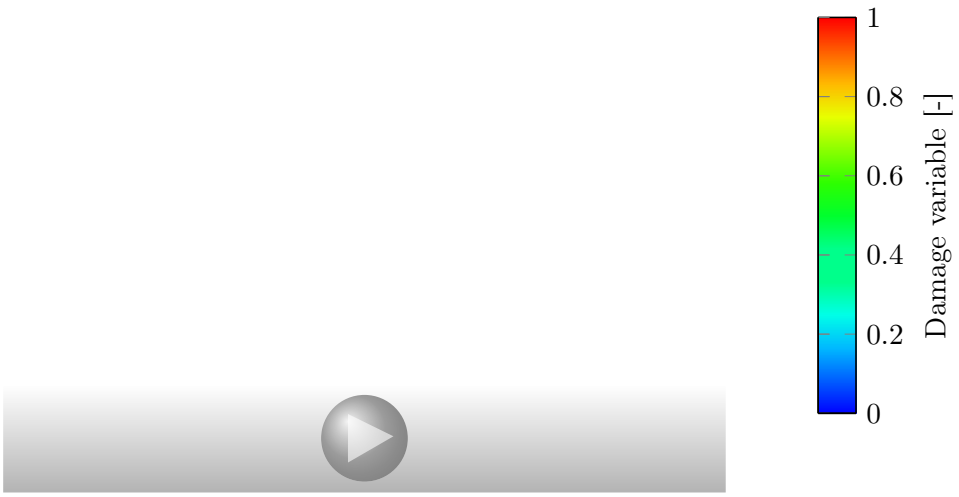


Figure 3.4.: Disk impact damage development video

Afterwards, the figure code can be included. The complete code is shown in section C.4.2. The result is shown in figure 3.5.

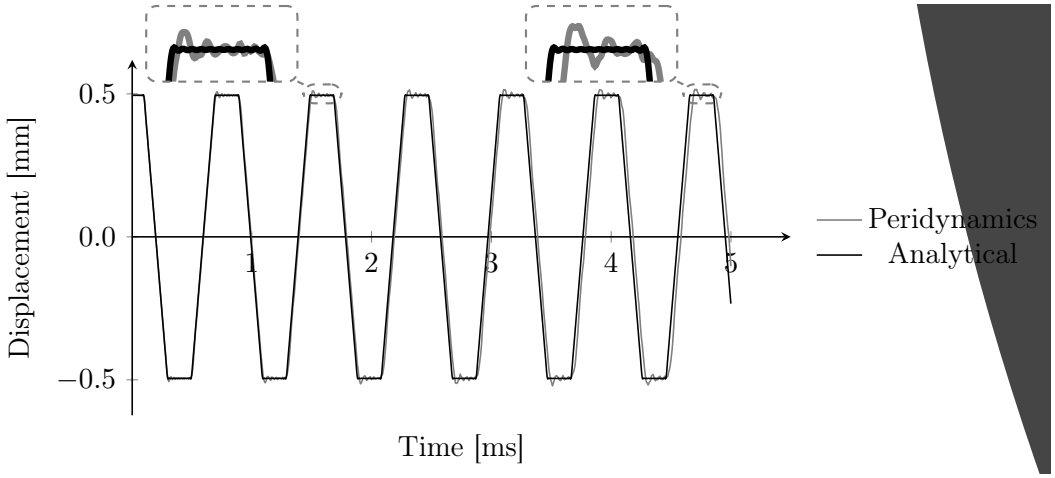


Figure 3.5.: Comparison of analytical and peridynamic solution of wave in bar

Since result files can contain a lot of data and therefore exaggerate the use of TeX-memory and compile time, consider externalization of the figure creation as explained in section 3.7.2.

3.7. Handling large result files in LaTeX

3.7.1. Increase TeX-memory

For large figures or plots of results using `pgfplots` the initial TeX memory might be insufficient. A first possibility to solve that problem is to increase the TeX memory.

