

Article

Simulating Controlled Radical Polymerizations with mcPolymer—A Monte Carlo Approach

Marco Drache ^{1,*} and Georg Drache ²

¹ Institute of Technical Chemistry, Clausthal University of Technology, Erzstr. 18, Clausthal-Zellerfeld 38678, Germany

² Faculty of Computer Science, TU Dresden, Nöthnitzer Straße 46, Dresden 01187, Germany; E-Mail: georg@drache.cc

* Author to whom correspondence should be addressed; E-Mail: marco.drache@tu-clausthal.de; Tel.: +49-5323-722048; Fax: +49-5323-723655.

Received: 25 June 2012; in revised form: 17 July 2012 / Accepted: 18 July 2012 /

Published: 30 July 2012

Abstract: Utilizing model calculations may lead to a better understanding of the complex kinetics of the controlled radical polymerization. We developed a universal simulation tool (mcPolymer), which is based on the widely used Monte Carlo simulation technique. This article focuses on the software architecture of the program, including its data management and optimization approaches. We were able to simulate polymer chains as individual objects, allowing us to gain more detailed microstructural information of the polymeric products. For all given examples of controlled radical polymerization (nitroxide mediated radical polymerization (NMRP) homo- and copolymerization, atom transfer radical polymerization (ATRP), reversible addition fragmentation chain transfer polymerization (RAFT)), we present detailed performance analyses demonstrating the influence of the system size, concentrations of reactants, and the peculiarities of data. Different possibilities were exemplarily illustrated for finding an adequate balance between precision, memory consumption, and computation time of the simulation. Due to its flexible software architecture, the application of mcPolymer is not limited to the controlled radical polymerization, but can be adjusted in a straightforward manner to further polymerization models.

Keywords: Monte Carlo simulation; controlled/living radical polymerization; NMRP; ATRP; RAFT; copolymerization

1. Introduction

The controlled radical polymerization includes a couple of polymerization techniques which are suitable for synthesizing polymers of well-defined architectures that have narrow molecular weight distributions. The three most mentionable representatives are the Reversible Addition Fragmentation Chain Transfer (RAFT) polymerization [1], the Atom Transfer Radical Polymerizations (ATRP) [2], and the Nitroxide Mediated Living Free Radical Polymerizations (NMP) [3]. For validating the underlying complex reaction mechanisms, modeling and simulation have proved to be valuable tools.

Two concepts are common for simulating kinetic models. The first is the mathematic-numeric approach based on a system of differential equations. One example is the method of moments. Recent applications in the field of controlled radical polymerization are the models of D'hooge *et al.* [4] (ATRP) and Saldivar-Guerra *et al.* [5,6] (RAFT, NMP). Another example for this solution is the popular commercial software PREDICI[®] by CiT (Computing in Technology, GmbH). Implementing the discrete Galerkin h-p-method, calculation of the full molecular weight distribution is possible [7,8]. Each previous specified type of the controlled radical polymerization (RAFT [9,10], ATRP [11], NMP [12]) has been simulated with PREDICI up to complex models of copolymerization [13].

The second approach, is the application of Monte Carlo (MC) methods. Simulating polymer chains as individual objects allows gaining more detailed microstructural information about the polymeric products. In contrast to the differential method, MC simulations have a defined number of objects used in the model. If a number is chosen which is too small, the simulation will lead to incorrect results, while a large model will increase the computation time and memory consumption. The ATRP polymerization [14], optionally with a bifunctional ATRP initiator [15], ATRP copolymerization [16], RAFT polymerization [17,18], NMP homopolymerization [19], NMP copolymerization [20], and gradient copolymerization with tracking the sequence distribution [21,22] are examples of the application of the Monte Carlo simulation.

This publication introduces an implementation of a universal Monte Carlo simulator 'mcPolymer' (open source software under the GNU General Public License [23]) which can handle versatile problems concerning the controlled radical polymerization (NMP, ATRP, RAFT). A first version of this tool was already used for simulating some kinetic models of the controlled radical polymerization based on experimental data [17].

In the following, the application of mcPolymer is demonstrated through various examples while validating the MC simulation, comparing the results with equivalent PREDICI models. Thereby, different technical aspects in running a Monte Carlo simulation are inspected, such as the influence of the simulation parameters, data structures of the polymeric reactants, and the computer architecture on the preciseness and performance of the simulation. Furthermore, we discuss the application of mcPolymer in parallel environments using the example of the RAFT polymerization and the modeling of the nitroxide controlled copolymerization.

2. Simulating Chemical Kinetics with Monte Carlo Methods

The Monte Carlo Simulation of chemical kinetics is based on the algorithm proposed by Gillespie [24]. The particles X_i of the species S_i ($i = 1, 2, \dots, N$) are located in a homogenous reaction

volume. The species S_i represent the reactants of the M reactions R_μ ($\mu = 1, 2, \dots, M$) of the kinetic model. According to the reaction probabilities p_ν , the next reaction to be executed is randomly determined (Equation 1). The reaction probabilities are proportional to the reaction rates (Equation 2). Since the reaction rates are based on a discrete number of particles X_i in the reaction volume, the macroscopic deterministic rate constants k^{exp} have to be transformed into microscopic stochastic rate constants k^{MC} (Equations 3–5) using the system volume V and the Avogadro's constant N_A . Therefore, the stochastic reaction rates R_ν can be calculated from the number of particles X_i (Equations 6–8). The time interval τ between two sequentially executed reactions is determined by Equation 9 utilizing a second random number r_2 . Due to the varying number of particles during the simulation, the reaction probabilities have to be updated after every reaction step, which also affects the time interval τ (Equation 9).

$$\sum_{\nu=1}^{\mu-1} p_\nu < r_1 \leq \sum_{\nu=1}^{\mu} p_\nu \quad (1)$$

$$p_\nu = \frac{R_\nu}{\sum_{\nu=1}^M R_\nu} \quad (2)$$

First order reaction:

$$k^{MC} = k^{\text{exp}} \quad (3)$$

Bimolecular reaction, different reacting species:

$$k^{MC} = \frac{k^{\text{exp}}}{VN_A} \quad (4)$$

Bimolecular reaction, identical reacting species:

$$k^{MC} = \frac{2k^{\text{exp}}}{VN_A} \quad (5)$$

First order reaction:

$$R_\nu = k_\nu^{MC} X_i \quad (6)$$

Bimolecular reaction, different reacting species:

$$R_\nu = k_\nu^{MC} X_i X_j \quad (7)$$

Bimolecular reaction, identical reacting species:

$$R_\nu = k_\nu^{MC} X_i (X_i - 1) \quad (8)$$

$$\tau = \frac{1}{\sum_{\nu=1}^M R_\nu} \ln\left(\frac{1}{r_2}\right) \quad (9)$$

The Monte Carlo simulation is processed in the following steps:

1. Initialization

First, the overall number of particles n_{SX} of the simulation is defined. Those particles are distributed to the species S_i proportional to the given concentrations (Equation 10). The reaction volume V can be calculated from the overall number of particles and the sum of concentrations $[S_i]_0$ (Equation 11). Afterwards, the stochastic rate constants k^{MC} can be determined with the Equations 3–5. The simulation time t is set to 0.0. Additionally, the reaction time of the simulation has to be defined as well as a specific time interval dt for recording intermediate results. This time interval dt has to be an integer divider of the reaction time.

$$X_i = \frac{[S_i]_0}{\sum_{i=1}^N [S_i]_0} n_{SX} \quad (10)$$

$$V = \frac{n_{SX}}{N_A \cdot \sum_{i=1}^N [S_i]_0} \quad (11)$$

2. Update of the reaction rates R_v and probabilities p_v (Equations 2, 6–8)

3. Generation of a random number r_1 and choice of a reaction (Equation 1)

In mcPolymer, the choice of the reaction is implemented using a linear scan through the reaction possibilities. For the kinetic model, we recommend formulating the reaction with the highest reaction probabilities first to achieve a good hit rate. Barner-Kowollik *et al.* [18] proposed an improved algorithm that reduces the complexity of reaction choice by using a special data structure (binary tree). This algorithm might have advantages when a large number of reactions in the kinetic model are included, which have widely scattered reaction probabilities.

4. Execution of the reaction

While performing the reaction, the particle numbers of the reactants X_i are changed by consuming the educts and creating the products. Other reactions may contain reactants of the executed reaction. Therefore, all related reaction rates R_v are marked for update (processed in step 2).

5. Calculation of the time interval τ (Equation 9) and update of the simulation time (Equation 12)

$$t = t + \tau \quad (12)$$

6. Writing results

After specified intervals, the simulation results are recorded. Optionally, we can determine derived parameters (e.g., the conversion of specific reactant) from the intermediate results and update reaction rate constants during the simulation. Therefore, simulating diffusion controlled reactions is feasible.

Steps 2 to 5 represent one Monte Carlo (MC) step. The implementation has to be very efficient and the random numbers (r_1, r_2) need to be of high quality for exact results. For our simulator, we utilize the C++ Mersenne Twister random number generator by Wagner based on the algorithm by

Matsumoto *et al.* [25]. Since we utilize an enormous number of high-quality random numbers for our simulation, we need a fast generator with a long period that matches these criteria.

3. Implementation of Polymerization Kinetics

3.1. Macromolecular Species

Kinetic models of radical polymerization contain low molecular weight species and macromolecules with different chain lengths. Furthermore, the macromolecules occur in complex architectures.

The simplest implementation is to store the macromolecules in a chain length distribution histogram. If the polymerization model contains linear homopolymers only, this alternative is very memory efficient. We only require a field, in which the number of chains is stored using the index as mapping reference to the chain length. One disadvantage is the time-consuming choice of a discrete macromolecule for executing a reaction.

A second alternative is storing the macromolecules in a vector of pointers to individual macromolecular objects with their own properties. This needs much more memory space, but offers a lot of advantages due to the individual characteristics of the discrete macromolecules. In these objects, complex molecular architectures (e.g., star polymers) can be stored, which can occur when modeling the controlled radical polymerization. Another example is storing the individual composition of each polymer chain in a copolymerization model. The implemented macromolecular species are summarized in Table 1. Another advantage of this mode comes from the much faster choice of the macromolecules as educts while executing the reactions.

Table 1. Molecular species.

