

The mcPolymer command-line interface

The mcPolymer program is used in a terminal environment. In the Windows operating system this is the Command Prompt (cmd.exe or cmd) and at Linux any command shell (e.g. tcsh, bash). The following examples assume that the mcPolymer program (Windows: mcPolymer.exe, Linux: mcPolymer) and the model file "Styrene.mcPolymer" are in the current directory.

Calling the program with the **--help** option you can see all the features of the command line interface and a basic example of their application:

```
c:\tmp\mcPolymer>mcPolymer --help
Example:
  mcPolymer -m Styrene.mcPolymer --time=3600 --dt=60 --temperature=70.0 --molecules=10000000000
Usage:
  mcPolymer -m modelFile [OPTION...]

  -m, --model modelFILE      Model
  --help                     Print help
  --license                  Print license
  --time double              reaction time / s
  --dt double                time interval for output concentrations / s
  --dtCLD double             time interval for output cld | mmd / s
  --temperature double       temperature / centigrade
  --pressure double          pressure / bar
  --molecules unsigned int   num molecules
```

The option **--license** provides mcPolymer license status information. The mcPolymer simulator is published as open source software under the GNU general public license. On the web page www.itc.tu-clausthal.de/de/mcpolymer/download/ the source code of the program is available for download. The following external software components are integrated in mcPolymer and are part of the distribution:

- File `cxcopts.hpp` contains one C++ option parser library, supporting the standard GNU style syntax for options. This is a header only library and was developed by Jarryd Beck. Source: GitHub. github.com/jarro2783/cxcopts (accessed on 07.08.2018).
- File `MersenneTwister.h` implements the Mersenne Twister random number generator, developed by Makoto Matsumoto, Takuji Nishimura, Shawn Cokus and Richard J. Wagner. Source: GitHub. github.com/shhyang/simbats/blob/master/MersenneTwister.h (accessed on 07.08.2018).

Running a simulation requires a kinetic model. This is provided in a file which name is entered in the command line with the option **-m filename** or **--model filename**. In the example shown with the **--help** option, the model file `Styrene.mcPolymer` is used for the simulation (**-m Styrene.mcPolymer**).

The **--time** option determines the reaction time in seconds running the kinetic model. This time can optionally also be specified in the model file (e.g. `time=3600`). The entry on the command line overrules the value in the model file. The reaction time is required for the simulation has to be specified on at least one of those two places (command line: `--time=value`; model file: `time=value`). Otherwise mcPolymer terminates with an error message: "reaction time undefined". Generally for the pattern `--option=value` no spaces are allowed in front of or behind the "=", otherwise the interpretation on the command line will fail.

Equally essential is the input of the reaction temperature in degrees Celsius, which can also be done in the model file (`temperature=value`) or on the command line (**--temperature=value**). The same rules as for the reaction time will also be applied here. The entry of **--pressure=value** is optional; 1 bar is the default pressure.

For the kinetic Monte Carlo (kMC) simulation, the definition of the sample size is required. This is done by entering the statement **--molecules=value** on the command line. In our example `--molecules=10000000000`, the $1e10$ molecules are distributed to the educts according to the concentrations specified in the model file ($[\text{styrene}]_0 = 8.725 \text{ mol}\cdot\text{L}^{-1} \rightarrow 9977129788 \text{ molecules}$, $[\text{AIBN}]_0 = 0.02 \text{ mol}\cdot\text{L}^{-1} \rightarrow 22870212 \text{ molecules}$). This corresponds to a control volume of $1.9e-15 \text{ L}$. It is clear that the sample size in the kMC has a significant impact on the precision of the simulation but also on computation time and memory requirements. In this example

(-m Styrene.mcPolymer --time=3600 --temperature=70.0 --molecules=1000000000) the simulation takes about 3 minutes on an Intel Core i7-6700, 3.4 GHz CPU and requires a maximum of 530 MB of RAM.

