

mlmc_transport

April 13, 2020

\$ Notebook MLMC Transport \$

```
[1]: %matplotlib inline
import subprocess
import os
import sys
sys.path.append('../tools')
from plot_statistics import *
sys.path.append('../notebooks')
from tp_utilities import *
from vtk_utilities import *
import matplotlib.pyplot as plt
from starter import *
from mlmc_solution import *
sys.path.append('.')
from python.mlmc_mppy import mpp
```

```
[2]: mpp.build()
mpp.mute=True
kernels = 32
```

```
===== build sprng5 =====
```

```
-- libsprng.a found.
```

```
===== running cmake =====
```

```
-- Compiler option -Ofast
-- A library with BLAS API found.
-- A library with BLAS API found.
-- A library with LAPACK API found.
-- Using SuperLU 4.0
-- Time dependent problem
-- 3 dimensional problem
-- General transformations
-- Configuring done
-- Generating done
```

```
-- Build files have been written to: /home2/buchholz/mlmc/build
```

```
===== running make =====
```

```
[ 5%] Built target gtest
[ 5%] Built target LIB_PS
[16%] Built target gtest_main
[19%] Built target gmock
[31%] Built target MLMC
[49%] Built target gmock_main
[98%] Built target SRC
[100%] Built target MLMC-M++
```

```
$ Konvergenztest $
```

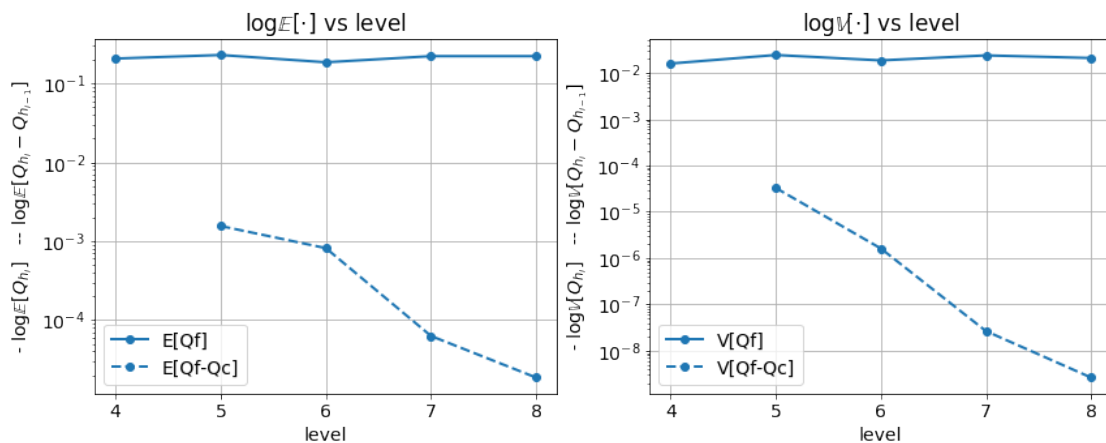
```
[3]: mpp.clean_data()
      mpp.run(kernels, config='mlmc_transport_ct')
      save("MLMCConvergenceTest/")
```

```
Created Folder
```

```
[4]: mpp.print_convergence_table()
```

	M	E[Qf-Qc]	E[Qf]	V[Qf-Qc]	V[Qf]	kurtosis	cost
1							
4	100	0.206865	0.206865	1.594270e-02	0.015943	2.49970	2.949120e+05
5	100	0.001558	0.229484	3.282100e-05	0.024385	4.00513	2.359300e+06
6	100	0.000817	0.185469	1.615470e-06	0.018631	3.99497	1.887440e+07
7	100	0.000063	0.221811	2.657520e-08	0.023902	2.75915	1.509950e+08
8	100	0.000019	0.221853	2.636770e-09	0.020987	3.37952	1.207960e+09

```
[5]: mpp.show_convergence_table()
```