3.7.1.1. MiKTeX on Windows

- Open a command prompt (cmd)
- Type `initexmf --edit-config-file pdflatex` and press Enter
- A file pops up in the editor
- Add `main_memory=8000000` to the file
- Save the file
- Go back to the command prompt
- Type `initexmf --dump=pdflatex` and press Enter
- After completion, close the command prompt

3.7.1.2. TeXLive on Linux

Untested:

- Open a terminal
- Type `kpsewhich texmf.cnf` and press Enter
- This gives the path to the `texmf.cnf` file
- Open the file
- Change or add `main_memory=8000000` to the file
- Save the file
- Go back to the command prompt
- Type `sudo fmtutil-sys --all` and press Enter
- After completion, close the command prompt

Be aware that using this method you modify the official config file which will be overwritten in an update. Thus, you have to repeat this step after an update. A better and update-independent solution is to create a local `texmf.cnf`-file and register it in the TeX path variables

3.7.2. Externalize TikZ- & pgfplots-figure creation

Creation of `TikZ` or `pgfplotsset`-figures can be quite memory and time consuming. The creation of these figures can be externalized to save time. This means the figure

is compiled once and written to a graphics file (pdf, png, ...). The next `pdflatex` run re-uses the created figure. It is not re-created for each compilation.

To achieve the externalization you need to add the following code to your preamble in addition to `\usepackage{pgfplots}`:

Listing 3.1: TikZ externalization code

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Header                           %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%
% This file handles all things considering the pgfplots externalization
%
% Output:
%
%  - Generates a pdf of every image where tikzexternalize is enabled
%  - Optionally also creates a png-copy of the pdf
%
% Requirements:
%
%  - This file must be included after a possible \makeindex[...] command
   from imakeidx-package
%  - shell-escape must be enabled during compilation with pdflatex in
   your TeX IDE:
%    - Kile:     Settings -> Configure Kile -> Tools -> Build -> PDFLaTeX
     -> Options:
%                --shell-escape -synctex=1 -interaction=nonstopmode %
   source
%    - TeXMaker: Optionen -> TexMaker Konfigurieren -> pdflatex:
%                pdflatex --shell-escape -synctex=1 -interaction=
   nonstopmode %.tex
%  - In case a png-copy is desired:
%    - Requires the installation of ImageMagick (www.imagemagick.org) to
   use the ``convert''/``magick'' command
%    - The path to ImageMagick binaries must be added to the system PATH
   variable
%    - Comment in one of the two lines starting with ``convert''/``magick
   '' below
%
% Revisions: 2016-03-06 Martin Raedel <martin.raedel@dlr.de>
%                       Initial draft
%            2017-11-22 Martin Raedel <martin.raedel@dlr.de>
%                       Added IDE Options for shell-escape
%
```

```
% Contact:   Martin Raedel,  martin.raedel@dlr.de
%            DLR Composite Structures and Adaptive Systems
%
%                                  __/|__
%                                 /_/_/_/
%            www.dlr.de/fa/en       |/ DLR
%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Content                                 %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

\makeatletter
\@ifpackageloaded{pgfplots}{}{
  \usepackage{pgfplots}
}
\makeatother

\usepgfplotslibrary{external}
\tikzexternalize[%
  mode=convert with system call,%
  shell escape=-enable-write18,%  % Use for MiKTeX
]
\tikzsetexternalprefix{ZZZ_TikZ/}     % Output folder
\tikzset{%
  external/system call={%
    pdflatex \tikzexternalcheckshellescape -halt-on-error %
    -interaction=batchmode -jobname "\image" "\texsource" %&& %
    %convert -density 600 -transparent white "\image.pdf" "\image.png" %
    for ImageMagick versions <7
    %magick  -density 600 -transparent white "\image.pdf" "\image.png" %
    for ImageMagick versions >= 7
  }
}
\tikzexternaldisable              % do not allow global externalization
                                  % of all tikz figures, call per figure
```

You can download the file from within this document.

Using one of the last two lines of the `tikzset` converts the pdf file of the figure created by `pdflatex` to a png file. It requires the installation of ImageMagick which provides the `convert` command for ImageMagick version prior to 7 or `magick` for version 7 or newer. For this to work, the path to the ImageMagick binaries `convert.exe` or `magick.exe` must be added to the system `PATH` environment variable.

In case you want to externalize the creation of a single figure add the following:

```
\tikzexternalenable                              % Enable externalization
\tikzsetnextfilename{Chart_Example}              % Figure file name
\input{Figures/LaTeX/Chart_Example}              % Path to figure
\tikzexternaldisable                             % Disable externalization
```

Beware, if you make changes to a figure you have to delete the present figure file before compiling your document. Otherwise, the update will not be effective because the old figure is re-used.

3.8. Tips and tricks with Biber

Appendices

A. This document

A.1. Repository

This document is part of the PeriDoX repository. The complete repository can be found at:

https://github.com/PeriDoX/PeriDoX

A.2. Typesetting

This document was originally typeset using the documentclass `dlrreprt` from the DLR-internal RM-LaTeX package.

The RM-LaTeX package is not publicly available. Therefore, this document is compatible with a bootstrap-version of the documentclass, called `bootstrap_dlrreprt`. `bootstrap_dlrreprt` class is part of this repository.

The compilation is performed with `pdflatex` with the following options:

```
pdflatex --shell-escape -synctex=1 -interaction=nonstopmode %source --
    extra-mem-top=60000000
```

The bibliography is compiled with `biber`. The glossary must be compiled with makeindex or, for Windows, the included batch-script may be used. The keyword index is created automatically.

The general compilation order is:

`pdflatex` → `biber` → `makeindex` → `pdflatex` → `pdflatex`

B. FAQ

B.1. LaTeX

B.2. Biber

I get an error running *biber* that says something like `encoding 'ascii': ascii "\xE2" does not map to Unicode at ....` What does that error mean. What can I do?

- There probably is a character in your bib-file that can not be read because it is not a Unicode character. This character must be replaced by proper LaTeX-syntax before the bib-file can be used.
- To find non-ASCII characters use either of the following possibilites:
  - Solution 1:
    - Open *Notepad++*
    - In the menu bar go to *Search* → *Find characters in range* → *Non-ASCII Characters (128-255)*
  - Solution 2:
    - Open *Notepad++*
    - Open a search (STRG-F)
    - Add expression `[^\x00-\x7F]`
    - In *Search Mode* check *Regular Expression*
    - Click *Find Next*
- Replace all non-ASCII characters and retry running *biber*

B.3. VLC

B.4. FFmpeg

C. LaTeX scripts

C.1. Single external pictures with colorbar

Below is the LaTeX code to produce figure 3.1. In order to use that script you need to specify `\usepackage{pgfplots}` in the preamble of your LaTeX document. Change all script values to your needs.

Listing C.1: LaTeX code for single external pictures with custom colorbar

```
\begin{tikzpicture}
\begin{axis}[
  hide axis,
  width=0.75\linewidth,                  % Width of the figure
  enlargelimits=false,                   % Do not add margins
  axis equal image,                      % Keep aspect ratio
  colormap/bluered,                      % Colormap preset
  colorbar,                              % Activate colorbar
  colorbar sampled,                      % Steps in colorbar
  colormap name=paraviewblueredcolormap,% Colormap
  colorbar style={
    separate axis lines,
    samples=256,                         % Number of steps
    ylabel=Damage variable [-],          % Label
    xshift=2em,                          % Shift for nicer appearance
  },
]
\addplot[%
  point meta min=0.0,                    % Minimum value colorbar
  point meta max=1.0,                    % Maximum value colorbar
] graphics [
  xmin=0,                                % Relative figure dimensions
  xmax=1.64,
  ymin=0,
  ymax=1,
]{Figures/Examples/Disk_Impact/Peridigm_Disk_Impact_47_Damage_ct};
\end{axis}
\end{tikzpicture}
```

You can download the file from within this document.

C.2. Multiple external pictures with one common colorbar

Below is the LaTeX code to produce figure 3.2. In order to use that script you need to specify `\usepackage{pgfplots}` in the preamble of your LaTeX document. Change all script values to your needs.

Listing C.2: LaTeX code for multiple external pictures with one common colorbar