Name	Mode of storage	Description
Species	–	any low molecular species
Monomer	–	Monomer
Initiator	–	Initiator, additional property: initiator efficiency $f(0.0 < f \leq 1.0)$
SpeciesMacroHomopolymer	histogram	homopolymer
SpeciesMacroD	histogram	dead homopolymer product in termination reactions
SpeciesMacro	vector	homopolymer object properties: chain length
SpeciesMacro_P-O-P	vector	homopolymer consisting of two polymer chains object properties: chain length of each chain
SpeciesMacro_P-O<PP	vector	homopolymer consisting of three polymer chains (star polymer) object properties: chain length of each chain
SpeciesMacroCopolymer	vector	copolymer object properties: number of comonomer units
SpeciesMacroCopolymerD	vector	dead copolymer object properties: number of comonomer units product in termination reactions

3.2. Reaction Templates

Beside elemental reactions, a set of templates for reactions with macromolecular reactants were provided. The reaction templates are summarized in Table 2. In the reaction templates, macromolecular species are both stored in the vector and the histogram mode. Occasionally, both modes are used, e.g., when generating dead polymer in a termination reaction.

Table 2. Reaction templates.

Name	Scheme	Description
Elemental_1E2P	$A \longrightarrow B + C$	Examples of elemental reactions with low molecular species (nomenclature: xE educts, yP products)
Elemental_2E1P	$A + B \longrightarrow C$	
InitiatorDecomposition	$I \longrightarrow 2f I^\bullet$	Initiator decomposition; the stoichiometry depends on the initiator efficiency f
ThermalInitiation	$3 \text{ Styrene} \longrightarrow 2 P_1^\bullet$	Hamielec model of thermal initiation [26]
Initiation	$I^\bullet + M \longrightarrow P_1^\bullet$	Initiation of new macromolecule with chain length 1
Propagation	$P_n^\bullet + M \longrightarrow P_{n+1}^\bullet$	Propagation reaction (homopolymerization)
Transfer2monomer	$P_n^\bullet + M \longrightarrow D_n + P_1^\bullet$	Radical transfer to monomer
Termination	$P_n^\bullet + P_m^\bullet \longrightarrow D_{n+m}$	Termination by combination
	$P_n^\bullet + Q_m^\bullet \longrightarrow D_{n+m}$	
Transfer_PL-P	$P_n^\bullet + A \longrightarrow Q_n$	Transfer reactions of macromolecules (P) with low molecular species (L)
Transfer_P-PL	$Q_n \longrightarrow P_n^\bullet + A$	
Transfer_PL-PL	$P_n^\bullet + A \longrightarrow Q_n + B$	
Transfer_PL-PL	$Q_n + B \longrightarrow P_n^\bullet + A$	
ChainAddition	$P_n^\bullet + Q_m \longrightarrow R_{c_n-o-c_m}$	Addition of polymer chains; the individual chain lengths (n, m, k) are preserved in the new macromolecule
	$P_n^\bullet + Q_{c_k-o-c_l} \longrightarrow R_{c_k-o-c_l c_n}$	
ChainFragmentation	$R_{c_n-o-c_m} \longrightarrow Q_m + P_n^\bullet$	Fragmentation into embodied chains (random choice)

The controlled radical polymerization involves a reversible deactivation of propagating polymer chains. This can be modeled using the reaction templates Transfer_PL-PL (ATRP) or Transfer_PL-P and Transfer_P-PL (NMP). For modeling the RAFT polymerization, both an addition and a fragmentation reaction is needed. Table 2 contains a suitable fragmentation template (ChainFragmentation) with a polymer species as educt typed SpeciesMacro_P-o-P, which requires an additional selection step for the specific chain to be transferred. Copolymerization models need templates for recording polymer chain compositions (Table 3). Penultimate effects can optionally be simulated by using the provided reaction templates. The appendices contain the concrete application of

the reaction templates for simulating controlled radical polymerization (NMP: Appxs 1, 2; ATRP: Appx 3; RAFT: Appx 4, NMP-Copolymerization: Appx 5).

The C++ STL offers highly efficient vector classes for storing macromolecular species. The vector only contains pointers to the potentially very complex objects. In the aforementioned examples, a macromolecular object has to be transferred from the vector of the educt species to the vector of a product species. Thereby, a macromolecular object is chosen using a random number. Since the arrangement of the object pointers is not relevant for our model, we only perform insert and delete operations at the tail of the vectors. This minimizes the administration overhead of the vector class and gives a measurable performance increase.

Table 3. Reaction templates for copolymerization models.

Name	Scheme	Description
InitiationCopolymer11	$RM_1^\bullet + M_1 \longrightarrow PM_1M_1^\bullet$	Initiation of copolymers with two monomer units for simulating penultimate effects
InitiationCopolymer12	$RM_1^\bullet + M_2 \longrightarrow PM_1M_2^\bullet$	
InitiationCopolymer21	$RM_2^\bullet + M_1 \longrightarrow PM_2M_1^\bullet$	
InitiationCopolymer22	$RM_2^\bullet + M_2 \longrightarrow PM_2M_2^\bullet$	
PropagationCopolymer1	$P_n^\bullet + M_1 \longrightarrow P_{n+1}^\bullet$	Adding a monomer type 1 or 2 to the copolymer
PropagationCopolymer2	$P_n^\bullet + M_2 \longrightarrow P_{n+1}^\bullet$	
TransferCopolymer_PL-P	$P_n^\bullet + A \longrightarrow Q_n$	Transfer reactions of macromolecules (P) with low molecular species (L)
TransferCopolymer_P-PL	$Q_n \longrightarrow P_n^\bullet + A$	
TerminationCombCopolymer	$P_n^\bullet + P_m^\bullet \longrightarrow D_{n+m}$	Termination by combination and disproportionation
	$P_n^\bullet + Q_m^\bullet \longrightarrow D_{n+m}$	
TerminationDispCopolymer	$P_n^\bullet + P_m^\bullet \longrightarrow D_n + D_m$	
	$P_n^\bullet + Q_m^\bullet \longrightarrow D_n + D_m$	

4. Technical Aspects of mcPolymer

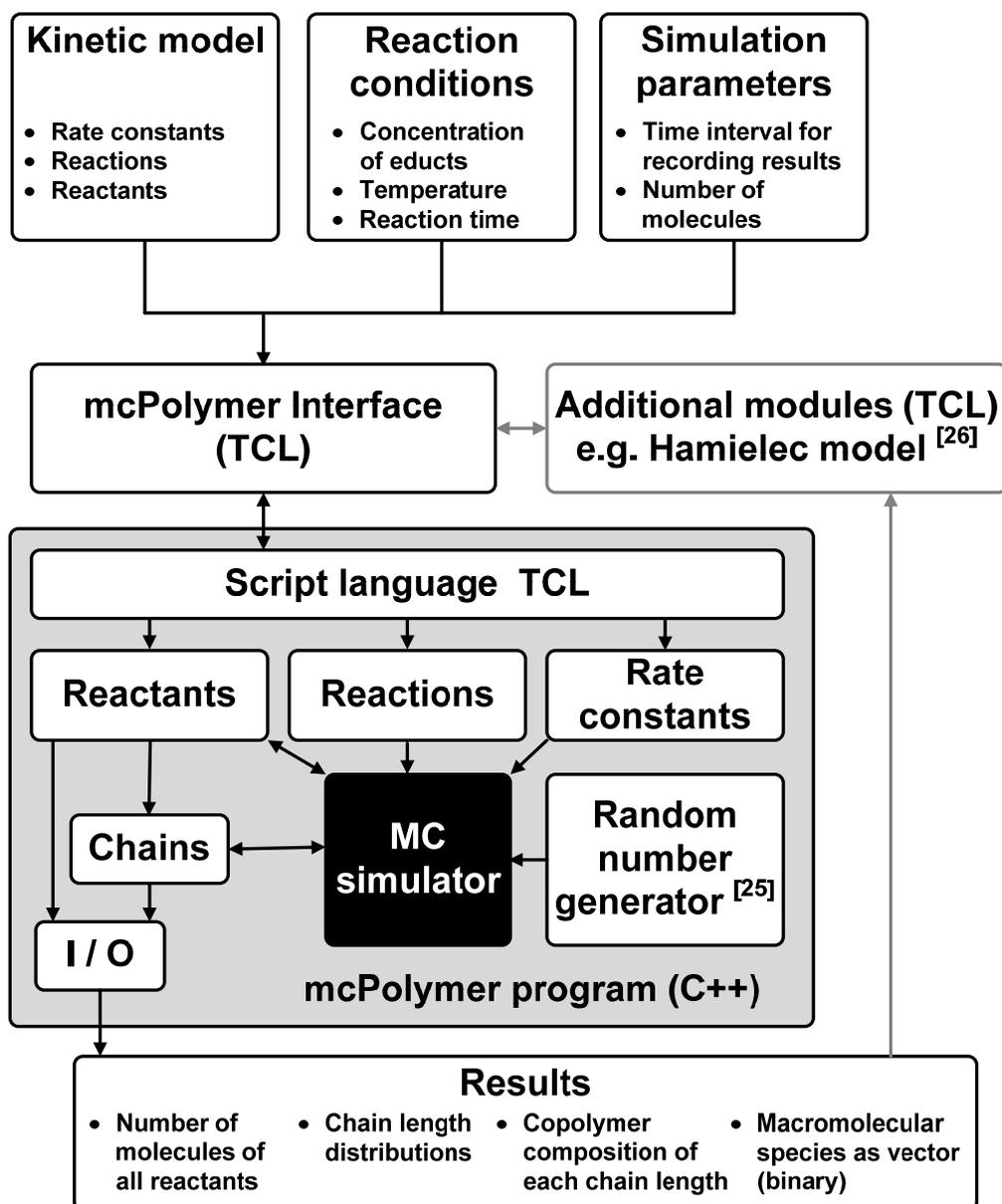
Apart from achieving a good performance of the MC simulation, another main goal of the implementation was a maximal flexibility in formulating the kinetic models and the reaction conditions. Therefore, the kinetic model was not implemented in the C++ core of the simulator, but rather as a script that is interpreted by the system. Utilizing the script language TCL, the kinetic model, the reaction conditions and the simulation parameters can be described in a flexible and readable form. The interpretation and test of plausibility is carried out by the mcPolymer Interface (TCL script). This interface initializes all objects in mcPolymer and then runs the simulation. A time interval parameter dt for writing intermediate results is set at the initialization of the simulation. Periodically completing dt, mcPolymer freezes the simulation in order to export the results as ASCII and binary files.

At this point, the interface is able to analyze the results (e.g., the conversion of educts) and modify derived reaction rate constants. An application is a periodic update of the reaction rate constant k_t of the termination reaction as function of the monomer conversion for simulating the gel effect. The

required calculations are not time-critical and can be provided as additional modules in the interface. For instance, it would be possible to calculate chain-length dependent rate coefficients periodically and apply them in the simulation.