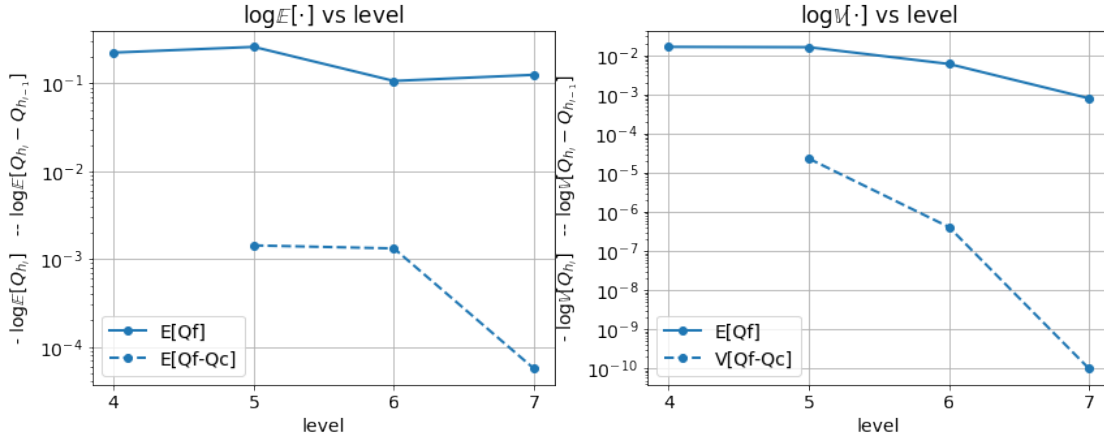
\$ MLMC Experiment epsilon = 0.01 \$

```
[6]: mpp.clean_data()
kwargs = {'epsilon': '0.01', 'initLevels': '4,5,6', 'initSampleAmount': '8,4,2'}
mpp.run(kernels, config='mlmc_transport', kwargs=kwargs)
save("MLMCExperiment/"+kwargs['epsilon']+"/")
```

```
[7]: statistics = read_log('../results/MLMCExperiment/0.01/log')
display(statistics)
```

	M	E[Qf-Qc]	E[Qf]	V[Qf-Qc]	V[Qf]	kurtosis	cost
1							
4	378	0.225017	0.225017	1.610760e-02	0.016108	2.991300	294912
5	6	0.001429	0.261486	2.279820e-05	0.015813	3.923450	2359300
6	2	0.001321	0.106957	4.096720e-07	0.005909	1.000000	18874400
7	2	0.000056	0.125627	1.000000e-10	0.000784	0.110661	150995000

```
[8]: plot_statistics(statistics, ['E[Qf]', 'V[Qf]'], None)
```



\$ MLMC Experiment epsilon = 0.005 \$

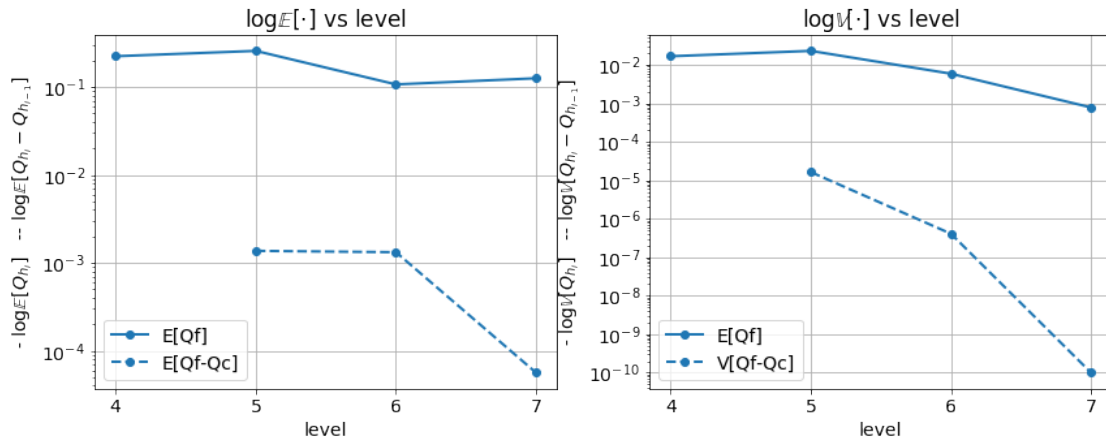
```
[9]: mpp.clean_data()
kwargs = {'epsilon': '0.005', 'initLevels': '4,5,6', 'initSampleAmount': '8,4,2'}
mpp.run(kernels, config='mlmc_transport', kwargs=kwargs)
save("MLMCExperiment/"+kwargs['epsilon']+"/")
```

```
[10]: statistics = read_log('../results/MLMCExperiment/0.005/log')
display(statistics)
```