```
\pgfplotsset{
  myaxis style/.style={
    hide axis,
    scale only axis,
    point meta min=0.0,                          % Minimum value colorbar
    point meta max=1.0,                          % Maximum value colorbar
    colormap/bluered,                            % Colormap preset
    colorbar sampled,                            % Steps in colorbar
  }
}
\pgfplotsset{
  mycolorbar style/.style={
    separate axis lines,
    samples=256,                                 % Number of steps
  }
}
\begin{tikzpicture}
  \pgfmathsetmacro{\xmin}{0}
  \pgfmathsetmacro{\xmax}{1.3544}
  \pgfmathsetmacro{\ymin}{0}
  \pgfmathsetmacro{\ymax}{1}
  \begin{groupplot}[
    group style={
      group name=my plots,
      group size= 2 by 4,
      vertical sep=1.5cm
    },
    width=7cm,
    hide axis,
    enlargelimits=false,
    axis equal image,
  ]
    \nextgroupplot
      \addplot graphics [xmin=\xmin,xmax=\xmax,ymin=\ymin,ymax=\ymax]
                    {Figures/Examples/Disk_Impact/
```

```
    Peridigm_Disk_Impact_33_Damage_ct};
      \coordinate (top) at (rel axis cs:0,1); % Top of 1st plot
    \nextgroupplot
%    [
%         point meta min=0.0,
%         point meta max=5.0,
    \nextgroupplot
      \addplot graphics [xmin=\xmin,xmax=\xmax,ymin=\ymin,ymax=\ymax]{
    Figures/Examples/Disk_Impact/Peridigm_Disk_Impact_60_Damage_ct};
      \coordinate (bot) at (rel axis cs:1,0); % Bottom of last plot
  \end{groupplot}
  % Labels
  \node[below = 0.25cm of my plots c1r1.south] {(a) Timestep 33};
  \node[below = 0.25cm of my plots c2r1.south] {(b) Timestep 36};
  \node[below = 0.25cm of my plots c1r2.south] {(c) Timestep 39};
  \node[below = 0.25cm of my plots c2r2.south] {(d) Timestep 60};
  % Node position middle right groupplot
  \path (top -| current bounding box.east) --
        coordinate(legendposright)
        (bot-|current bounding box.east);
  % Node position middle above groupplot
  \path (top |- current bounding box.north) --
        coordinate(legendposabove)
        (bot|-current bounding box.north);
  % Node markers
  %\node [circle,fill=black,minimum size=10pt] at (legendposright){};
  %\node [circle,fill=black,minimum size=10pt] at (legendposabove){};
  %Colorbar above
  \begin{axis}[%
    myaxis style,
    yshift=1.5cm,
    anchor=south,
    colorbar horizontal,                        % Active colorbar
    colorbar style={
      mycolorbar style,
      xticklabel pos=upper,
      xlabel=Damage variable [-],               % Label
    },
    %every colorbar/.append style={
    %  width=\pgfkeysvalueof{/pgfplots/parent axis width}%+
    %             %\pgfkeysvalueof{/pgfplots/group/vertical sep}
    %}
  ]
    \addplot [draw=none] coordinate {(0,0)};
```

```
    at={(legendposright.east)},
    anchor=west,
    xshift=-4.0cm,
    colorbar right,                          % Activate colorbar
    colorbar style={
      mycolorbar style,
      ylabel=Damage variable [-],            % Label
    },
    every colorbar/.append style={
      height=\pgfkeysvalueof{/pgfplots/parent axis height}%+
                  %\pgfkeysvalueof{/pgfplots/group/vertical sep}
    }
  ]
    \addplot [draw=none] coordinates {(0,0)}; % Dummy plot
  \end{axis}
\end{tikzpicture}
```

You can download the file from within this document.

C.3. Labels in and on external pictures

Listing C.3: LaTeX code for labels on external picture

