

pes — Potential energy surface diagrams*

Sverre Løyland[†]

Released 2018/06/07

Abstract

The `pes` package aims to make it easy to draw potential energy surface diagrams while being highly customizable and extendable. The package defines a `pes` environment in which the `\level` and `\edge` macros work.

Contents

1	Using the package	1
1.1	Package loading and package options	1
1.2	The <code>pes</code> environment	2
2	Complete example	2
3	Implementation	3
3.1	Package options	3
3.2	Environment options	4
	Change History	6
	Index	6

1 Using the package

1.1 Package loading and package options

<code>env</code>	The package is loaded normally using
<code>width</code>	<code>\usepackage{pes}</code>
<code>height</code>	
<code>xlabel</code>	The packages can take a series of options. Every package option is locally overridable by
<code>ylabel</code>	environment options in the <code>pes</code> environment. The options sets the default values for the
	<code>pes</code> environment options. See Section 1.2 for more details on each option.

1.2 The `pes` environment

pes This package provides the `pes` environment to draw the potential energy surface diagrams. The environment draws appropriate axis and allows for using the `\level` and `\edge` macros to draw energy levels (eg. ground states and transition states) and connecting edges between the levels respectively.

env The environment consists actually of an outer `figure` environment with a `\centering` macro. This outer environment can be changed by using the `env` option, eg. the `sidewaysfigure` environment from the `rotating` package.

```
\usepackage{rotating}
\begin{pes}[env=sidewaysfigure]
...
\end{pes}
```

Supplying `none` to the `env` option removes the floating `figure` environment and the `\centering` macro.

caption The outer environment can have a caption and a label as is common with `figure`
label environments. These can be specified using the `caption` and `label` options.

```
\begin{pes}[caption={This is a potential energy surface},
label={fig:pes}]
...
\end{pes}
```

width The `width` and `height` options specify the axis's width and height, respectively.

height The `xlabel` and `ylabel` specify the x and y axis's labels, respectively.

xlabel All the levels in the `pes` environment can be shifted by the `zero` option.

ylabel Inside the outer environment is a `tikz tikzpicture` environment with a `pgfplots`
zero `axis` environment with appropriate options. When macros are used inside the `pes` environment, they are placed inside the `axis` environment so other `tikz`, `pgf`, `pgfplots` etc. macros can be used as well. If you want to use these functions, you should study the source code of the package to interact correctly with its elements.

```
\level {<name>} {<x coordinate>} {<y coordinate>} [<graphics>]
```

The `\level` macro has to be used inside the `pes` environment. The macro takes the name, x coordinate and y coordinate as arguments with an optional fourth argument for displaying a graphic, eg. a molecular structure.

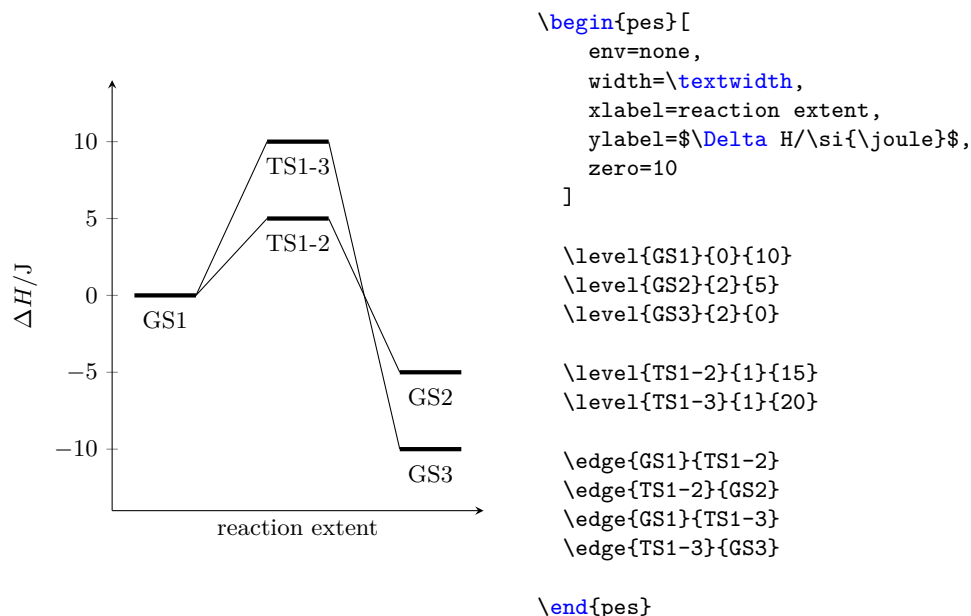
```
\edge {<left level>} {<right level>}
```

The `\edge` macro has to be used inside the `pes` environment. The macro takes the left of the left level and right level as arguments.

2 Complete example

*This file describes version v1.0, last revised 2018/06/07.

†Contact: <https://github.com/sver1/pes>



3 Implementation

```

1 <*package>
2 <@@=pes>
   Version data and required packages and setup.
3 \ProvidesExplPackage
4   {pes} {2018/06/07} {1.0}
5   {Potential energy surface diagrams}
6
7 \RequirePackage{siunitx}
8 \DeclareSIUnit[number-unit-product = {}]{\Hartree}{Ha}
9
10 \RequirePackage{pgfplots}
11 \pgfplotsset{compat=1.15}

```

3.1 Package options

The package takes a series of options which becomes the default for each environment unless specified in the specific environment.

```

  \__pes_env
  \__pes_width
  \__pes_height
  \__pes_xlabel
  \__pes_ylabel
12 \keys_define:nn {pes} {
13   env .tl_set:N = \__pes_env,
14   env .initial:n = {figure},
15
16   width .tl_set:N = \__pes_width,
17   width .initial:n = {\axisdefaultwidth},
18
19   height .tl_set:N = \__pes_height,
20   height .initial:n = {\axisdefaultheight},
21

```

```

22 xlabel .tl_set:N = \__pes_xlabel,
23 xlabel .initial:n = {reaction-coordinate},
24
25 ylabel .tl_set:N = \__pes_ylabel,
26 ylabel .initial:n = {\$E/\si{\Hartree}$},
27 }
28 \ProcessKeysOptions{pes}

```

(End definition for __pes_env and others.)

3.2 Environment options

The environments takes series of options to specify the styles. If the options are already defined in the package environment, they will be overridden inside the specific environment.

```

\__pes_env
\__pes_width 29 \keys_define:nn {pes/env} {
\__pes_height 30   env .choice:,
\__pes_xlabel 31   env / none .code:n = { \tl_set_eq:NN \__pes_env \c__pes_env_none },
\__pes_ylabel 32   env / .code:n = { \tl_set_eq:NN \__pes_env figure },
\__pes_caption 33   env / unknown .code:n = { \tl_set_eq:NN \__pes_env #1 },
\__pes_label 34
\__pes_zero 35   width .tl_set:N = \__pes_width,
36   width .default:n = {\axisdefaultwidth},
37
38   height .tl_set:N = \__pes_height,
39   height .default:n = {\axisdefaultheight},
40
41   xlabel .tl_set:N = \__pes_xlabel,
42   xlabel .default:n = {reaction-coordinate},
43
44   ylabel .tl_set:N = \__pes_ylabel,
45   ylabel .default:n = {\$E/\si{\Hartree}$},
46
47   caption .tl_set:N = \__pes_caption,
48
49   label .tl_set:N = \__pes_label,
50
51   zero .tl_set:N = \__pes_zero,
52   zero .initial:n = {0},
53 }

```

(End definition for __pes_env and others.)

pes The pes environment

```

54 \NewDocumentEnvironment{pes}{0{}}{
55   \keys_set:nn {pes/env}{#1}
56
57   \tl_if_eq:NNTF \__pes_env \__pes_env_none {} {
58     \begin{\__pes_env}
59     \centering
60   }
61   \begin{tikzpicture}

```

```

62     \begin{axis}[width=\__pes_width,
63     height=\__pes_height,
64     axis~lines=left,
65     enlarge~x~limits=0.2,
66     enlarge~y~limits=0.2,
67     xlabel=\__pes_xlabel,
68     ylabel=\__pes_ylabel,
69     xmajor ticks=false]
70   }{
71     \end{axis}
72     \end{tikzpicture}
73
74     \tl_if_empty:NF \__pes_caption {\caption{\__pes_caption}}
75     \tl_if_empty:NF \__pes_label {\label{\__pes_label}}
76
77     \tl_if_eq:NNTF \__pes_env \__pes_env_none {} {
78       \end{\__pes_env}
79     }
80   }

```

\level Draw a level by plotting a point at the desired coordinate, drawing a horizontal line, adding the label underneath and optionally adding a graphic above. The first argument is the label, the second is the x -coordinate, the third is the y -coordinate and the fourth is an optional graphic. The `coord`, `level`, `levelline` and `levellabel` styles specify the styles of the level.

```

81 \tikzset{coord/.style={mark=none}}
82 \tikzset{level/.style={minimum~width=8mm}}
83 \tikzset{levelline/.style={ultra~thick}}
84 \tikzset{levellabel/.style={node~distance=1em}}
85 \NewDocumentCommand{\level}{mmm}{
86   \addplot[coord] coordinates{(#2,#3-\__pes_zero)}
87   \node[#1][level]{#4}
88   \IfValueT{#4}{
89     \node[graphics]{#4}
90   };
91   \draw[levelline] (#1.west)--(#1.east);
92   \node[levellabel,below~of=#1]{#1};
93 }

```

(End definition for `\level`. This function is documented on page 2.)

\edge Draw an edge between two levels. The arguments are the names of the leftmost and rightmost levels respectively. The style is specified by the `edge` style.

```

94 \tikzset{edge/.style={}}
95 \NewDocumentCommand{\edge}{mm}{
96   \draw[edge] (#1.east)--(#2.west);
97 }

```

(End definition for `\edge`. This function is documented on page 2.)

```

98 \end{package}

```

Change History

v1.0

General: Initial public release 1

Index

The italic numbers denote the pages where the corresponding entry is described, numbers underlined point to the definition, all others indicate the places where it is used.

C	
caption (option)	2
caption internal commands:	
<code>__pes_caption</code>	<u>29</u> , 74
E	
<code>\edge</code>	1, 2, 2, <u>94</u>
env (option)	1, 2
env internal commands:	
<code>__pes_env</code>	<u>12</u> , <u>29</u> , 57, 58, 77, 78
environments:	
pes	2, <u>54</u>
H	
height (option)	1, 2
height internal commands:	
<code>__pes_height</code>	<u>12</u> , <u>29</u> , 63
L	
label (option)	2
label internal commands:	
<code>__pes_label</code>	<u>29</u> , 75
<code>\level</code>	1, 2, 2, <u>81</u>
O	
options:	
caption	2
env	1, 2
height	1, 2
label	2
width	1, 2
xlabel	1, 2
P	
pes (environment)	2, <u>54</u>
pes internal commands:	
<code>__pes_env_none</code>	57, 77
<code>\c__pes_env_none</code>	31
T	
tl commands:	
<code>\tl_if_empty:NTF</code>	74, 75
W	
width (option)	1, 2
width internal commands:	
<code>__pes_width</code>	<u>12</u> , <u>29</u> , 62
X	
xlabel (option)	1, 2
xlabel internal commands:	
<code>__pes_xlabel</code>	<u>12</u> , <u>29</u> , 67
Y	
ylabel (option)	1, 2
ylabel internal commands:	
<code>__pes_ylabel</code>	<u>12</u> , <u>29</u> , 68
Z	
zero (option)	2
zero internal commands:	
<code>__pes_zero</code>	<u>29</u> , 86