	M	E[Qf-Qc]	E[Qf]	V[Qf-Qc]	V[Qf]	kurtosis	cost
1							
4	1539	0.223618	0.223618	1.678900e-02	0.016789	2.863540	294912

5	22	0.001368	0.257407	1.640940e-05	0.023179	3.119730	2359300
6	2	0.001321	0.106957	4.096720e-07	0.005909	1.000000	18874400
7	2	0.000056	0.125627	1.000000e-10	0.000784	0.110661	150995000

```
[11]: plot_statistics(statistics, ['E[Qf]', 'V[Qf]'], None)
```



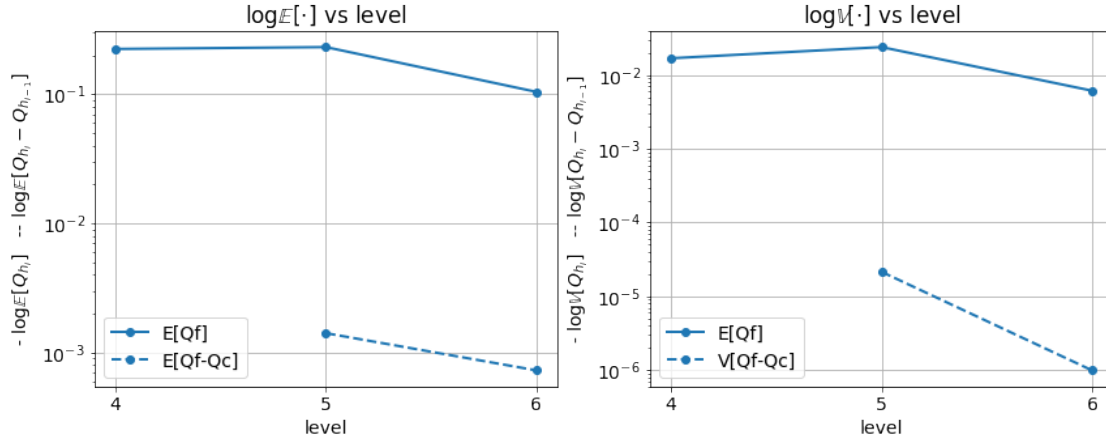
\$ MLMC Experiment epsilon = 0.003 \$

```
[12]: mpp.clean_data()
kwargs = {'epsilon': '0.003', 'initLevels': '4,5,6', 'initSampleAmount': '8,4,2'}
mpp.run(kernels, config='mlmc_transport', kwargs=kwargs)
save("MLMCExperiment/"+kwargs['epsilon']+"/")
```

```
[13]: statistics = read_log('../results/MLMCExperiment/0.003/log')
display(statistics)
```

	M	E[Qf-Qc]	E[Qf]	V[Qf-Qc]	V[Qf]	kurtosis	cost
1							
4	4374	0.224298	0.224298	1.693870e-02	0.016939	2.75002	294912
5	59	0.001411	0.231847	2.145810e-05	0.023991	2.59309	2359300
6	5	0.000727	0.104317	9.886560e-07	0.006146	1.66323	18874400

```
[14]: plot_statistics(statistics, ['E[Qf]', 'V[Qf]'], None)
```