The command line options `--dt=value` and `--dtCLD =value` control the logging of results. Both parameters have default values ($dt=1/60$ ·reaction time; dt =reaction time), which are used if these options are not set. After starting the program with the options mentioned in the example, the current parameters are displayed. In addition, an interval of dt (`--dt=60`) seconds is used to output the current reaction time in the running kMC simulation:

```
c:\tmp\mcPolymer>mcPolymer.exe -m Styrene.mcPolymer --time=3600 --dt=60 --temperature=70 --molecules=1000000000
model file: Styrene.mcPolymer
model dimension: 10000000000 ( 1.000000e+10 ) molecules
reaction time 3600 s
concentration output dt 60 s
cld | mmd output dtCLD 3600 s, set to reaction time by default
reaction temperature 70 degree Celsius
pressure 1 bar, set by default
time: 60
time: 120
time: 180
time: 240
```

mcPolymer output after starting the simulation

The concentration time and molecule number-time profiles are written in two files, which are updated at intervals of dt . The file names are derived from the name of the model file:

- *modelfilename.Concentrations.dat* (e.g. Styrene.mcPolymer.Concentrations.dat)
- *modelfilename.numMolecules.dat* (e.g. Styrene.mcPolymer.numMolecules.dat).

Both files are ASCII files in which the columns are separated by a tab character. This makes a data analysis or visualization in other programs easily possible. For each reaction time, the concentration ($\text{mol}\cdot\text{L}^{-1}$) or the number of molecules of each reactant contained in the kinetic model is available:

	A	B	C	D	E	F
1	time	styrene	AIBN	I	P	D
2	0	8.725	0.02	0	0	0
3	60	8.70611	0.0199537	0	8.13E-08	2.95E-05
4	120	8.68729	0.0199075	0	6.91E-08	5.91E-05
5	180	8.6685	0.0198613	8.75E-10	8.31E-08	8.86E-05
6	240	8.64981	0.0198152	0	7.70E-08	0.00011817

Styrene.mcPolymer.Concentrations.dat

	A	B	C	D	E	F
1	time	styrene	AIBN	I	P	D
2	0	9977129788	22870212	0	0	0
3	60	9955526425	22817296	0	93	33719
4	120	9934010153	22764382	0	79	67566
5	180	9912518701	22711604	1	95	101372
6	240	9891153744	22658910	0	88	135128

Styrene.mcPolymer.numMolecules.dat

Additionally, chain length distributions and molar mass distributions of all macromolecular species defined in the kinetic model are available as output files written at intervals specified by `--dtCLD`. The file name contains the name of the species and the reaction time. At the end of the simulation, two ASCII files are written e.g. for the dead polymer D defined in the model Styrene.mcPolymer: chain length distribution → D.3600.000000.CLD.dat and molar mass distribution → D.3600.000000.CLD.dat.WlogM.dat. The molar mass distribution is stored in equidistant X-axis steps ($\log(M)$) and normalized to area 1. The file with the chain length distribution has the following structure:

	A	B	C	D	E	F
1	numMonomers:	1				
2	styrene:	104.15				
3	Mn:	64543.1	Mw:	96980.8	Mw/Mn:	1.50258
4	numChains:	1896911	numSCB:	0	numLCB:	0
5	Pn:	619.713	Pw:	931.165	Pw/Pn:	1.50258
6	chainLength	numChains	numSCB	numLCB	styrene	
7	2	27	0	0	54	
8	3	52	0	0	156	
9	4	94	0	0	376	
10	5	101	0	0	505	
11	6	134	0	0	804	
12	7	141	0	0	987	
13	8	151	0	0	1208	
14	9	197	0	0	1773	
15	10	217	0	0	2170	
16	11	217	0	0	2387	

3671	4894	1	0	0	4894	
3672	4951	1	0	0	4951	
3673	5095	1	0	0	5095	

D.3600.000000.CLD.dat

In the file header, the mean values and the number of polymer molecules are summarized. The chain length distribution begins in this example at line 6 and is located in columns A (chain length) and B (number of chains). In addition, each chain length is assigned the number of short chain branches (column C, numSCB) and long chain branches (column D, numLCB). In this simple example Styrol.mcPolymer, no branching reactions occur. In column E, the number of monomer units in the chains of the corresponding chain length is given. In the styrene homopolymerization, only column E is occupied. In a copolymerization, the number of these monomers is located in additional columns, starting with F.

