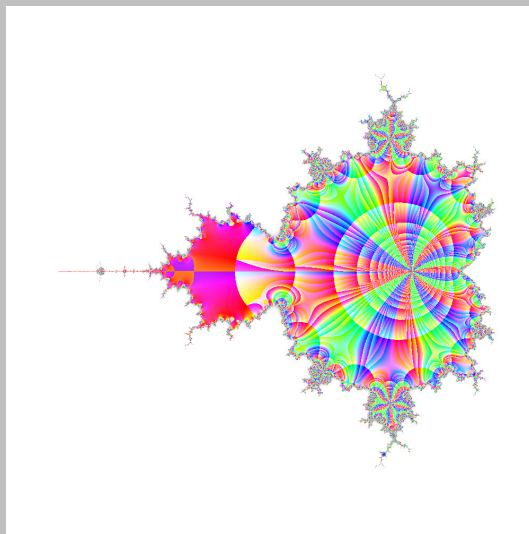


PSTricks

Domain Coloring of complex functions version 0.02

August 22, 2024



Package author(s):
Herbert Voß

Contents

1	Introduction	2
1.1	Loading the package	2
1.2	Using the macro	2
1.3	Options	2
2	Examples	2
2.1	The default with function $f(z) = z$	2
2.2	Defining domain, color mode, resolution and filename	3
2.3	Option for <code>\includegraphics</code>	4
2.4	Higher resolution	4
2.5	hsv to rgb conversion	5
2.6	External function definition	6

1 Introduction

This package works only with `lualatex`!

1.1 Loading the package

The package `domaincoloring` creates a colored interpretation of the domain of a complex function. The package itself has no options and should be loaded as usual:

```
\usepackage{domaincoloring}
```

The needs the following Lua modules:

- `domaincoloring.lua` the main module
- `domaincoloring-complex-numbers.lua` for complex math operations
- `domaincoloring-functions.lua` for predefined complex functions

The function module has to be managed by the user himself, if needed.

1.2 Using the macro

There is only one macro which does the external call of the Lua program `domaincoloring.lua`. This program creates the image which is then included into the document. The $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ -run needs the `--shell-escape` option to allow the external run of the program to convert the created ppm-file into a png file..

```
\DomainColoring[options]{complex function in Lua notation}
```

Every math function has to be preceded by `cmath` if it has a complex argument.

1.3 Options

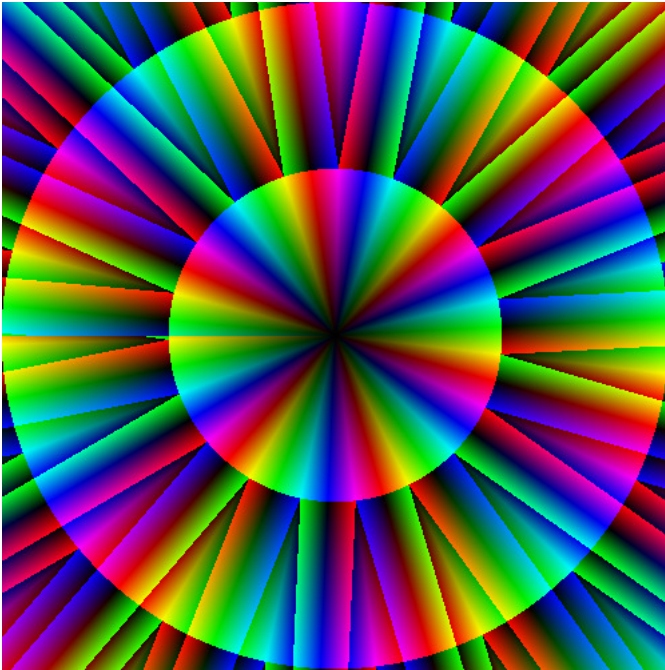
<i>name</i>	<i>value</i>	<i>meaning</i>
<code>domain</code>	<code>-2,2,-2,2</code>	the (re,im)-coordinates for the complex system
<code>resolution</code>	<code>500</code>	the number of steps for the re,im interval. At the moment only for both coordinates the same. Will be in future versions different for the both axes.
<code>filename</code>	<code>\jobname-tmp</code>	name of the created external file, must be unique
<code>grfOptions</code>	<code>scale=0.5</code>	optional arguments for <code>\includegraphics</code>

hsvrgb	phi,1,r	for the conversion into rgb
funcName		corresponding to external file
bgcolor	0	change color to white as background for all values $r + g + b \leq \text{bgcolor}$
invers	false	inverted colors with $\text{color} = \text{color} - 255 $