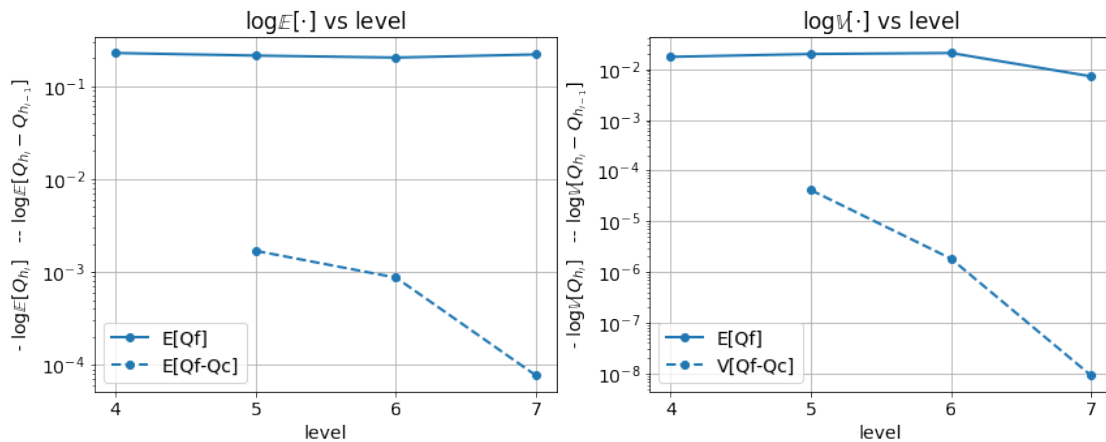
\$ MLMC Experiment epsilon = 0.001 \$

```
[15]: mpp.clean_data()
      kwargs = {'epsilon': '0.001', 'initLevels': '4,5,6', 'initSampleAmount': '8,4,2'}
      mpp.run(kernels, config='mlmc_transport', kwargs=kwargs)
      save("MLMCExperiment/"+kwargs['epsilon']+"/")
```

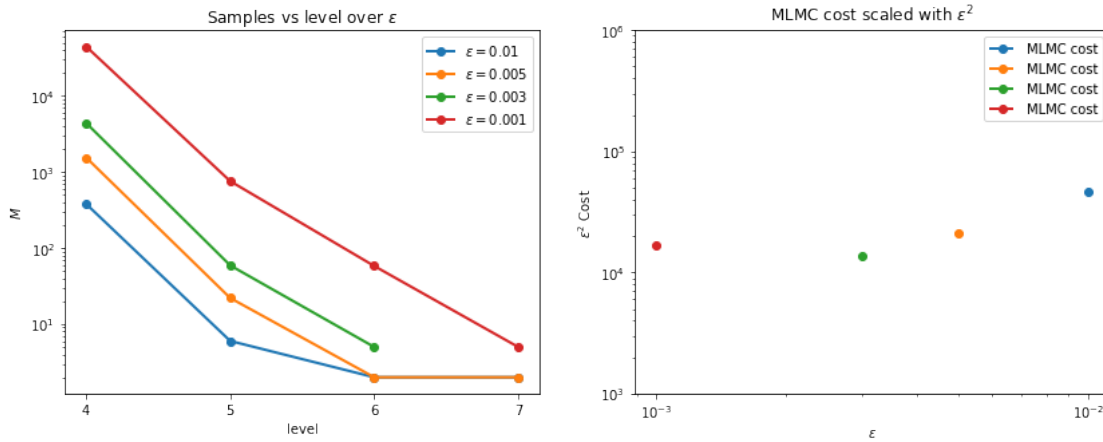
```
[16]: statistics = read_log('../results/MLMCExperiment/0.001/log')
      display(statistics)
```

	M	E[Qf-Qc]	E[Qf]	V[Qf-Qc]	V[Qf]	kurtosis	cost
1							
4	44254	0.226394	0.226394	1.743050e-02	0.017431	2.65924	294912
5	755	0.001687	0.212340	4.127960e-05	0.019814	3.68994	2359300
6	58	0.000869	0.202110	1.839350e-06	0.020755	3.80032	18874400
7	5	0.000077	0.218324	9.122700e-09	0.007189	2.45257	150995000

```
[17]: plot_statistics(statistics, ['E[Qf]', 'V[Qf]', None])
```