Tutorial: example model file Styrene.mcPolymer

```

1 M(styrene)=104.15
2 // styrene bulk polymerization
3 [styrene] = 8.725
4
5 [AIBN]=0.02
6 f(AIBN)=0.64
7
8 kiAIBN=4.31e15*exp(-131.7/RT)
9 kt=1.341e10 * exp(-14.34/RT)
10 kp=4.2658e+07* exp(-32.51/RT)
11
12 temperature = 70.0
13
14 // styrene polymerization
15 AIBN --kiAIBN--> I + I
16 I + styrene --kp--> P(n)
17 P(n) + styrene --kp--> P(n+1)
18 P(n) + styrene --km--> D(n) + P(1)
19 P(n) + P(m) --kt--> D(n+m)
20
21 time=3600

```

Model file Styrene.mcPolymer, displayed with line numbers.

```
1 M(styrene)=104.15
```

The molecular weight 104.15 g·mol⁻¹ is assigned to the monomer styrene. This statement implicitly defines the species styrene as a monomer.

```
2 // styrene bulk polymerization
```

This line is a comment. Comments are initiated with the string "//".

```
3 [styrene] = 8.725
```

The styrene concentration is set to 0.02 mol·L⁻¹.

```
4
```

Blank lines in the command file improve readability and are not interpreted.

```
5 [AIBN]=0.02
```

It implicitly defines a low molecular weight species called "AIBN". The concentration is set to 0.02 mol·L⁻¹ for the beginning of the reaction.

```
6 f(AIBN)=0.64
```

This statement implicitly defines the low molecular weight species AIBN as initiator. The initiator efficiency is 0.64 and can be chosen in the range of 0 to 1.

```
8 kiAIBN=4.31e15*exp(-131.7/RT)
```

It defines a reaction rate coefficient with the pre-exponential factor 4.31·10¹⁵ s⁻¹ and the activation energy 131.7 kJ·mol⁻¹.

```
9 kt=1.341e10 * exp(-14.34/RT)
```

It defines a reaction rate coefficient with the pre-exponential factor 1.341·10¹⁰ L·mol⁻¹·s⁻¹ and the activation energy 14.34 kJ·mol⁻¹. The spaces (e.g., before and after "*") generally do not affect the interpretation of the instruction.

```
12 temperature = 70.0
```

The temperature is set to 70 °C. If such an instruction is included in the model file, the temperature at the start of the program need not be entered on the command line. With an optional input on the command line (e.g., --Temperature = 60.0), this value is overruled and the simulation runs at 60°C.

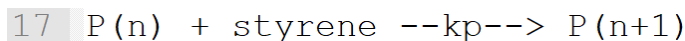
```
15 AIBN --kiAIBN--> I + I
```

The initiator decomposition of AIBN becomes part of the kinetic model. The reaction is derived from the reaction template "initiator --k --> I + I" (see Table Reaction Templates). AIBN stands for the initiator, I for the primary radicals and kiAIBN for the reaction rate coefficient. The reaction template "initiator --k --> I + I" is only recognized correctly if the initiator has been assigned an initiator efficiency (see line 6). The primary radicals I are implicitly defined as low molecular weight species when used in a reaction. A separate definition (e.g., [I] = 0.0) is not necessary.

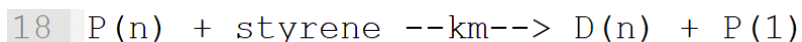
```
16 I + styrene --kp--> P(1)
```

This reaction describes the initiation of a macro radical from the primary radical I with the monomer styrene. The concrete reaction was derived from Template "I + M --k--> P(1)" (see Table Reaction Templates). This

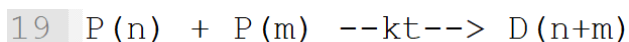
template expects a monomer at position "M". Therefore, it is essential that the species styrene is assigned a molecular weight (see line 1). Only with this assignment the species styrene does get the property "monomer" and can be used in this reaction. The result is a macro radical P(n) with the current chain length $n = 1$. In formulating the reaction, the macroradical must be labeled as a macromolecule. In this initiation reaction, this is done by the name suffix (1).