Using the TCL libraries, mcPolymer itself is a TCL shell that can run TCL scripts including the mcPolymerInterface file. The interaction between the specific program modules is illustrated in Figure 1.

Figure 1. Software architecture of the mcPolymer simulator.



The program mcPolymer can be initialized using intermediate results. The macromolecular species are imported from the binary files. This feature is essential in applying mcPolymer in parallel environments.

The simulation program can run on different platforms: from workstation computers with Microsoft Windows up to compute servers running UNIX (Linux). Utilizing the script language TCL for the interfaces, an integration of graphic user interfaces with data visualizing programs like gnuplot [27] for result presentation are possible.

Along with this paper, the mcPolymer simulator is published as open source software under the GNU general public license (GPL) [23]. The source code of mcPolymer is part of the appendix of this publication. It also contains instructions how to compile and use the software with the given example models. Anyone can modify and extend the software (e.g., adding new reaction templates). Furthermore a distribution and usage is possible observing the conditions of the GPL.

5. Simulation of Nitroxide-Controlled Homopolymerization

5.1. Kinetic Model

The simulation of the controlled radical polymerization with mcPolymer is exemplified first of all by a thermal-initiated nitroxide-mediated styrene homopolymerization in bulk. In addition to the reactions of the styrene homopolymerization (Figure 2, left), the reactions of alkoxyamine activation and the equilibrium of reversible activation of dormant chains (Figure 2, right) are required. The rate constants of styrene homopolymerization (Table 4) are well established in the literature. For modeling the thermal initiation of styrene and the gel effect we used the empirical Hamielec model [26] (Equations 13 and 14). In our simulation we applied the nitroxide 2,2,5-trimethyl-4-(isopropyl)-3-azahexane-3-oxyle (BIPNO, Figure 3). The corresponding rate constants k_d and k_c (Table 5) were determined using time dependent measurements of the free nitroxide and alkoxyamine concentration and applying the persistent radical effect theory [28].

Figure 2. Kinetic model of nitroxide mediated radical polymerization (NMP).

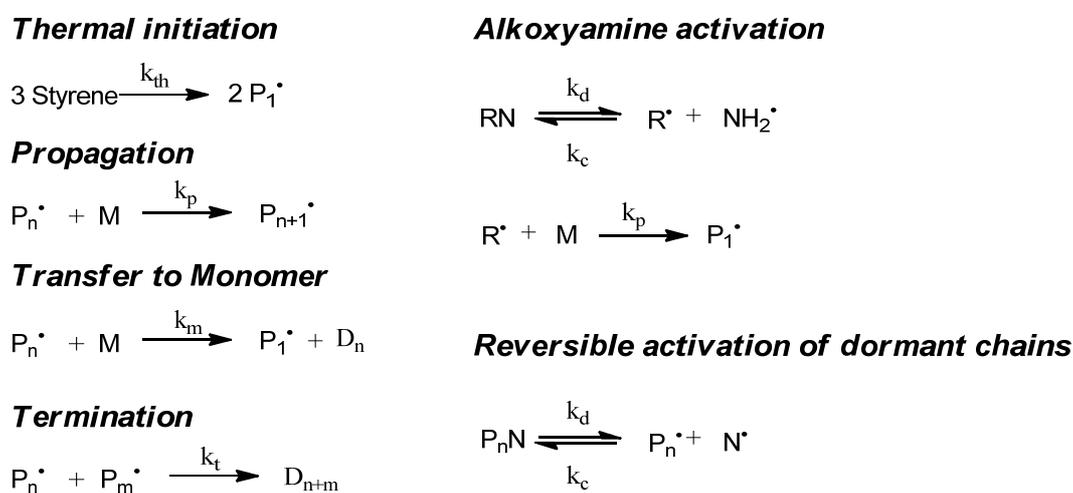


Figure 3. Structure of nitroxide BIPNO (N) and the corresponding alkoxyamine PhEt-BIPNO (RN).

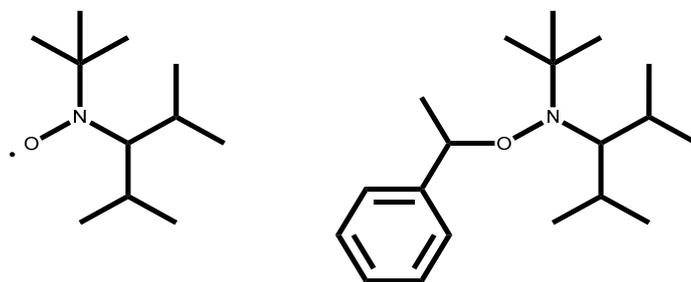


Table 4. Rate constants of styrene homopolymerization.

Rate Constant	Value	Reference
k_{th} ($L^2 mol^{-2} s^{-1}$)	$2.2 \times 10^5 \exp(-13810/T)$	[26]
k_p ($L mol^{-1} s^{-1}$)	$4.2658 \times 10^7 \exp(-3910/T)$	[29]
k_m ($L mol^{-1} s^{-1}$)	$2.31 \times 10^6 \exp(-6377/T)$	[26]
k_t ($L mol^{-1} s^{-1}$)	$2.12 \times 10^9 \exp(-743/T)$	[30]

$$k_{i*} = k_i \cdot \exp(-2 \cdot (A1 \cdot X + A2 \cdot X^2 + A3 \cdot X^3)) \quad (13)$$

$$A1 = 2.57 - 5.05 \cdot 10^{-3} T; \quad A2 = 9.56 - 1.7 \cdot 10^{-2} T; \quad A3 = -3.03 + 7.85 \cdot 10^{-3} T \quad (14)$$

Table 5. Rate constants concerning reversible deactivation of propagating chains of styrene with nitroxide BIPNO at 123 °C.

Rate Constant	Value	Reference
k_d (s^{-1})	6.8×10^{-3}	[28]
k_c ($L mol^{-1} s^{-1}$)	6.32×10^5	[28]

5.2. Implementation with mcPolymer

Figure 4. TCL scheme of an exemplary mcPolymer model.

<pre># include Interface source mcPolymerInterface.tcl # include Hamielec model source Hamielec.tcl # define low molecular species Monomer Styrene Species I# Species R# Species N# Species RN # define macromolecules SpeciesMacro P# SpeciesMacro D SpeciesMacro PN # set initial concentrations Concentration Styrene 8.725 Concentration RN 0.02 # define reaction rate constants RateConstantArrhenius kp 4.2658e+07 3.91e+3 RateConstantArrhenius km 2.31e+6 6.377e+3 RateConstantArrhenius kt 2.12e+09 7.53e+2 RateConstantArrhenius kth 2.2e+05 1.381e+4 RateConstant kd 6.8e-3 RateConstant kc 6.32e5</pre>	<pre># define reactions ThermalInitiation Styrene + Styrene + Styrene --> P# + P# kth Propagation P# + Styrene --> P# kp Termination P# + P# --> D kt Transfer2monomer P# + Styrene --> D + P# km Initiation I# + Styrene --> P# kp Elemental_1E2P RN --> R# + N# kd Elemental_2E1P R# + N# --> RN kc Initiation R# + Styrene --> P# kp Transfer_PL-P P# + N# --> PN kc Transfer_P-PL PN --> P# + N# kd # set reaction temperature Temperature 123.0 # set frequency of intermediate results Setdt 60.0 # set mode of storage: vector SetCalculateMacromolecules on # activate Hamielec model SetUseHamielecModel on # define nsx: number of molecules for simulation InitSimulation 1e9 # start simulation to overall reaction time Simulation 3600</pre>
--	--

Figure 4 illustrates the simple implementation of the kinetic model as a TCL script. At first, we include the mcPolymerInterface and the Hamielec Model. Afterwards, we have to define the low molecular and macromolecular species and set their initial concentrations. The rate constants are declared as either temperature dependent or fixed. The reaction templates formulate the kinetic model using the formerly declared definitions.

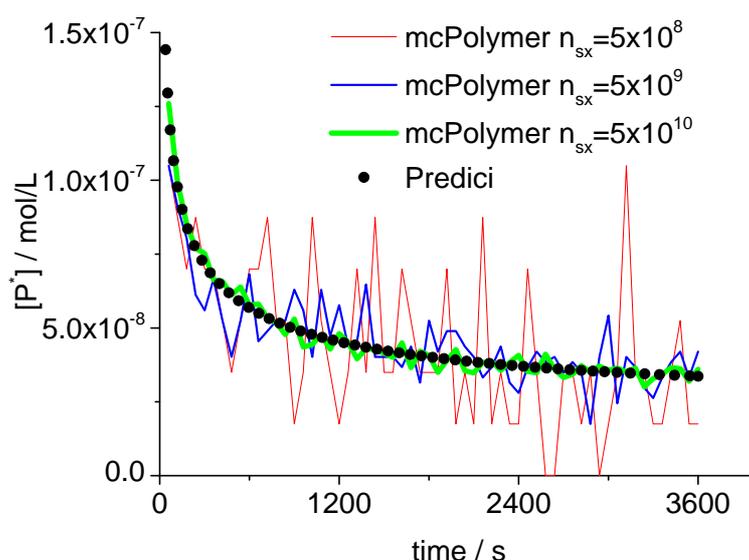
InitSimulation defines the number of molecules n_{sx} being used in the simulation. In this application, the time interval parameter dt not only defines the frequency of the output of the intermediate results, but also the time stamps for recalculating k_t , applying Equations 13 and 14 as function of the monomer conversion (X).

5.3. Validation of the Monte Carlo Approach

The Monte Carlo simulation only yields correct results if the concentrations of the involved species are represented by an adequate number of molecules (n_{sx}). A general statement for the required system size cannot be made since it depends on the concrete reaction conditions. In our example, the monomer concentration scales at 10^1 while the radical concentration P_n^{\cdot} is about 10^{-8} .

Therefore, we can derive a minimum number of 10^8 molecules for the actual simulation. Proper results considering the monomer conversion and the molar mass distribution can already be obtained with 5×10^8 molecules. Evoked by the persistent radical effect [31], we are in a domain of non-stationary radical concentration. This effect does not show up until in a simulation with at least 5×10^9 molecules (Figure 5). At the point of initialization (Equation 10) as well as during the simulation, the actual number of molecules is directly proportional to the total number of particles (n_{sx}). Here ($n_{sx} = 5 \times 10^8$), the propagating polymer chains P_n^{\cdot} were represented by approximately 20–70 discrete objects in the MC simulation.

Figure 5. Concentration of propagating chains compared to PREDICI simulation results.