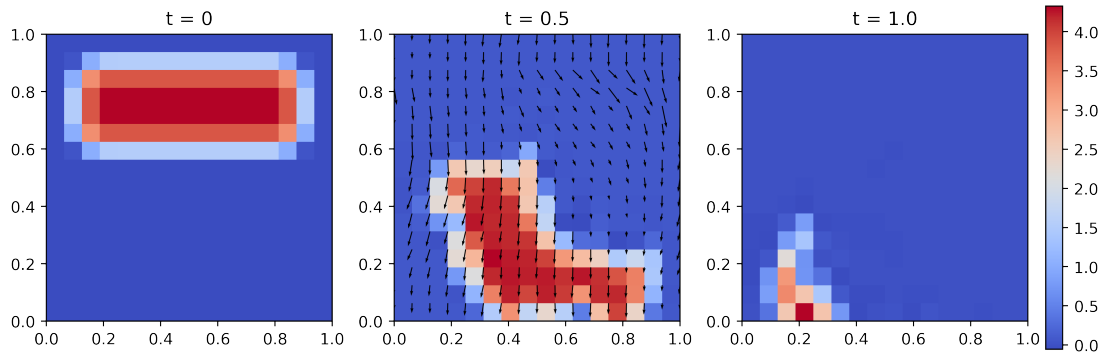
```
[18]: wd = '../results/MLMCEperiment/'
log_files = ['0.01/log', '0.005/log', '0.003/log', '0.001/log']
mpp.show_combined_mlmc_table(log_files= log_files, wd = wd )
```



\$ Beispielsamples\$

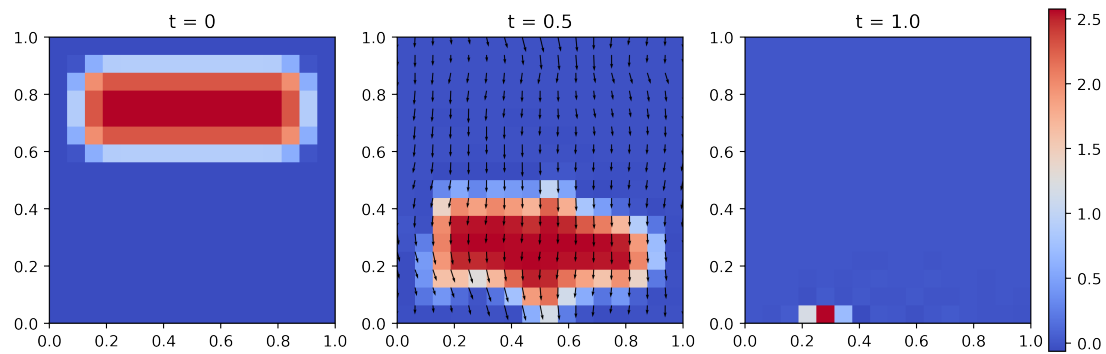
\$ Level 4 (baselevel) \$

```
[19]: solution_3(wd="../results/MLMCEperiment/0.001/vtk/",sample="sample_4_1/
↪",quiver_filter=1,quiver_scale=0.10)
```

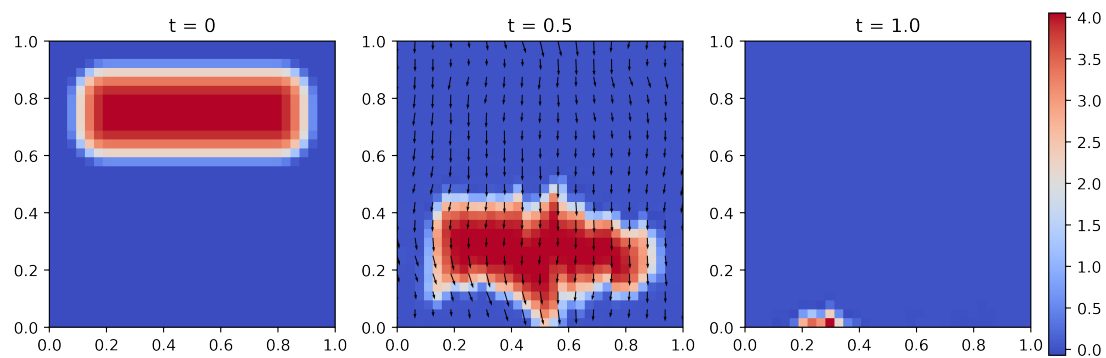


\$ Level 5 (und zugehöriges Vergleichssample auf Level 4) \$

```
[20]: solution_3(wd="../results/MLMCEperiment/0.001/vtk/",sample="sample_coarse_5_1/
↪",quiver_filter=1,quiver_scale=0.10)
```