2 Examples

2.1 The default with function $f(z) = z$

```
\DomainColoring{z}\quad % default filename \jobname-tmp.png
```

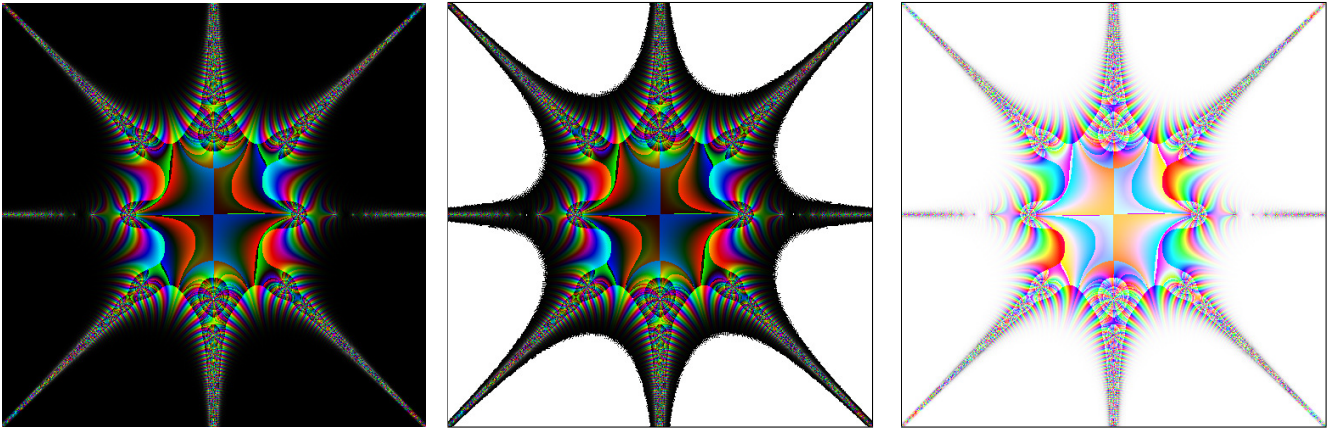


2.2 Defining domain, color mode, resolution and filename

$$f(z) = \cos(z) / \sin(z^4 - 1) \quad (1)$$

in Lua-notation: `cmath.cos(z)/cmath.sin(z^4-1)`.

```
\DomainColoring[domain={-2.5,2.5,-2.5,2.5},resolution=500,
  grfOptions={width=0.32\linewidth},
  filename=\jobname-tmpla]{cmath.cos(z)/cmath.sin(z^4-1)}
\hfill
\frame{\DomainColoring[domain={-2.5,2.5,-2.5,2.5},resolution=500,
  bgcolor=true,grfOptions={width=0.32\linewidth},bgcolor=3,
  filename=\jobname-tmplb]{cmath.cos(z)/cmath.sin(z^4-1)}}
\hfill
\frame{\DomainColoring[domain={-2.5,2.5,-2.5,2.5},resolution=500,
  invers=true,,grfOptions={width=0.32\linewidth},
  filename=\jobname-tmplc]{cmath.cos(z)/cmath.sin(z^4-1)}}
```

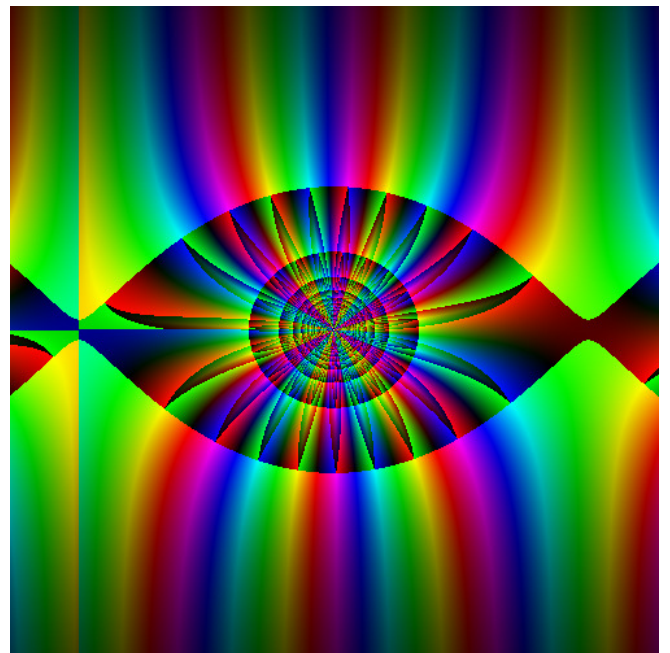
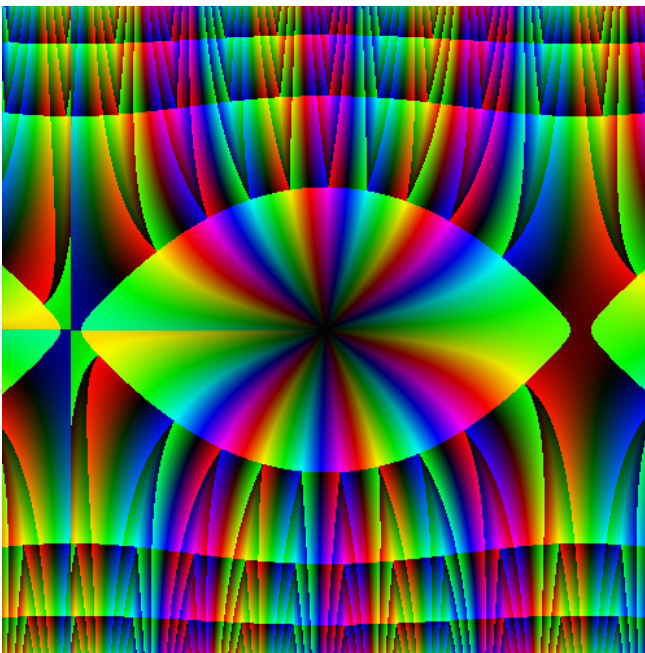


With `\bgcolor=<int>` all RGB-combinations with a sum $R + G + B \leq \text{int}$ are set to the color white.

2.3 Option for `\includegraphics`

With `grfOptions` one can define optional arguments for `\includegraphics`:

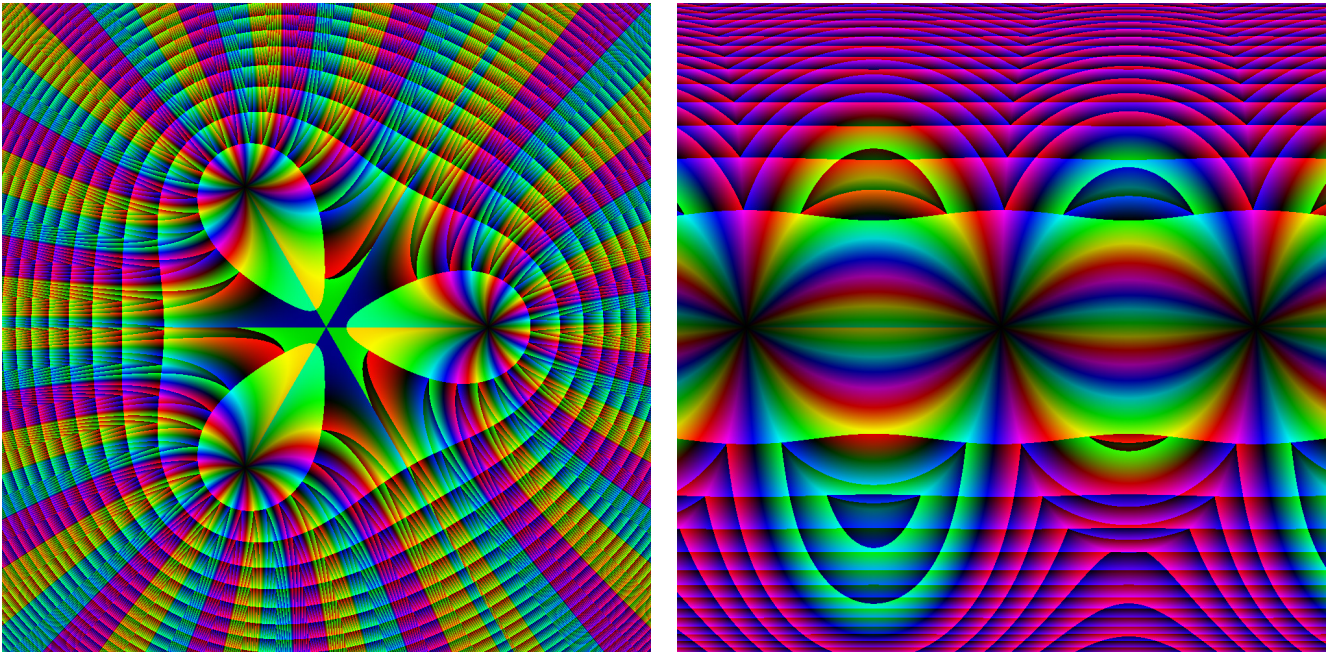
```
\DomainColoring[grfOptions={width=0.49\linewidth}, filename=\jobname-tmp2a]{\cmath.sin(z)}\hfill
\hfill
\DomainColoring[grfOptions={width=0.49\linewidth}, filename=\jobname-tmp2b]{1/\cmath.sin(z)}
```



2.4 Higher resolution

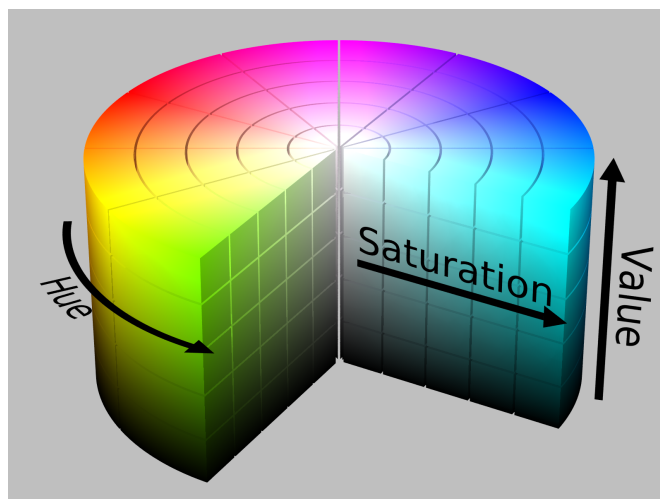
The resolution is more or less the number of pixels for the given domain. In later versions of this package it may be possible to have different values for the two coordinates. By now the one value for resolution is for both axes.

```
\DomainColoring[filename=\jobname-tmp3, resolution=1000,
  grfOptions={width=0.49\linewidth}]{z^3-1}
\hfill
\DomainColoring[filename=\jobname-tmp4, resolution=1000,
  grfOptions={width=0.49\linewidth}]{(z+1)^2*(z-1)/((z+i)*(z-i)^2)}
```

2.5 hsv to rgb conversion

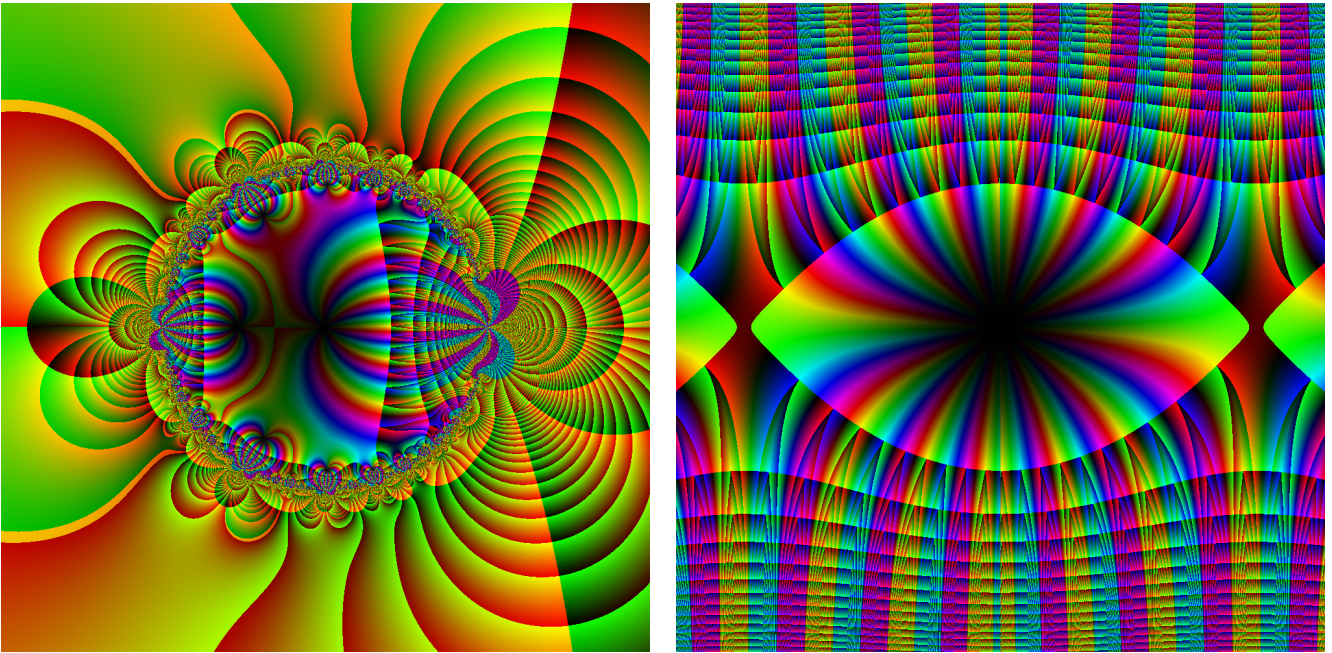
The color model (Wikipedia)



http://en.wikipedia.org/wiki/File:HSV_color_solid_cylinder_alpha_lowgamma.png

The complex number $z = x + iy$ is converted into its trigonometrical representation $x = r \cdot \cos \phi$ and $y = r \cdot \sin \phi$ with $r = \sqrt{x^2 + y^2}$. The values r and ϕ are now taken as values for the hsv color model with a constant second value for saturation. ϕ is used for hue. For example: `hsvrgb=phi,1,r`, which gives

```
\DomainColoring[filename=\jobname-tmp5a,resolution=1000,
  grfOptions={width=0.49\linewidth},hsvrgb={phi,1,r}]{cmath.sin(z)*cmath.sin(0.99*z)}
\hfill
\DomainColoring[filename=\jobname-tmp5b,resolution=1000,
  grfOptions={width=0.49\linewidth}]{cmath.sin(z)*cmath.sin(0.99*z)} % the default
```



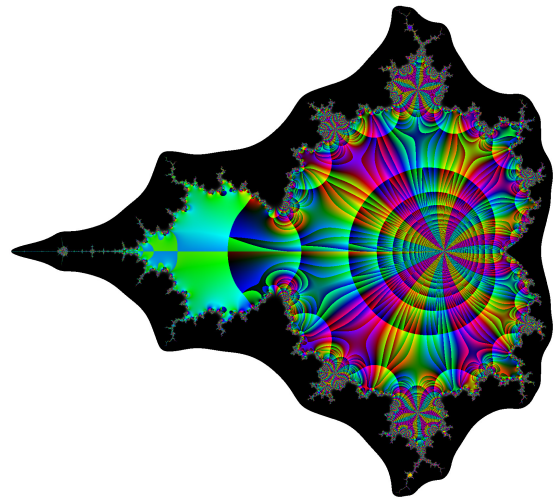
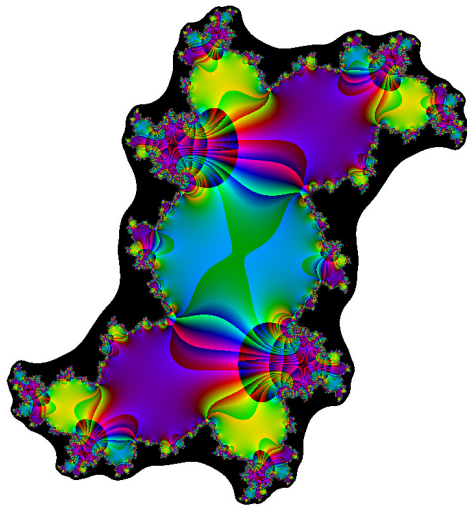
The optional argument `hsvrgb` must define three values which can use the arguments `phi` and `r` in any mathematical combination. It must only be compatible to the Lua math conventions, e.g. `hsvrgb={phi+2,0.5,2/r}`

2.6 External function definition

The already existing file `domaincoloring-functions.lua` collects some definitions of complex functions $f(z)$, which can be used from inside \LaTeX with the optional argument `funcName`. In this case the mandatory argument of `\DomainColoring` has no meaning and can be empty.

```
\DomainColoring[filename=\jobname-tmp6a,resolution=1000,
  domain={-1.5,1.5,-1.5,1.5}, grfOptions={width=0.49\linewidth},
  hsvrgb={phi+2*pi,1,1/r}, funcName=f12]{}% using predefined function
\hfill
\DomainColoring[filename=\jobname-tmp6b,resolution=2000,
  domain={-2.3,0.7,-1.5,1.5}, grfOptions={width=0.49\linewidth},
  hsvrgb={phi+2*pi,1,1/r}, funcName=f13]{}

```



The contents of the function file of the current version of domaincoloring is::

```
-- $Id: domaincoloring-functions.lua 946 2024-08-21 19:32:10Z herbert $
```

```
kpse.set_program_name("luatex")
```

```
function f0(z)
    return cmath.sin(1/z)*cmath.cos(1/z)/z^3+1/z
end
```

```
function f1(z)
    return cmath.cos(z)/cmath.sin(z^4-1)
end
```

```
function f2(z)
    local c = complex(1,-1)
    local d = complex(0,0.28)
    return cmath.cos(c^2*z^2)/cmath.cos(c*(z-d))
end
```

```
function f3(z)
    return z*(z+i)^2/(z-i)^2
end
```

```
function f4(z)
    if abs(z) < 0.1 then
        return complex(0.001,0.001)
    else
        return cmath.sin(1/(z*z))
    end
end
```

```
function f5(z)
    return cmath.sqrt(1-1/(z*z)+z^3)
end
```

```
function f9(z)
```

```
local c = complex(1,-1)
local d = complex(1,1)
return z^2*c^2*(z*c-1-i)/(z*c-2*d)
end

function f10(z)
local sum = complex(0,0)
for n=1,20 do
    sum = sum + z^n/(1-z^n)
end
return sum
end

function f11(z)
local iterateNo = 3
for n=1,iterateNo do
    z = z^2
end
return z
end

function f12(z) -- julia
local iterateNo = 15
for n=1,iterateNo do
    z = z^2 + complex(0.25,-0.5)
end
return z
end

function f13(z) -- mandelbrot
local iterateNo = 15
local c = z
z = complex(0,0)
for n=1,iterateNo do
    z = z^2 + c
end
return z
end
```

References

- [1] Konstantin Poelke and Konrad Polthier. *Domain Coloring of Complex Functions*. Aug. 18, 2024. URL: https://www.mi.fu-berlin.de/en/math/groups/ag-geom/publications/db/ieee_article_old_low_v3_1.pdf (visited on 08/18/2024).
- [2] vismath. *Thema Domain Coloring*. Aug. 18, 2024. URL: <https://www.vismath.eu/de/blog/domain-coloring/> (visited on 08/18/2024).
- [3] WIKIPEDIA. *Domain Coloring*. Aug. 18, 2024. URL: https://en.wikipedia.org/wiki/Domain_coloring (visited on 08/18/2024).