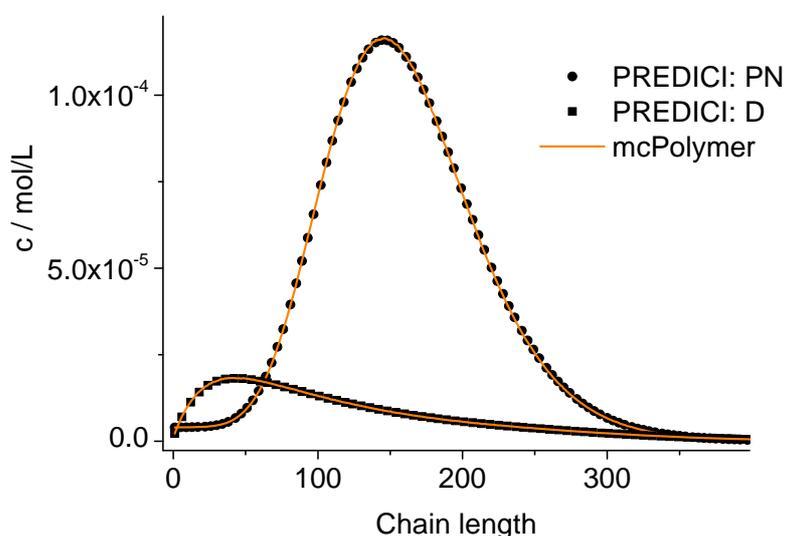


A validation of the MC simulation can be carried out by a comparison with the simulation results of the commercial program PREDICI ($[\text{PhEt-BIPNO}]_0 = 0.02 \text{ mol/L}$; Reaction Time = 1 h). PREDICI is a deterministic differential equation solver, so the formerly discussed simulation parameter does not

matter. Comparing computation performance between PREDICI and a Monte Carlo simulator are not very useful, since the computation time of the latter varies dependent on the system size, while PREDICI's remains constant. The execution time of the model discussed here is approx. 15 seconds in PREDICI (PREDICI Version 6.20.2, Intel Core i5-650, OS: MS Windows 7).

The correctness of the applied kinetic model was proven earlier on the basis of experimental data [19]. Considering the perfect correlation between the simulation by mcPolymer and the results obtained by the PREDICI simulation (Figure 6), we conclude a correct implementation of the kinetic model of the nitroxide-controlled polymerization with mcPolymer. Due to the comparatively high concentrations of the polymer species (PN: 1.45×10^{-2} mol/L, D: 3.8×10^{-3} mol/L) presented in Figure 6, no noise appears in the chain length distribution results.

Figure 6. Chain length distributions compared to PREDICI simulation results.



5.4. Performance Analysis

The computation time depends on the number of molecules (n_{sx}), the mode of storage of the macromolecular species (histogram/vector), and the concentration of the reactants. In addition to the mere computation time, the speed performance of the simulation can be rated by the number of Monte Carlo steps executed per second. We introduced this performance parameter with the first application of mcPolymer [17]. Later on, it was used by Barner Kowollik *et al.* [18] for benchmarking parallelized Monte Carlo simulations and by Broadbelt *et al.* [32] for performance analysis of nitroxide-controlled copolymerization.

The performance analysis has been processed on the following two systems:

- 2 AMD Opteron 6134 Magny Cours CPU (16 cores, 2.2 GHz), 128 GB RAM, OS: Open SuSE Linux 11.4, compiler: GNU Compiler Collection (gcc) 4.5.1 → Linux-Opteron;
- 1 Intel Core i5-650 Clarkdale (2 Cores, 3.2–3.46 GHz), 16 GB RAM, OS: MS Windows 7 Enterprise 64 bit, compiler: Visual Studio 2010 Ultimate 64 bit → Windows-Core i5.

The results of the test runs are summarized in Tables 6 and 7. As one can see, the number of MC steps is directly proportional to the system size n_{sx} . Due to persistent radical effect, the radical

concentration is not constant and depends on the initial concentration of the alkoxyamine. Therefore, at the same reaction time and higher alkoxyamine concentration, a higher monomer conversion is achieved and thus more Monte Carlo steps are required.

Table 6. Performance analysis on the Linux-Opteron system, reaction time = 1 h.

n_{sx}	[PhEt-BIPNO] ₀ = 0.02 mol/L			[PhEt-BIPNO] ₀ = 0.01 mol/L		
	MC steps	Time/s	MCpsec/s ⁻¹	MC steps	Time/s	MCpsec/s ⁻¹
1×10^9	3.85×10^8	108	3.53×10^6	2.85×10^8	24	3.91×10^6
5×10^9	1.92×10^9	559	3.43×10^6	1.42×10^9	122	3.88×10^6
1×10^{10}	3.85×10^9	1174	3.28×10^6	2.85×10^9	250	3.79×10^6
5×10^{10}	1.92×10^{10}	6156	3.12×10^6	1.42×10^{10}	1273	3.73×10^6
1×10^{11}	3.85×10^{10}	13239	2.91×10^6	2.85×10^{10}	2621	3.62×10^6

Table 7. Performance analysis on the Windows-Core i5 system; [PhEt-BIPNO]₀ = 0.02 mol/L, reaction time = 1 h.

n_{sx}	MC steps	Vector storage mode		Histogram mode	
		Time/s	MCpsec/s ⁻¹	Time/s	MCpsec/s ⁻¹
1×10^9	3.85×10^8	61	6.31×10^6	73	5.27×10^6
5×10^9	1.92×10^9	319	6.03×10^6	357	5.39×10^6
1×10^{10}	3.85×10^9	685	5.61×10^6	711	5.41×10^6
5×10^{10}	1.92×10^{10}	3689	5.21×10^6	3568	5.39×10^6
1×10^{11}	3.85×10^{10}	–	–	7140	5.39×10^6

Since the simulation program is executed on one CPU core only, it is not surprising that the Windows system runs faster due to the higher CPU clocking. Using the vector storage mode for the macromolecular species, the program performance increases due to the simpler selection of the macromolecules, but leads to a massive increase of the memory requirements (each polymer object has to be stored individually in the RAM). The latter depend on the initial concentration of the alkoxyamine and the simulation parameter n_{sx} . Taking the example of [BIPNO]₀ = 0.02 mol/L and $n_{sx} = 10^{10}$, the memory requirements are up to 2.1 GB RAM in vector storage mode compared to 0.014 GB RAM in histogram mode.

The number of Monte Carlo steps executed per seconds (MCpsec) remains constant during the simulation in histogram mode, while it decreases in vector mode. We assume the reason might lie in the increasing number of cache misses. An indication is the stronger correlation between MCpsec and n_{sx} on the Windows system.

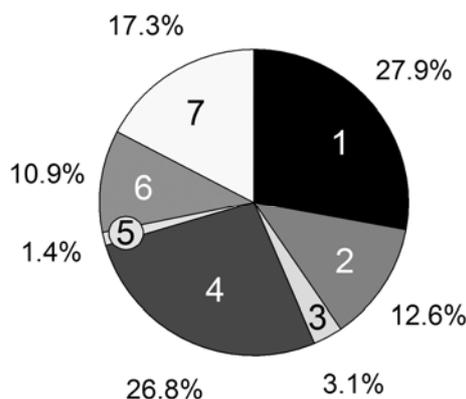
For further improvements of our program, we analyzed the statistical time consumption of the particular parts of a MC simulation step. Using the CPU profiler of the Google Performance Tools [33], we could measure the computation time of every single line of code. The following relevant parts are summarized in Figure 7:

- 1 Update of the reaction rates R_v and probabilities p_v (Equations 2, 6–8);
- 2 Choice of a reaction (Equation 1);
- 3 Particle numbers of the reactants X_i are changed by consuming the educts and creating the products;

- 4 Execution of the propagation reaction;
- 5 Execution of the Transfer_PL-P reaction;
- 6 Execution of the Transfer_P-PL reaction;
- 7 Calculation of the time interval (Equation 9).

For the computation time only, the propagation reaction and the reversible deactivation of the propagating chain by the nitroxide are important, while all other reactions are negligible. Remarkable is the large difference between the reactions Transfer_PL-P (5) and Transfer_P-PL (6), which are performed almost equally as often. The reason is the vast concentration difference between the propagating chain (macromolecular educt in Transfer_PL-P) and the dormant chains (macromolecular educt in Transfer_P-PL). The number of propagating chains is comparatively small and can be kept in cache, causing a very fast individual educt molecule selection.

Figure 7. Persistence simulation time; $[\text{PhEt-BIPNO}]_0 = 0.02 \text{ mol/L}$, Reaction Time = 1 h; Vector storage mode; 1: Update R_p and p_p ; 2: Reaction choice; 3: Execute reaction (X_i); 4: Propagation; 5: Transfer_PL-P; 6: Transfer_P-PL; 7: Calculate τ .



6. Simulation of ATRP Homopolymerization

6.1. Kinetic Model and Implementation with mcPolymer

The presented kinetic model of the ATRP polymerization is well-known in the literature. Matyjaszewski *et al.* published a modeling approach of chain-end functionality in atom transfer radical polymerization using PREDICI [34,35]. Based on these models, Soares *et al.* introduced a Monte Carlo simulation of ATRP [14]. It was validated with experimental data and by the simulation method of moments. Its implementation with MATLAB only allowed comparatively small molecule numbers (n_{sx} up to 2.7×10^7). This model was extended to the simulation of ATRP with bifunctional initiators [15] and to the modeling of the semibatch copolymerization [36]. Najafi *et al.* utilized the ATRP model for simulating the chain-length dependent termination [37].

Figure 8 illustrates the applied model of ATRP polymerization. The styrene homopolymerization and the corresponding reaction rates (Table 4) were used identically as in the NMP model presented earlier.

The specific reactions of ATRP on the right side of the scheme were implemented by two elemental reactions using template ‘Elemental_2E2P’; One reaction applying template ‘Initiation’ and two transfer reactions of the macromolecular species using the template ‘Transfer_PL-PL’. All templates were introduced in Table 2. The required reaction rate constants (Table 8) were experimentally determined by Fukuda *et al.* [38] (k_a) and Matyjaszewski *et al.* [39] ($K = k_a/k_d$) for the catalyst $\text{Cu}^I\text{X}/2\text{L}$ (X: Br, L: 4,4'-di-*n*-heptyl-2,2'-bipyridine).

Figure 8. Kinetic model of atom transfer radical polymerization (ATRP).

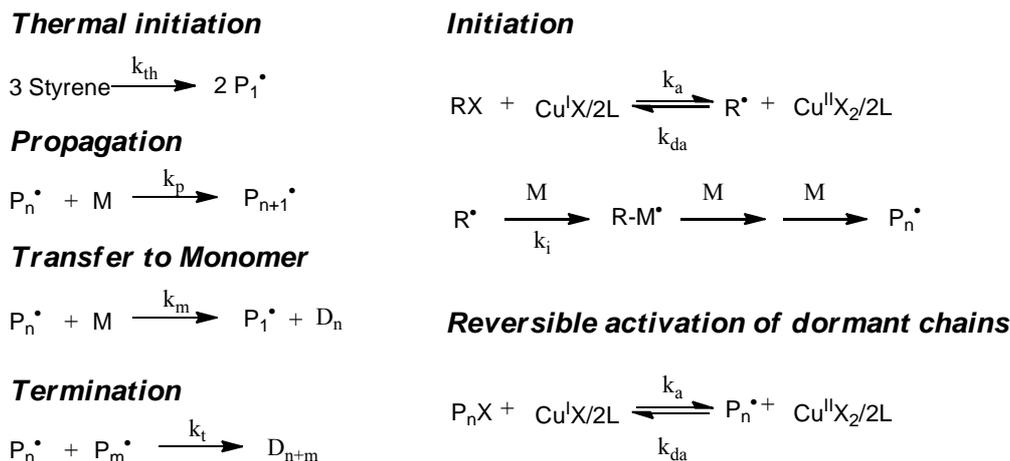


Table 8. Rate constants concerning reversible activation of dormant chains at 110 °C.

Rate Constant	Value	Reference
k_a (L mol ⁻¹ s ⁻¹)	4.5×10^{-1}	[38]
k_{da} (L mol ⁻¹ s ⁻¹)	1.1×10^7	[39]

6.2. Validation and Performance Analysis

For the validation of our Monte Carlo implementation, we compared the results with those obtained by a PREDICI simulation. Figure 9 shows a good correlation between the results concerning monomer concentration and molar masses, which are already reached at moderate system sizes regarding n_{sx} .

Comparing to the nitroxide-controlled polymerization, the ATRP simulation requires a lot more computation time in relation to the system size n_{sx} (Table 9). This effect can be explained by the reaction rates of the reversible activation equilibrium (k_a/k_d), which are about two orders of magnitude higher than those of nitroxide-controlled polymerization. Another aspect is the higher concentration of the controlled species and therefore that of the dormant chains. Both factors cause an increase in the MC steps at the same reaction time. The performance identifiers MCpsec of both models are contiguous, which is expected since in both cases the relevant reaction templates (NMP: Transfer_PL-P, Transfer_P-PL; ATRP: Transfer_PL-PL) transfer one macromolecular species into another reactant. Additionally, the result points out the comparability of the performance analysis using the parameter MCpsec.

Figure 9. Validation of the mcPolymer simulation with PREDICI; $n_{sx} = 10^8$; $[RX]_0 = [Cu^I X/2L]_0 = 0.08725$ mol/L.

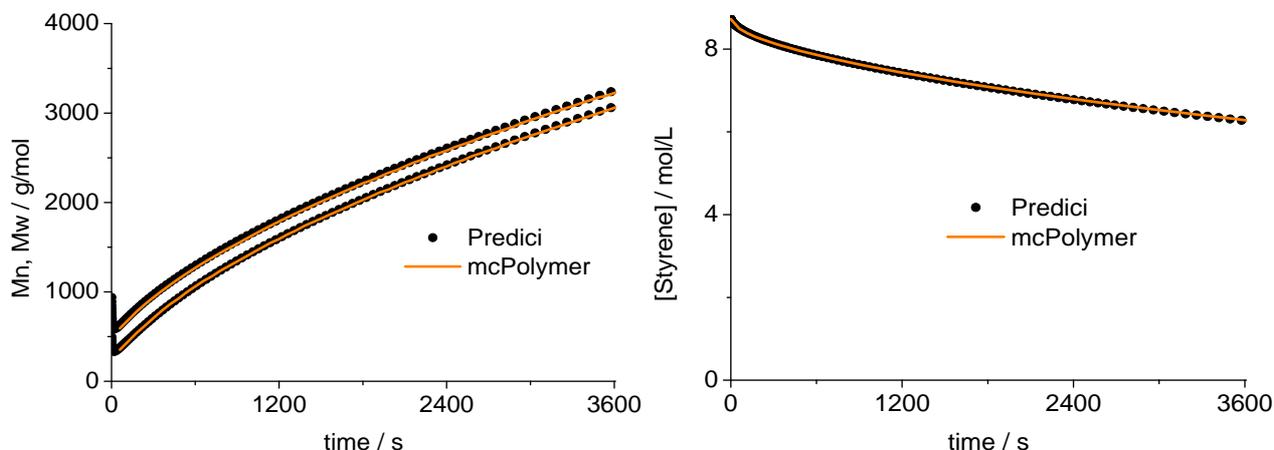


Table 9. Performance analysis; reaction time = 1 h; $[RX]_0 = [Cu^I X/2L]_0 = 0.08725$ mol/L.

n_{sx}	MC steps	Linux-Opteron		Windows-Core i5 system	
		Time/s	MCpsec/s ⁻¹	Time/s	MCpsec/s ⁻¹
1×10^8	2.71×10^8	83	3.27×10^6	72	3.77×10^6
5×10^8	1.36×10^9	448	3.03×10^6	353	3.84×10^6
1×10^9	2.71×10^9	933	2.91×10^6	725	3.74×10^6
5×10^9	1.36×10^{10}	4811	2.82×10^6	4014	3.38×10^6
1×10^{10}	2.71×10^{10}	9750	2.78×10^6	—	—

7. Simulation of RAFT Polymerization

7.1. Kinetic Model

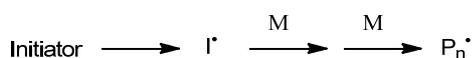
The kinetic model of the Cumyl dithiobenzoate (CDB) mediated methyl acrylate (MA) RAFT polymerization that we applied is illustrated in Figure 10. We determined the corresponding reaction rates (Table 10) by fitting the model to the experimental data [17]. As a side reaction, the model contains a cross termination between the propagating chain and the intermediate radical of the RAFT equilibrium. The ongoing discussion “cross termination vs. stable intermediate model” [40–44] is not the subject of this paper. Perrier *et al.* demonstrated that both approaches were able to describe the experimental data of the RAFT polymerization adequately [41].

Table 10. Rate constants cumyl dithiobenzoate mediated methyl acrylate homopolymerization.

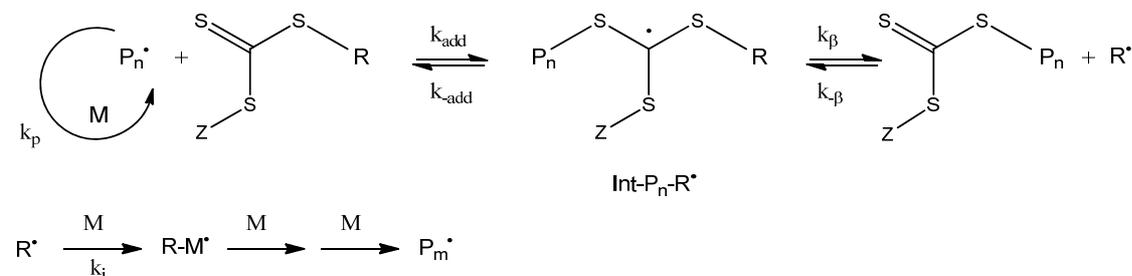
Rate constant	Value	Reference
k_i (s ⁻¹)	$2.89 \times 10^{15} \exp(-15663/T)$	[45]
f	0.64	[46]
k_p (L mol ⁻¹ s ⁻¹)	$3.61 \times 10^6 \exp(-1671/T)$	[47]
k_t (L mol ⁻¹ s ⁻¹)	$1.59 \times 10^{10} \exp(-1082/T)$	[48]
k_{add} (L mol ⁻¹ s ⁻¹)	3.06×10^8	[17]
k_{-add} (s ⁻¹)	20×9	[17]
k_{addP} (L mol ⁻¹ s ⁻¹)	9.36×10^6	[17]
k_{-addP} (s ⁻¹)	784	[17]
α	0.15	[17]

Figure 10. Kinetic model of reversible addition fragmentation chain transfer (RAFT) polymerization.

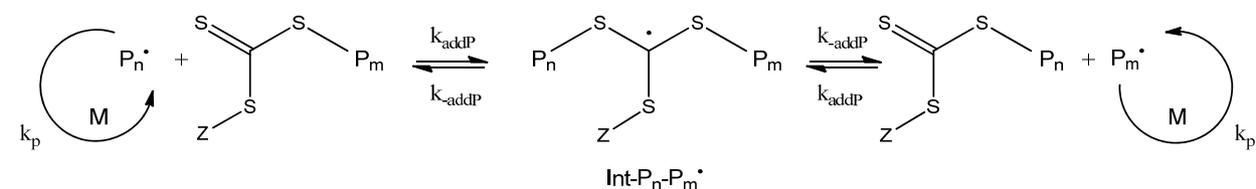
Initiation



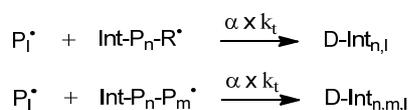
Reversible chain transfer/propagation



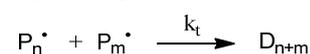
Chain equilibration/propagation



Cross-termination



Termination



The validation of the kinetic model was performed [17] using experimental data measuring the monomer conversion as function of time, the full molecular weight distributions, and the concentrations of the intermediate RAFT radical with experimental electron spin resonance spectroscopic data.

7.2. Implementation with mcPolymer

The RAFT polymerization is an example for the advantages of Monte Carlo methods when simulating polymerization models containing complex macromolecular species. In the main equilibrium of the RAFT polymerization, a species with two polymer chains appears as an intermediate radical. During a MC simulation, we can conserve both polymer chains in a molecular object and, therefore, are able to model a fragmentation of this molecule quite simply.

The mcPolymer interface offers the macromolecular species “SpeciesMacro_P-O-P” (Table 1) for modeling the intermediate radical in the main equilibrium. During the cross termination reaction, a three-arm star polymer is created, which is represented by a macromolecular object of type “SpeciesMacro_P-O<PP” (Table 1).

For modeling the pre-equilibrium, we can use the reaction templates “Transfer_PL-P” and “Transfer_P-PL” as we have already used them for the NMP, while the main equilibrium of the RAFT polymerization can be implemented utilizing the templates “ChainAddition” and “ChainFragmentation” (Table 1).

In contrast to mcPolymer, RAFT polymerization simulations with PREDICI have to simulate every polymer chain separately as a unique pseudo species [9]. This approach also leads to correct results [10], but the analyses are more complex.

As a conclusion, simulating RAFT polymerizations with mcPolymer offers a deep view inside the complex macromolecular structures using a straight model representation, while the produced results are simple for handling and analyses.

7.3. mcPolymer in Parallel Environments

Barner-Kowollik *et al.* presented a parallel high performance Monte Carlo simulation approach for complex polymerization using the example of RAFT polymerization [18]. In this paper, the simulation was compared to PREDICI regarding the influence of the following simulation parameters: the system size (n_{sx} range between 1×10^9 and 1×10^{10} particles), the number of parallel processes (4 to 16), and the reaction time between synchronization (2 to 500 s). It was not surprising that the only important factor was the number of particles available for each process. As long the number is large enough to remain in the continuum, the simulation provides correct results independent of the synchronization time.

We implemented a parallel simulation of the formerly discussed RAFT polymerization model with mcPolymer. The system size was set up in the range of 1×10^9 – 2.5×10^{10} for each process (10 processes total). After a fixed synchronization time, all molecules, including the complex macromolecular objects are collected, mixed up, and statistically redistributed over the processes. Each process requires approx. 2.5 GB of RAM in vector storage mode ($n_{sx} = 1 \times 10^{10}$, 10 processes). Therefore, 25 GB have to be shifted at every single synchronization step. The results in Table 11 make obvious that synchronizations after every 5 min of reaction time (12 cycles) determine the speed of the whole simulation. The simulation was executed with overall 10^{11} molecules in 11723 s. The results agree with those obtained in a single process set up with 10^{11} particles, but the latter required 86897 s. In our test runs we could not detect an influence of the synchronization time on the simulation results, which leads to the conclusion that we can generally omit the synchronization steps and merge the individual results (chain length distribution) at the end. The results are qualitatively the same as in the single process—as long as the precondition of every simulation process remaining in the continuum is fulfilled. The speed-up factor reaches 14.3 with 10 processes (Table 11).

Using parallelization, a large number of molecules can be obtained, which can be used to smooth all concentration-time curves. Additionally, the number of polymer species used for statistical analysis is increased. Unfortunately, the underlying single MC simulation is not accelerated, since it still needs a certain system size.

An effective acceleration could be achieved, if we were able to parallelize each Monte Carlo step. As an example, the calculation of the reaction probabilities could be processed in parallel. Since the sequential part of a Monte Carlo step is over 50 percent of the time consumption and the mean

execution time of a MC step (e.g., 5.5×10^{-7} s from 1.81×10^6 MC steps/s; $n_{sx} = 5 \times 10^9$) is in dimension of the memory synchronization between the processes, the attempt of parallelization at the level of a single MC step and the common x86 CPU architecture is futile.

Table 11. Performance analysis in parallel environments (10 processes); Reaction time = 1 h; Linux-Opteron; $[CDB]_0 = 0.025$ mol/L; $[MA]_0 = 11.04$ mol/L.

System size n_{sx} of one process	Overall system size	Synchronization cycles	Parallel simulation time/s	Sequential synchronization time/s	Overall simulation time/s
1×10^9	1×10^{10}	12	817	317	1134
1×10^9	1×10^{10}	24	1059	669	1728
1×10^9	1×10^{10}	60	1826	1692	3518
1×10^{10}	1×10^{11}	12	8453	3270	11723
1×10^{10}	1×10^{11}	24	8974	6741	15715
2.5×10^{10}	2.5×10^{11}	12	23022	9423	32445

7.4. Performance Analysis

The addition reaction of a propagating chain (P_n^\cdot) to a dormant species (macro RAFT), predominantly determines the computation time of the RAFT polymerization with mcPolymer (Figure 11). For modeling the intermediate radical in the main equilibrium, the complex macromolecular species “SpeciesMacro_P-O-P” (Table 1) is built up by two polymer chains. While performing ‘ChainAddition’, three steps are necessary that consume varying computation time:

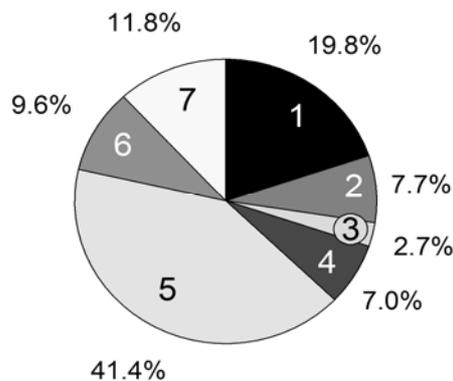
- Selecting propagating chain “SpeciesMacro”: 2.3% computation time
- Selecting dormant species “SpeciesMacro”: 34.4% computation time
- Building up the intermediate radical “SpeciesMacro_P-O-P”: 63.3% computation time

When performing this reaction, a discrepancy concerning the time of memory access shows up. The concentration of propagating chains P_n^\cdot is a few orders of magnitude smaller than that of the dormant species. Therefore, the small vector can mostly be kept in the CPU cache while the dormant chain vector occupies the larger part of the RAM. Thus, the performance advantage of the Core i5 system is significantly smaller than expected and decreases consistently with increasing system size (Table 12). In contrast to the nitroxide-controlled polymerization, the histogram storage mode is no alternative due to the complex data structure.

Table 12. Performance analysis; Reaction time = 1 h; $[CDB]_0 = 0.025$ mol/L; $[MA]_0 = 11.04$ mol/L.

n_{sx}	MC steps	Linux-Opteron		Windows-Core i5 system	
		Time/s	MCpsec/s ⁻¹	Time/s	MCpsec/s ⁻¹
1×10^9	1.05×10^9	525	2.00×10^6	400	2.63×10^6
5×10^9	5.27×10^9	2912	1.81×10^6	2534	2.08×10^6
1×10^{10}	1.05×10^{10}	6083	1.73×10^6	5582	1.89×10^6
5×10^{10}	5.27×10^{10}	39848	1.33×10^6	35814	1.47×10^6
1×10^{11}	1.05×10^{11}	86897	1.21×10^6	–	–

Figure 11. Persistence simulation time; 1: Update R_v and p_v ; 2: Reaction choice; 3: Execute reaction (X_i); 4: Propagation; 5: ChainAddition; 6: ChainFragmentation; 7: calculate τ .



8. Simulation of Nitroxide-Controlled Copolymerization

8.1. Kinetic Model

A nitroxide-controlled homopolymerization of methyl methacrylate (MMA) is not possible due to the degradation of nitroxide by H-abstraction [49]. The SG1-controlled radical copolymerization of MMA and styrene (S) is therefore of particular interest, since Charleux *et al.* [50] demonstrated the possibility of performing a nitroxide-controlled polymerization of methyl methacrylate with a small amount of styrene. Besides experimental work, the influence of the copolymer styrene was analyzed by model calculations [51]. Charleux *et al.* implemented the copolymerization model in PREDICI considering the penultimate effect [51]. Furthermore, Broadbelt *et al.* created Monte Carlo models of SG1-controlled MMA/S copolymerization, examining the copolymer composition and sequence effects [20–22].

We implemented the model of SG1 controlled MMA/S copolymerization published by Charleux *et al.* [51]. It contains the following reactions:

- Dissociation equilibrium of the SG1-alcoxyamine;
- Initiation of new polymer chains in consideration of the first two monomer units for modeling the penultimate effect;
- Combination reactions of low molecular radicals;
- Propagation reactions;
- Reversible deactivation of propagating chains by the nitroxide SG1;
- Termination of macromolecules by combination and disproportionation;
- β -hydrogen transfer from propagating macroradicals with chain end MMA to nitroxide SG1.

Formulating the entire 52 reactions and 37 reaction rate constants along with their references would go beyond the scope of this paper and can be found in the appendix of the publication by Charleux *et al.* [51] and in our appendix material as mcPolymer model (Appendix-5-Example-SG1-MMA-S-copolymerization.tcl). The initial concentrations of the reactants are: $[MMA]_0 = 8.48 \text{ mol/L}$, $[Styrene]_0 = 0.82 \text{ mol/L}$, $[SG1\text{-alcoxyamine}]_0 = 0.0273 \text{ mol/L}$ and $[SG1]_0 = 0.00274 \text{ mol/L}$.

8.2. Validation of the Monte Carlo Approach

The mcPolymer simulation was validated using a PREDICI model. We examined the time-dependent concentrations of the MMA and S comonomers (Figure 12) and, therefore, the copolymer composition.

Figure 12 shows the agreement of our implementation (mcPolymer) with the PREDICI model considering the concentration-time profile of the essential reactants of the polymerization:

- The nitroxide: SG1;
- The SG1 terminated polymer chains with a chain end sequence MMA-S: P-MMA-S-SG1;
- The dead polymers resulting from termination reactions: D;
- The dead polymers formed by β -hydrogen transfer from propagating macroradicals with chain end MMA to nitroxide SG1: D-Htr.

The chain length distributions of the products P-MMA-S-SG1, D and D-Htr determined after two hour reaction time also results in a perfect correlation between our simulator and the PREDICI model (Figure 13).

Figure 12. Validation of mcPolymer with PREDICI: Concentration-time-profiles.

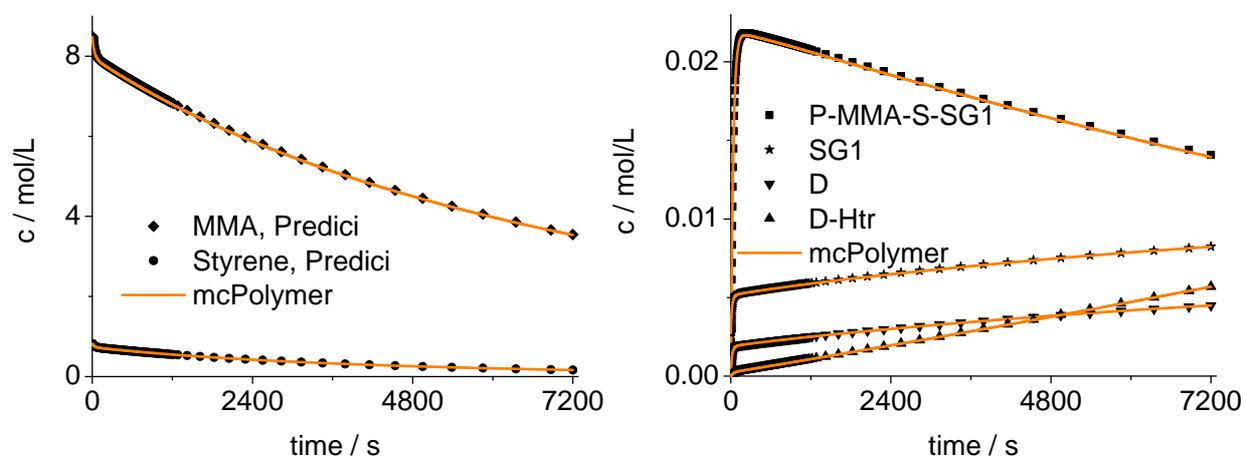
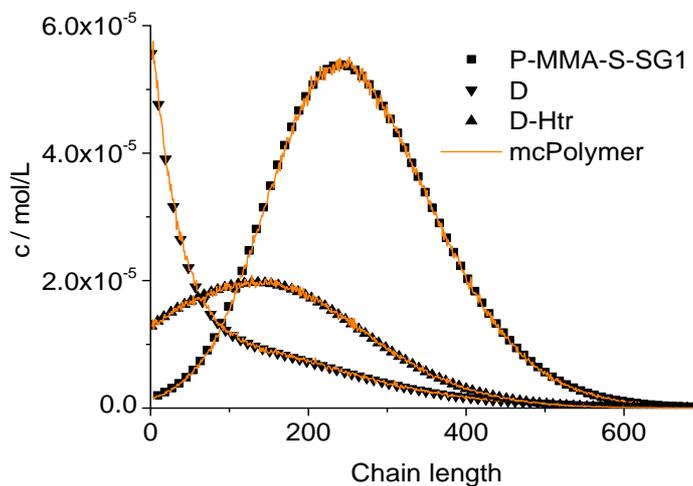


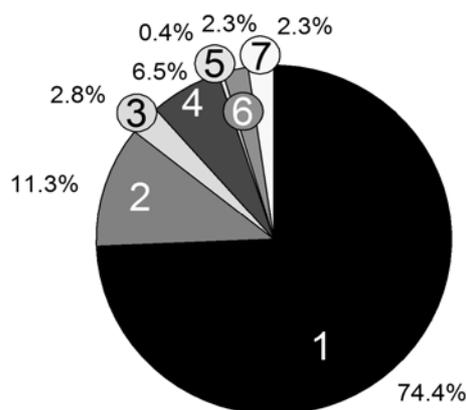
Figure 13. Validation of mcPolymer with PREDICI: Chain length distributions.



8.3. Performance Analysis

The model of the nitroxide-controlled copolymerization contains a total of 52 reactions. Therefore, the calculations of reaction probabilities (Equations 2, 6–8) and the selection of the reaction to be executed (Equation 1) are decisive for the simulation speed. Overall, approximately 86% of computation time is consumed for those two steps (Figure 14).

Figure 14. Persistence simulation time; 1: Update R_v and p_v ; 2: Reaction choice; 3: Execute reaction (X_i); 4: Propagation; 5: TransferCopolymer_PL-P; 6: TransferCopolymer_P-PL; 7: calculate τ .



In comparison to our BIPNO-controlled styrene homopolymerization model, just under half of the MCpsec rate is achieved with a similar concentration of dormant chains (Tables 7 and 13). Since no memory-intensive activities are speed determined, the MCpsec rate remains nearly constant during increasing system size (Table 13). The performance factor between the Core i5 and Opteron system is approximately 1.8 and is independent of the system size.

Table 13. Performance analysis; Reaction time = 1 h.

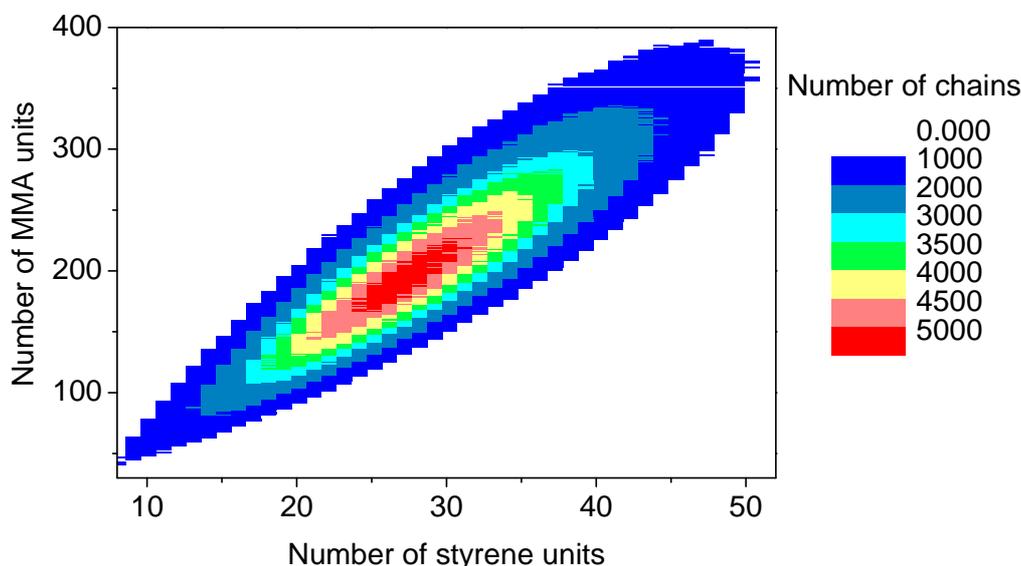
n_{sx}	MC steps	Linux-Opteron		Windows-Core i5 system	
		Time/s	MCpsec/s ⁻¹	Time/s	MCpsec/s ⁻¹
1×10^9	4.80×10^8	313	1.54×10^6	177	2.71×10^6
5×10^9	2.40×10^9	1602	1.50×10^6	874	2.75×10^6
1×10^{10}	4.80×10^9	3210	1.49×10^6	1754	2.74×10^6
5×10^{10}	2.40×10^{10}	16120	1.49×10^6	8924	2.69×10^6
1×10^{11}	4.80×10^{10}	32847	1.46×10^6	18012	2.66×10^6

8.4. Analysis of Simulation Results

The discrete recording of the composition of each polymer chain offers a thorough analysis of the molecular and chemical structure of the polymers formed by the copolymerization. Contour plots (Figure 15) are an example of coupled breakdown of chain length and copolymer composition of polymers, known from the MALDI-TOF analytic of copolymers [52,53]. The addition of the number of styrene units (x -axis) with the number of MMA units (y -axis) results in the chain length of the corresponding data point. Due to the concentration relation in the reaction solution and the reactivity

ratios ($[MMA]_0 = 8.48$ mol/L, $[styrene]_0 = 0.82$ mol/L, $r_{MMA/S} = 0.420$ [52], $r_{S/MMA} = 0.517$ [52]), MMA is predominantly built in in the polymer chain. Since the simulation was performed with a large number of molecules ($n_{sx} = 5 \times 10^{10}$), the frequency distributions are well separated. The chain length and composition distribution of a total number of 1.5×10^7 polymer chains is presented in this plot.

Figure 15. Contour plot: Coupling copolymer composition with chain length distribution.



9. Conclusions

In this paper, a Monte Carlo simulation program named mcPolymer for simulating radical polymerizations was presented. It was provided as an open source project under the GPL license. Our mcPolymer simulator runs on MS Windows and Linux platforms. Regarding the intensive memory consumption, it is recommended to use 64 bit systems.

The software architecture of mcPolymer was introduced focusing on the data management of polymer chains. Due to the flexible interface, the formulation of complex polymerization models and the setup of its reaction conditions are quite simple.

The application was demonstrated on examples of controlled radical polymerizations (nitroxide-controlled, ATRP, RAFT, nitroxide-controlled copolymerization). We relied on formerly published models validated by experimental data. Using the example of RAFT polymerization, we discussed applying mcPolymer in parallel environments. The example of nitroxide-controlled copolymerization demonstrated how to also obtain the copolymer composition of each polymer chain along with molar mass distribution as a simulation result.

PREDICI models were used for validating the MC program. A crucial point of Monte Carlo simulations for modeling radical polymerization, is the system size. Different possibilities were exemplarily illustrated for finding an adequate balance between precision and computation time of the simulation.

For all examples of controlled radical polymerization, we presented detailed performance analyses demonstrating the influence of the system size, concentrations of reactants and, therefore, of the dormant chains, as well as the peculiarities of data management, and the CPU architecture.

mcPolymer is a powerful tool for simulating controlled radical polymerizations. Due to its flexible software architecture and interface, new reaction templates can easily be implemented and, thus, be adjusted to further polymerization models.

References

1. Moad, G.; Rizzardo, E.; Thang, S.H. Living radical polymerization by the RAFT process—A second update. *Austr. J. Chem.* **2009**, *62*, 1402–1472.
2. Matyjaszewski, K. Atom transfer radical polymerization (ATRP): Current status and future perspectives. *Macromolecules* **2012**, *45*, 4015–4039.
3. Goto, A.; Fukuda, T. Kinetics of living radical polymerization. *Prog. Polym. Sci.* **2004**, *29*, 329–385.
4. D'hooge, D.R.; Reyniers, M.F.; Stadler, F.J.; Dervaux, B.; Bailly, C.; Du Prez, F.E.; Marin, G.B. Atom transfer radical polymerization of isobornyl acrylate: A kinetic modeling study. *Macromolecules* **2010**, *43*, 8766–8781.
5. Zapata-González, I.; Saldívar-Guerra, E.; Ortiz-Cisneros, J. Full molecular weight distribution in RAFT polymerization. New mechanistic insight by direct integration of the equations. *Macromol. Theory Simul.* **2011**, *20*, 370–388.
6. Zapata-González, I.; Saldívar-Guerra, E.; Flores-Tlacuahuac, A.; Vivaldo-Lima, E.; Ortiz-Cisneros, J. Efficient numerical integration of stiff differential equations in polymerisation reaction engineering: Computational aspects and applications. *Can. J. Chem. Eng.* **2012**, *90*, 804–823.
7. Wulkow, M. The simulation of molecular weight distributions in polyreaction kinetics by discrete Galerkin methods. *Macromol. Theory Simul.* **1996**, *5*, 393–416.
8. Wulkow, M. Computer aided modeling of polymer reaction engineering—The status of Predici, I-Simulation. *Macromol. React. Eng.* **2008**, *2*, 461–494.
9. Barner-Kowollik, C.; Quinn, J.F.; Morsley, D.R.; Davis, T.P. Modeling the reversible addition-fragmentation chain transfer process in cumyl dithiobenzoate-mediated styrene homopolymerizations: Assessing rate coefficients for the addition-fragmentation equilibrium. *J. Polym. Sci. Part A Polym. Chem.* **2001**, *39*, 1353–1365.
10. Wulkow, M.; Busch, M.; Davis, T.P.; Barner-Kowollik, C. Implementing the reversible addition-fragmentation chain transfer process in PREDICI. *J. Polym. Sci. Part A Polym. Chem.* **2004**, *42*, 1441–1448.
11. Tang, W.; Matyjaszewski, K. Kinetic modeling of normal ATRP, normal ATRP with $[Cu^{II}]_0$, reverse ATRP and SR&NI ATRP. *Macromol. Theory Simul.* **2008**, *17*, 359–375.
12. Gimes, D.; Bertin, D.; Lefay, C.; Guillaneuf, Y. Kinetic modeling of nitroxide-mediated polymerization: conditions for living and controlled polymerization. *Macromol. Theory Simul.* **2009**, *18*, 402–419.
13. Nicolas, J.; Mueller, L.; Dire, C.; Matyjaszewski, K.; Charleux, B. Comprehensive modeling study of nitroxide-mediated controlled/living radical copolymerization of methyl methacrylate with a small amount of styrene. *Macromolecules* **2009**, *42*, 4470–4478.

14. Al-Harathi, M.; Soares, J.B.P.; Simon, L.C. Dynamic Monte Carlo simulation of atom-transfer radical polymerization. *Macromol. Mater. Eng.* **2006**, *291*, 993–1003.
15. Al-Harathi, M.; Soares, J.B.P.; Simon, L.C. Dynamic Monte Carlo simulation of ATRP with bifunctional initiators. *Macromol. React. Eng.* **2007**, *1*, 95–105.
16. Al-Harathi, M.A.; Masihullah, J.K.; Abbasi, S.H.; Soares, J.B.P. Dynamic Monte Carlo simulation of ATRP in a batch reactor. *Macromol. Theory Simul.* **2009**, *18*, 307–316.
17. Drache, M.; Schmidt-Naake, G.; Buback, M.; Vana, P. Modeling RAFT polymerization kinetics via Monte Carlo methods: Cumyl dithiobenzoate mediated methyl acrylate polymerization. *Polymer* **2005**, *46*, 8483–8493.
18. Chaffey-Millar, H.; Stewart, D.; Chakravarty, M.M.T.; Keller, G.; Barner-Kowollik, C. A parallelised high performance Monte Carlo simulation approach for complex polymerisation kinetics. *Macromol. Theory Simul.* **2007**, *16*, 575–592.
19. Drache, M. Modeling the product composition during controlled radical polymerizations with mono- and bifunctional alkoxyamines. *Macromol. Symp.* **2009**, *275–276*, 52–58.
20. Wang, L.; Broadbelt, L.J. Kinetics of segment formation in nitroxide-mediated controlled radical polymerization: Comparison with classic theory. *Macromolecules* **2010**, *43*, 2228–2235.
21. Wang, L.; Broadbelt, L.J. Explicit sequence of styrene/methyl methacrylate gradient copolymers synthesized by forced gradient copolymerization with nitroxide-mediated controlled radical polymerization. *Macromolecules* **2009**, *42*, 7961–7968.
22. Wang, L.; Broadbelt, L.J. Factors affecting the formation of the monomer sequence along styrene/methyl methacrylate gradient copolymer chains. *Macromolecules* **2009**, *42*, 8118–8128.
23. GNU General Public License. Available online: <http://www.gnu.de/documents/gpl-3.0.en.html> (accessed on 17 June 2012).
24. Gillespie, D.T. Exact stochastic simulation of coupled chemical reactions. *J. Phys. Chem.* **1977**, *81*, 2340–2361.
25. Matsumoto, M.; Nishimura, T. Mersenne twister: A 623-dimensionally equidistributed uniform pseudorandom number generator. *ACM Trans. Model. Comput. Simul.* **1998**, *8*, 3–30.
26. Hui, A.W.; Hamielec, A.E. Thermal polymerization of styrene at high conversions and temperatures. An Experimental study. *J. Appl. Polym. Sci.* **1972**, *16*, 749–769.
27. Gnuplot Home Page. Available online: <http://www.gnuplot.info> (accessed on 17 June 2012).
28. Drache, M.; Mandel, K.; Schmidt-Naake, G. Kinetics of nitroxide-controlled radical polymerization during the non-stationary state. *Polymer* **2007**, *48*, 1875–1883.
29. Gilbert, R.G. Critically-evaluated propagation rate coefficients in free radical polymerizations. I. Styrene and methyl methacrylate. *Pure Appl. Chem.* **1996**, *68*, 1491–1494.
30. Buback, M.; Kuchta, F. Termination kinetics of free-radical polymerization of styrene over an extended temperature and pressure range. *Macromol. Chem. Phys.* **1997**, *198*, 1455–1480.
31. Tang, W.; Fukuda, T.; Matyjaszewski, K. Reevaluation of persistent radical effect in NMP. *Macromolecules* **2006**, *39*, 4332–4337.
32. Wang, L.; Broadbelt, L.J. Tracking explicit chain sequence in kinetic Monte Carlo simulations. *Macromol. Theory Simul.* **2011**, *20*, 54–64.
33. Gperftools. Available online: <http://www.code.google.com/p/gperftools> (accessed on 18 June 2012).

34. Lutz, J.-F.; Matyjaszewski, K. Kinetic modeling of the chain-end functionality in atom transfer radical polymerization. *Macromol. Chem. Phys.* **2002**, *203*, 1385–1395.
35. Lutz, J.-F.; Matyjaszewski, K. Nuclear magnetic resonance monitoring of chain-end functionality in the atom transfer radical polymerization of styrene. *J. Polym. Sci. Part A Polym. Chem.* **2005**, *43*, 897–910.
36. Fu, Y.; Mirzaei, A.; Cunningham, M.F.; Hutchinson, R.A. Atom-transfer radical batch and semibatch polymerization of styrene. *Macromol. React. Eng.* **2007**, *1*, 425–439.
37. Najafi, M.; Roghani-Mamaqani, H.; Salami-Kalajahi, M.; Haddadi-Asl, V. A comprehensive Monte Carlo simulation of styrene atom transfer radical polymerization. *Chin. J. Polym. Sci.* **2010**, *28*, 483–497.
38. Ohno, K.; Goto, A.; Fukuda, T.; Xia, J.; Matyjaszewski, K. Kinetic study on the activation process in an atom transfer radical polymerization. *Macromolecules* **1998**, *31*, 2699–2701.
39. Matyjaszewski, K.; Patten, T.E.; Xia, J. Controlled/“living” radical polymerization. Kinetics of the homogeneous atom transfer radical polymerization of styrene. *J. Am. Chem. Soc.* **1997**, *119*, 674–680.
40. Barner-Kowollik, C.; Buback, M.; Charleux, B.; Coote, M.L.; Drache, M.; Fukuda, T.; Goto, A.; Klumperman, B.; Lowe, A.B.; Mcleary, J.B.; Moad, G.; Monteiro, M.J.; Sanderson, R.D.; Tonge, M.P.; Vana, P. Mechanism and kinetics of dithiobenzoate-mediated RAFT polymerization. I. The current situation. *J. Polym. Sci. Part A Polym. Chem.* **2006**, *44*, 5809–5831.
41. Konkolewicz, D.; Hawckett, B.S.; Gray-Weale, A.; Perrier, S. RAFT polymerization kinetics: Combination of apparently conflicting models. *Macromolecules* **2008**, *41*, 6400–6412.
42. Klumperman, B.; van den Dungen, E.T.A.; Heuts, J.P.A.; Monteiro, M.J. RAFT-mediated polymerization—A story of incompatible data? *Macromol. Rapid Commun.* **2010**, *31*, 1846–1862.
43. Meiser, W.; Buback, M. Assessing the RAFT equilibrium constant via model systems: An EPR study. *Macromol. Rapid Commun.* **2011**, *32*, 1490–1494.
44. Junkers, T.; Barner-Kowollik, C.; Coote, M.L. Revealing model dependencies in “assessing the RAFT equilibrium constant via model systems: An EPR study”. *Macromol. Rapid Commun.* **2011**, *32*, 1891–1898.
45. Dixon, K.W. Decomposition rates of organic free radical initiators. In *Polymer Handbook*, 4th ed.; Brandrup, J. Immergut, E.H., Grulke, E.A., Eds.; Wiley-Interscience: New York, NY, USA, 1999.
46. Buback, M.; Huckestein, B.; Kuchta, F.-D.; Russell, G.T.; Schmid, E. Initiator efficiencies in 2,2'-azoisobutyronitrile-initiated free-radical polymerizations of styrene. *Macromol. Chem. Phys.* **1994**, *195*, 2117–2140.
47. Buback, M.; Kurz, C.H.; Schmaltz, C. Pressure dependence of propagation rate coefficients in free-radical homopolymerizations of methyl acrylate and dodecyl acrylate. *Macromol. Chem. Phys.* **1998**, *199*, 1721–1727.
48. Buback, M.; Kuelpmann, A.; Kurz, C. Termination kinetics of methyl acrylate and dodecyl acrylate free-radical homopolymerizations up to high pressure. *Macromol. Chem. Phys.* **2002**, *203*, 1065–1070.

49. Dire, C.; Belleney, J.; Nicolas, J.; Bertin, D.; Magnet, S.; Charleux, B. β -hydrogen transfer from poly(methyl methacrylate) propagating radicals to the nitroxide SG1: analysis of the chain-end and determination of the rate constant. *J. Polym. Sci. Part A Polym. Chem.* **2008**, *46*, 6333–6345.
50. Nicolas, J.; Dire, C.; Mueller, L.; Belleney, J.; Charleux, B.; Marque, S.R.A.; Bertin, D.; Magnet, S.; Couvreur, L. Living character of polymer chains prepared via nitroxide-mediated controlled free-radical polymerization of methyl methacrylate in the presence of a small amount of styrene at low temperature. *Macromolecules* **2006**, *39*, 8274–8282.
51. Nicolas, J.; Mueller, L.; Dire, C.; Matyjaszewski, K.; Charleux, B. Comprehensive modeling study of nitroxide-mediated controlled/living radical copolymerization of methyl methacrylate with a small amount of styrene. *Macromolecules* **2009**, *42*, 4470–4478.
52. Willemsse, R.X.E.; van Herk, A.M. Copolymerization kinetics of methyl methacrylate-styrene obtained by PLP-MALDI-ToF-MS. *J. Am. Chem. Soc.* **2006**, *128*, 4471–4480.
53. Huijser, S.; Mooiweer, G.D.; van der Hofstad, R.; Staal, B.B.P.; Feenstra, J.; van Herk, A.M.; Koning, C.E.; Duchateau, R. Reactivity ratios of comonomers from a single MALDI-ToF-MS measurement at one feed composition. *Macromolecules* **2012**, *45*, 4500–4510.

© 2012 by the authors; licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution license (<http://creativecommons.org/licenses/by/3.0/>).