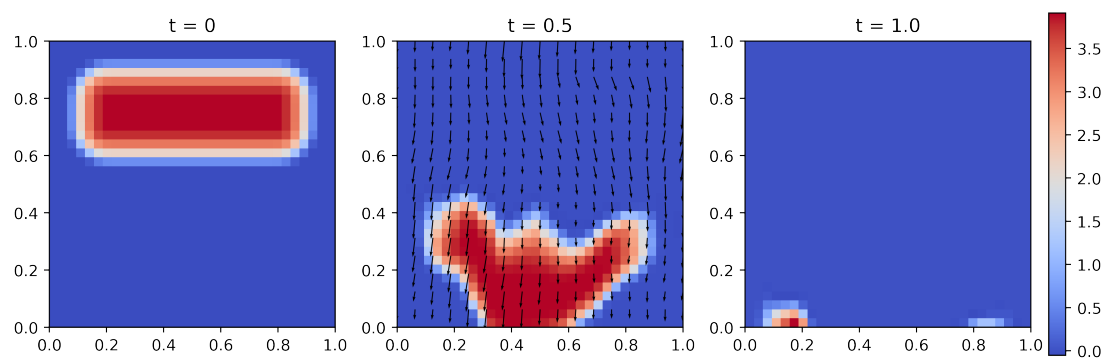


```
[21]: solution_3(wd="../results/MLMCEExperiment/0.001/vtk/",sample="sample_5_1/
↪",quiver_filter=2,quiver_scale=0.10)
```

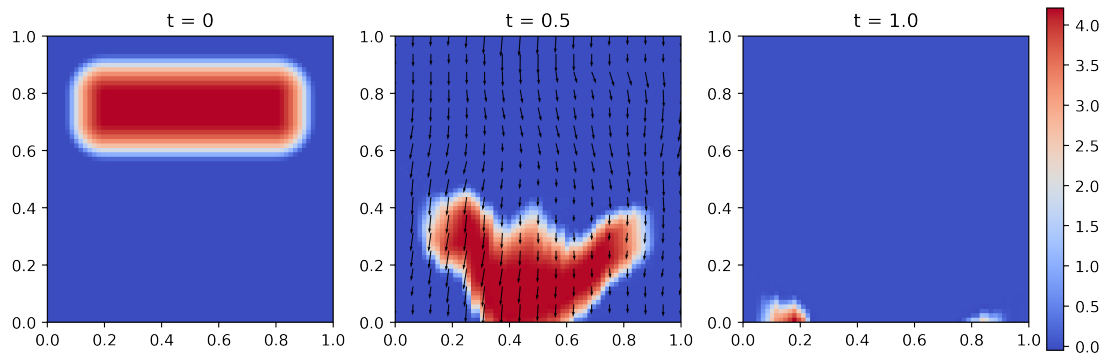


\$ Level 6 (und zugehöriges Vergleichssample auf Level 5) \$

```
[22]: solution_3(wd="../results/MLMCEExperiment/0.001/vtk/",sample="sample_coarse_6_0/
↪",quiver_filter=2,quiver_scale=0.12)
```

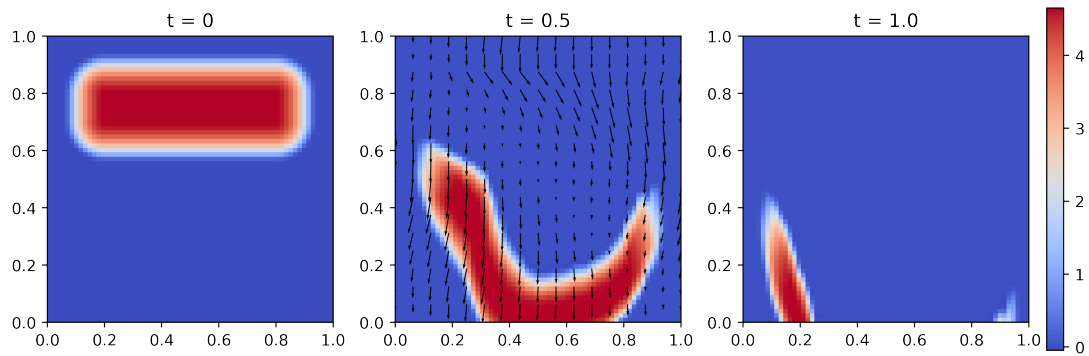


```
[23]: solution_3(wd="../results/MLMCEExperiment/0.001/vtk/",sample="sample_6_0/
↪",quiver_filter=4,quiver_scale=0.12)
```