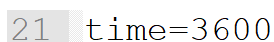
The propagation reaction formulated on this line is derived from Template $P(n) + M \xrightarrow{-k-} P(n+1)$. Again, a monomer is expected at position "M". While the name of the macroradical is selectable by the user (e.g., P or MR), the extensions (n) in the educt and (n + 1) in the product should be used exactly as shown.



In this transfer reaction with the monomer styrene, a dead polymer D(n) and a new macroradical P with chain length 1 are formed. The associated reaction template is: $P(n) + M \xrightarrow{-k-} D(n) + Q(1)$. It must be ensured that at the position "M" is a monomer and the macromolecular species with (n), (n) and (1) are marked exactly at the position shown.



In this termination by combination, a dead polymer D with the chain length $n + m$ is formed. The rate coefficient k_t is applied according to the IUPAC conventions. The label of the macromolecular species must be used as given in the reaction template " $P(n) + P(m) \xrightarrow{-k-} D(n+m)$ ".



The simulation ends with a reaction time of 3600 seconds. This default can be changed by an input on the command line at the start of the program, e.g. `--time=7200`.

Reaction templates in mcPolymer

Table: Reaction templates: M, M1, M2 – monomer, P(n), Q(m) – macromolecular species, X, A, B – low molecular species, st – monomer styrene, SCB – short chain branch, TDB – terminal double bond, k – reaction rate coefficient

template	description	example	model file (.mcPolymer)
initiator --k--> I + I	initiator decomposition	AIBN --kiAIBN--> I + I	Styrene
I + M --k--> P(1)	initiation	I + styrene --kp--> P(n)	Styrene
st + st + st --k--> P(1) + P(1)	thermal initiation (styrene)	styrene + styrene + styrene --kth--> P(1) + P(1)	ATRP
P(n) + M --k--> P(n+1)	propagation (homo- or copolymerization)	P(n) + styrene --kp--> P(n+1)	Styrene
P(n) + M --k--> Q(n+1)		PM1M1(n) + styrene --kp112--> PM1M2(n+1)	MMA-S-SG1
P(n) + M --k--> Q(n+1,SCB)	propagation + SCB	MCR(n) + BA --kpt--> SPR(n,SCB)	BA
P(n) + P(m) --k--> D(n+m)	termination by combination	P(n) + P(m) --kt--> D(n+m)	Styrene
P(n) + Q(m) --k--> D(n+m)		SPR(n) + MCR(m) --ktsm--> D(n+m)	BA
P(n) + P(m) --k--> D(n) + D(m,TDB)	termination by disproportionation	PM1M1(n) + PM1M1(m) --ktd11--> D(n) + D(m,TDB)	MMA-S-SG1
P(n) + Q(m) --k--> D(n) + D(m,TDB)		PM1M1(n) + PM2M2(m) --ktd12--> D(n) + D(m,TDB)	
P(n) --k--> Q(n)	transfer	SPR(n) --kbb--> MCR(n)	BA
P(n) + X --k--> Q(n)		P(n) + N --kc--> PN(n)	NMP
P(n) --k--> Q(n) + X		PN(n) --kd--> P(n) + N	NMP
P(n) + M --k--> D(n) + Q(1)	transfer to monomer	P(n) + styrene --km--> D(n) + P(1)	Styrene
P(n) + D(m, M1) --k--> D(n) + MCRL(m, M2)	transfer to polymer	SPR(n) + D(m,BA) --kmp--> D(n) + MCRL(m,BAX)	BA
P(n) --k--> Q(n-s) + D(s)	fragmentation	PSCSP(n) --kfrac--> P(n-m) + PSCS(m)	RAFT
A --k--> B + C	elementary reaction	RN --kd--> R + N	NMP
A + B --k--> C	elementary reaction	R + N --kc--> RN	NMP