```
\begin{figure}[htbp]
\centering
\begin{tikzpicture}
% External figure
\node[anchor=south west,inner sep=0] (image) at (0,0) {\includegraphics[
    width=0.7\linewidth]{Figures/Examples/Disk_Impact/
    Peridigm_Disk_Impact_47_Damage_ct}};
% Relative node positioning to picture
\node[anchor=east] (impactorlabel) at ($(image.west)       + (-0.5cm,
    0.5)$) {Impactor};
\node[anchor=west] (disklabel)     at ($(image.south east) + ( 0.5cm,0.5)
    $) {Disk};
% Figure scope
\begin{scope}[
  x={(image.south east)},
  y={(image.north west)},
]
  % Some label
  \node [
    align=center,
    fill=white,
    rounded corners,
    anchor=south west
  ] at (0.05,0.1) {%
    A label with a manual linebreak\\%
    thanks to align option\\%
    using picture coordinate system%
  };
  % Some label
  \node [
    text width=3cm,
    align=center,
    fill=white,
    rounded corners
  ] at (0.75,0.65) {%
    A label with automatic linebreaks%
    thanks to text width option%
    using picture coordinate system%
  };
  % Arrows
```

```
  \draw[-latex, gray]   (impactorlabel.east) -- (0.5,0.65);
  \draw[-stealth, gray] (disklabel.west)     -- (0.9,0.2);
  % Highlight
  \draw[red,ultra thick,rounded corners] (0.72,0.05) rectangle
    (0.88,0.15);
  % Help grid and labels
  \draw[help lines,xstep=.1,ystep=.1] (0,0) grid (1,1);
  \foreach \x in {0,1,...,9} {\node [anchor=north] at (\x/10,0) {0.\x};}
  \foreach \y in {0,1,...,9} {\node [anchor=east]  at (0,\y/10) {0.\y};}
\end{scope}
% Outside label
\end{tikzpicture}
\caption{Labels on external picture example}
\label{fig:Labels_on_external_picture}
\end{figure}
```

You can download the file from within this document.

C.4. Nice charts

C.4.1. pgfplotsset

Listing C.4: Pgfplotssets for charts

```
\tikzset{
  every picture/.style=semithick              % Adjust default line width
}

\pgfplotsset{
  chart style/.style={
    width=0.7\linewidth,
    height=0.35\textheight,
    axis x line=middle,                       % Middle x-axis
    axis y line=left,
    enlarge x limits={auto,upper},            % Add this at positive x
    enlarge y limits,                         % Add this at y
    x label style={                           % xlabel style
      at={(axis description cs:0.5,-0.1)},    % Position
      anchor=north                            % Anchor
    },
    y label style={                           % ylabel style
      anchor=south                            % Anchor
    },
    x tick label style={
      /pgf/number format/fixed,
      /pgf/number format/precision=5,         % Nr of decimal digits
    },
    y tick label style={
      /pgf/number format/fixed,
      /pgf/number format/fixed zerofill,      % Allow trailing zeros
      /pgf/number format/precision=1,         % Nr of decimal digits
    },
    scaled ticks=false,
    legend columns=1,                         % Nr of colums in legend
    legend style={
      draw=none,                              % No border
      fill=none,                              % No fill color
      at={(1.025,0.5)},                       % Position
      anchor=west,                            % Anchor
      %transpose legend,
      %reverse legend,
      %/tikz/column 2/.style={
```

```
      %  column sep=5pt,
      %}
      /tikz/row 2/.style={
        row sep=5pt,
      }
    }
  }
}

\pgfplotsset{
  xaxis style/.style={
    each nth point=1,
    restrict expr to domain={rawx}{0:5},      % restrict x-axis values
  }
}
```

You can download the file from within this document.

C.4.2. Figure code

Listing C.5: Pgfplots chart example

```
\begin{tikzpicture}[%
  spy using outlines={%
    rectangle,                                % Spy form
    rounded corners,                          % Spy edge shape
    magnification=4,                          % Magnification factor
    height=0.6cm,                             % Height in original chart
    width=1.25cm,                             % Width in original chart
    dashed,                                   % Dashed line
    draw=gray,                                % Line color
    every spy on node/.append style={thick},  % Line style for spy
    connect spies,                            % Connect orig. & detail
    spy connection path={                     % Line style for spy
   connection
      \draw[thick] (tikzspyonnode) -- (tikzspyinnode);
    },
  }
]
  \begin{axis}[
    chart style,
    xlabel={Time [ms]},
    ylabel={Displacement [mm]},
```

```
  ]
    % Graph 1
    \addplot[
      gray,                                  % Plot color
      xaxis style,                           % Predefined style
    ] table [
      x expr=\thisrowno{0}*1000,             % Scale to ms
      y expr=\thisrowno{1}*1000,             % Scale to mm
    ]{Results/coord_disp_pd_nt100.txt};      % Input file
    \addlegendentry{Peridynamics}            % Legend entry
    % Graph 2
    \addplot[
      black,                                 % Plot color
      xaxis style,                           % Predefined style
    ] table [
      x expr=\thisrowno{0}*1000,             % Scale to ms
      y expr=\thisrowno{2}*1000,             % Scale to mm
    ]{Results/coord_disp_pd_nt100.txt};      % Input file
    \addlegendentry{Analytical}              % Legend entry
    % Spy 1
    \coordinate (spypoint1) at (axis cs:1.59,0.5);
    \coordinate (spyviewer1) at (axis cs:0.75,0.675);
    \spy[
      width=2.0cm,
      height=1.0cm,
    ] on (spypoint1) in node [
      fill=white,
      draw=gray,
      dashed
    ] at (spyviewer1);
    % Spy 2
    \coordinate (spypoint2) at (axis cs:4.76,0.5);
    \coordinate (spyviewer2) at (axis cs:3.92,0.675);
    \spy[
      width=2.0cm,
      height=1.0cm,
    ] on (spypoint2) in node [
      fill=white,
      draw=gray,
      dashed
    ] at (spyviewer2);
  \end{axis}
\end{tikzpicture}
```

You can download the file from within this document.

C.4.3. LaTeX beamer video with custom colorbar

Below is the LaTeX code to produce figure 3.4. In order to use that script you need to specify `\usepackage{media9}` in the preamble of your LaTeX document. Change all script values to your needs.

Listing C.6: LaTeX beamer code for a frame with video

```
\begin{frame}

\begin{itemize}
 \item Video demonstration: just click on the disk
 \item Works with Adobe Reader on Windows and ? on Linux
\end{itemize}

\begin{center}
\begin{tikzpicture}
  % Axis style
  \pgfplotsset{
    myaxis style/.style={
      hide axis,
      scale only axis,
      point meta min=0.0,                        % Minimum value colorbar
      point meta max=1.0,                        % Maximum value colorbar
      colormap/bluered,                          % Colormap preset
      colorbar sampled,                          % Steps in colorbar
    }
  }
  % Colorbar style
  \pgfplotsset{
    mycolorbar style/.style={
      separate axis lines,
      samples=256,                               % Number of steps
    }
  }
  % Video
  \node (video) {
    \includemedia[
      height=0.65\textheight,
      width=0.65\textwidth,
      keepaspectratio,
      activate=pageopen,
      addresource=Videos/Peridigm_Disk_Impact_Damage.mp4,
      flashvars={source=Videos/Peridigm_Disk_Impact_Damage.mp4}
      ]{}{VPlayer.swf}
```

```
  };
  % Colorbar right
  \begin{axis}[%
    myaxis style,
    at={(video.east)},
    anchor=east,
    height=0.65\textheight,
    xshift=0.25cm,
    colorbar right,                          % Activate colorbar
    colorbar style={
      mycolorbar style,
      ylabel=Damage variable [-],            % Label
    },
  ]
    \addplot [draw=none] coordinates {(0,0)}; % Dummy plot
  \end{axis}
\end{tikzpicture}
\end{center}

\end{frame}
```

You can download the file from within this document.

BSD Documentation License