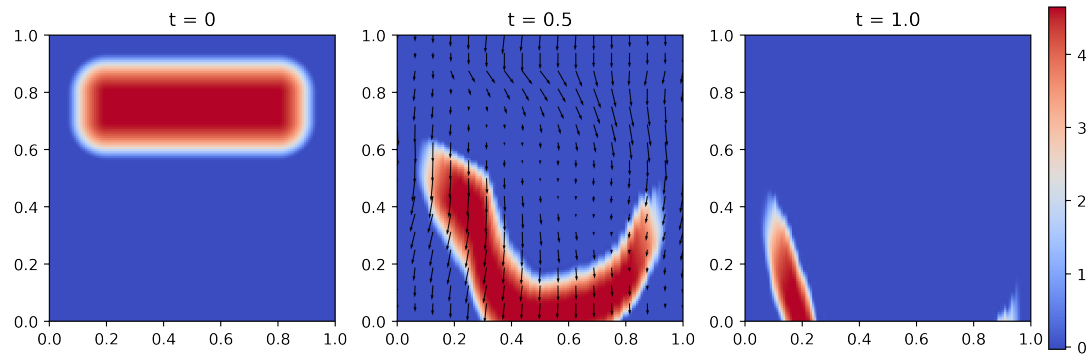


\$ Level 7 (und zugehöriges Vergleichssample auf Level 6) \$

```
[24]: solution_3(wd="../results/MLMCEExperiment/0.001/vtk/",sample="sample_coarse_7_0/
↪",quiver_filter=4,quiver_scale=0.12)
```



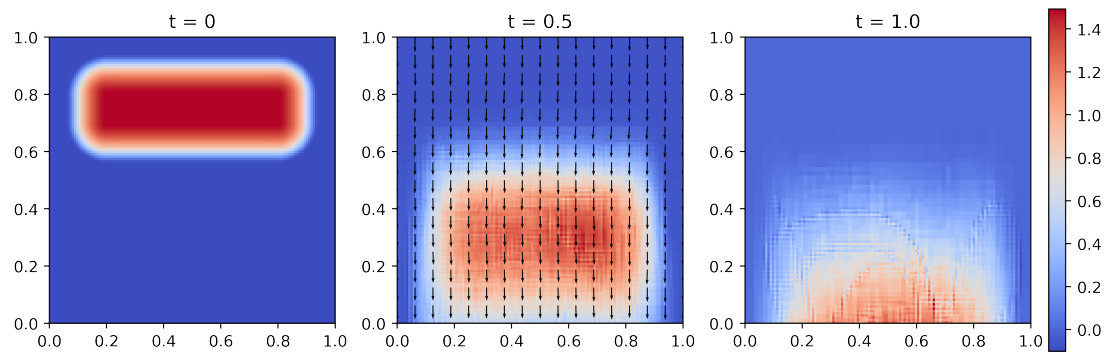
```
[25]: solution_3(wd="../results/MLMCEExperiment/0.001/vtk/",sample="sample_7_0/
↪",quiver_filter=8,quiver_scale=0.12)
```

\$ Lösung im MLMC Sinne basierend auf gegebenem Zielfunktional (hier Masse zum Zeitpunkt $t = 1.0$)\$

```
[26]: result = tail("../results/MLMCEperiment/0.001/log",n=22)
      levels, sample_amount = parse_nofsamples(result)
      mlmcmeshes(working_dir="../results/MLMCEperiment/0.001/vtk/"
      ↪,levels=levels,sample_amount=sample_amount)
```

```
[27]: solution_3(wd="../results/MLMCEperiment/0.001/vtk/",sample="mlmc/"
      ↪,quiver_filter=8,quiver_scale=0.12)
```



[28]: