Parallel High-Performance Grid Computing: Capabilities and Opportunities of a Novel Demanding Service and Business Class Allowing Highest Ressource Effiziency

Nick KEPPER[a,b,c], Ramona ETTIG[c], Frank DICKMANN[d], Rene STEHR[e],
Frank G. GROSVELD[f], Gero WEDEMANN[e], and Tobias A. KNOCH[a,b,1]

[a]Biophysical Genomics, Dept. Cell Biology & Genetics, Erasmus MC,
Dr. Molewaterplein 50, 3015 GE Rotterdam, The Netherlands.
[b]Genome Organization & Function, BioQuant & German Cancer Research Center,
Im Neuenheimer Feld 267, 69120 Heidelberg, Germany.
[c]Kirchhoff Institute of Physics, University of Heidelberg,
Im Neuenheimer Feld 227, 69120 Heidelberg, Germany.
[d]Department of Medical Informatics, University of Göttingen,
Robert-Koch-Straße 40, 37075 Göttingen, Germany.
[e]System Engineering & Information Management, University of Applied Sciences
Stralsund, Zur Schwedenschanze 15, 18435 Stralsund, Germany.
[f]Dept. Cell Biology & Genetics, Erasmus MC, Dr. Molewaterplein 50,
3015 GE Rotterdam, The Netherlands.

Abstract. Especially in the life-science and the health-care sectors the huge IT requirements are imminent due to the large and complex systems to be analysed and simulated. Grid infrastructures play here a rapidly increasing role for research, diagnostics, and treatment, since they provide the necessary large-scale resources efficiently. Whereas grids were first used for huge number crunching of trivially parallelizable problems, increasingly parallel high-performance computing is required. Here, we show for the prime example of molecular dynamic simulations how the presence of large grid clusters including very fast network interconnects within grid infrastructures allows now parallel high-performance grid computing efficiently and thus combines the benefits of dedicated super-computing centres and grid infrastructures. The demands for this service class are the highest since the user group has very heterogeneous requirements: i) two to many thousands of CPUs, ii) different memory architectures, iii) huge storage capabilities, and iv) fast communication via network interconnects, are all needed in different combinations and must be considered in a highly dedicated manner to reach highest performance efficiency. Beyond, advanced and dedicated i) interaction with users, ii) the management of jobs, iii) accounting, and iv) billing, not only combines classic with parallel high-performance grid usage, but more importantly is also able to increase the efficiency of IT resource providers. Consequently, the mere “yes-we-can” becomes a huge opportunity like e.g. the life-science and health-care sectors as well as grid infrastructures by reaching higher level of resource efficiency.

Keywords. High-performance parallel computing, grid and GPU computing, molecular dynamics simulation, NAMD, e-Human Grid Ecology.

[1] Corresponding author: email: TA.Knoch@taknoch.org

Introduction

With the advancement of science and technology the requirements for IT are growing exponentially. Especially in the life-science and the health-care sectors the extreme demands are obvious due to system sizes and complexities [1, 2]. The human genome e.g. consists of ~7x10^9 base pairs storing ~1.75 GByte [3, 4]. Its huge complexity can be shown by its molecular length of 2 m and seven compactions levels to fit in small 10^{-5} m wide cell nuclei. Thus length and time scales from 10^{-9} - 10^{-4} m and 10^{-10} - 10^4 s are bridged. Estimates for molecular dynamics computational times assuming a tenfold CPU speed increase every five years suggest the simulation of a complete Escherichia coli bacterium (10^{11} atoms) on a nanosecond scale by 2034, a mammalian cell (10^{15} atoms) by 2056, and a complete human being (10^{27} atoms) in real time by 2172 [5]. Clever multi-scale approximations have allowed major insights in such huge systems, showing the feasibility and necessity of such approaches [3, 6].

Grid infrastructures are already major providers for the necessary computing resources in research, diagnostics, and treatment, due to their efficiency and cost effectiveness [7]. In the early days, grids were mainly optimized for huge number crunching of relatively trivial parallelizable problems [8], since i) different infrastructure types and ii) distributed resources can most easily be combined by straight forward middleware approaches. Also the job management, accounting, and billing, leaves a lot of freedom for different scenarios yielding high resource efficiency [7]. However, already online visualizations, where entire teams work telematically together, suggest higher demands [2]. Increasingly parallel high-performance computing is required and only statistical relevance is achieved trivially [3, 9, 10]. Parallel HPC has been and is still the classic domain of super-computing centres, despite that grid-like management by mixing trivial with non-trivial parallel jobs, lead to efficiencies above 99.9% [8]. Since grids like the MediGRID (German D-Grid) have clusters with several 10^3 or even 10^4 cores including very fast network interconnects, this allows now parallel high-performance grid computing efficiently. Therefore, high heterogeneous requirements for i) CPU, ii) memory, iii) storage, and iv) communication, need to be combined with advanced management of i) users, ii) jobs, iii) accounting, and iv) billing approaches. For the prime case of molecular dynamics simulations, we show here that combining trivial with parallel high-performance computing grids, beyond the mere “yes-we-can”, allows the maximum of grid efficiency to be reached. This is a huge opportunity e.g. for the life-science and health-care sectors as well as grid infrastructures.

1. Heterogeneous high-performance requirements for parallel applications

The hardware and software requirements for parallel applications depend on: i) the problem size and type, i.e. algorithmic mathematics/physics/chemistry/biology and the number particles/parameters, ii) the degree of parallelization possible, i.e. the optimal load balancing over different processors/memory, and iii) calculation/simulation type and size scenario, i.e. the input/output and storage amount. These determine the suitability of a hardware and the optimal software solution optimizing the hardware use. In general, computational time is the meta-parameter for optimization. The i) interaction with users, ii) the management of jobs, iii) accounting, and iv) billing, are getting increasingly important too and make further optimizations necessary. Dedicated hardware for parallel high-performance computing differs in i) the CPU amount and distribution, ii) the memory amount and distribution, iii) the storage architecture, and iv) the type and architecture of fast communication interfaces. Either the hardware is optimized for a special task and thus for any of these parameters, or the hardware covers a broad problem spectrum. The latter means that mostly a compromise has to be made and is consequently hardly ever optimal. This applies also to grid clusters if supercomputing centres have not donated their special machines. E.g. some algorithms work most effectively on shared memory architectures. Modern multi-core processors support with OpenMP up to 16 threads, vector computers and cell processors up to 64, and Graphics Processing Units (GPU) extent this to several 10^3 cores. Beyond, however, with increasing problem size distribution between processors gets inevitable. Here the communication between processors and the job management concerning waiting times for available processors are the main bottleneck.

The software to tackle a given life-science or health-care challenge has to translate the i) problem type and size, ii) algorithmic basis, and iii) the actual calculation or simulation scenario, within a given hardware optimally. More and more the problems dictate here already *a priori* a special hardware usage. The optimization challenge in grid infrastructure lies beyond optimization for a specific setting in the replacement of certain hardware features or their entire virtualization. The management of grid resources including accounting/billing features is here the final polishing.

Consequently, parallel high-performance grid computing has to combine the challenges of optimizing a dedicated problem within a heterogeneous hardware setting with the right management. By combination of small classic trivial grid jobs and with huge parallel jobs, CPU/core usage efficiencies > 99.9% can be reached [8].

2. DNA/nucleosome MD simulations on a parallel high-performance grid

One of the most computationally challenging tasks is the molecular dynamics (MD) simulation of DNA and the nucleosome with atomic resolution [3, 9, 11]. This first DNA packaging level consists of a histone protein octamer core to which ~208 DNA base pairs are associated - 146 bp bind directly and an additional histone functions as a "lock" [3, 9, 11]. The dynamics of DNA/nucleosomes plays an important role in all genomic processes. MD simulations are here very important to investigate the behaviour of each atom in the complex (e.g. the DNA-core interaction) of a simulated trajectory and thus to elucidate the conformation pathways that are hidden in most experiments [11]. In MD, for each atom a force field is approximated from quantum mechanical simulations. MD has atomic resolution, but covers only short time periods (~ns). For this kind of protein folding, docking, and binding problems several well established and highly optimized packages exist: NAMD [12], AMBER [13, 14], GROMACS [15] or CHARMM [16]. NAMD is also able simulate and measure energetic parameters by so called steered molecular dynamics (SMD) simulations. Especially biomolecule simulations in their natural water environment contain in SMD simulations enormous amounts of water molecules, sometimes tenfold the atoms in the molecule. With water and ions, the system is energetically minimized to avoid sterical clashes between atoms. Thereafter, the system has to be heated up. These preparations are already computationally very intensive before the productive MD simulations can be started and before the final outcome is analysed, which is most of time also nontrivial. Currently, 10^{-7} s for $\sim 10^5$ atoms are achievable on the largest parallel high-performance computers, and are thus optimized for distributed hardware.

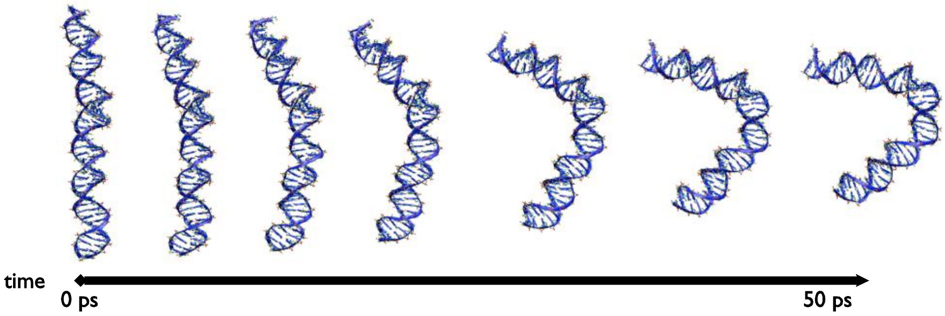

Figure 1. SMD simulation of a 44 base pair long DNA segment over 50 ps. The complete simulation was performed in a water box with ions, to neutralize the charges of the DNA (not visualized). In the middle of the DNA, an atom was selected and a pulling force (to the right) was applied resulting in bending.

2.1. Simulation of a 44 base pair long DNA sequence and entire nucleosomes

How DNA sequences behave under local stress, i.e. whether and how it is bending due to mechanical impacts or electrostatic fields e.g. during protein docking or binding, is of major importance [17, 18]. The typical simulation of a small 44 bp long DNA has together with the water shell and the ions (neutralizing the negative charges of the DNA) a size of ~8.5x10^4 atoms (Figure 1). To elucidate its rigidity, different SMD simulations are the prime choice and show in detail how pulling deforms and finally bends the structure (Figure 1). Different force fields result in important insights of more complex docking and binding scenarios as e.g. during nucleosome formation, the rearrangement during genetic processes, or the interaction of nucleosome remodelers, which move the nucleosome along the DNA strand [10, 17, 18].

To simulate a complete nucleosome increases the size and complexity since it is a complex of several interacting biomolecules: Simulating the 146 bp directly binding to the core involves ~3x10^4 atoms (Figure 2). To understand the internal nucleosome core dynamics, i) classic MD simulations with a smaller water shell and altogether ~2.5x10^5 atoms allow basic structural insights, whereas ii) SMD simulations with ~7x10^5 atoms are important to understand the dynamic histone protein-protein and protein-DNA interactions. Due to the increased motion in the latter case with applied external forces, the water shell has to be much larger. Without applied pulling forces (Figure 2), still motions occur, but the changes are not that prominent as in SMD simulations (Figure 1), and thus would not represent the real biological processes and thus functions adequately. The simulations have e.g. shown the detailed changes and dynamics in nucleosomes up to 50 ps and that these are much less prominent compared to that of the DNA, i.e. that the nucleosome core is a very stable structure [10, 17, 18].

2.2. Parallelity and efficiency of MD/SMD simulations

Parallel programmes never scale perfectly (communication, serial algorithm parts, etc.). The speedup measures the parallelity while comparing different CPU/core numbers (Figure 3). If e.g. the algorithms profit from more cache memory provided by the additionally used hardware, the scaling can be hyper-linear (Figure 3, JUMP cluster with 64 cores). The speedup is strongly depending on the used hardware. For our MD

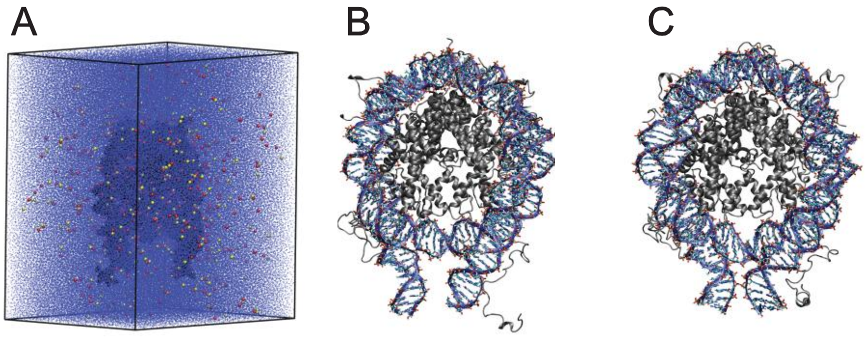

Figure 2. A) Nucleosome (black) in a box with water molecules (blue) and ions (red/ yellow) in a so called VDW visualization. B) Start structure of a nucleosome for an MD simulation (water/ions are not shown). The DNA (blue) is wrapped around the histone core (gray). C) Same nucleosome as in B) but after 20 ns of simulation. Changes in the linker DNA and histone tails are clearly visible.

nucleosome simulation with about ~2.5x10^5 atoms, the speedup was tested at D-Grid resources: the JUMP (IBM Power6 575) and the Jugene (IBM Blue Gene/P with 73,728 compute nodes) clusters, both at the Supercomputing Center Jülich, and the HLRN-II cluster (a SGI Altix ICE 8200 system) of the North German Supercomputing Alliance. The optimum seems to be 512 cores at the HLRN-II cluster (Figure 3, Table 1). Comparing the speedup gives an impression of the overall efficiency. For 512 cores the HLRN-II cluster is more efficient than the Jugene. Concerning the system size, the SMD simulation of DNA have ~12% of the atoms than that of the entire nucleosome, but uses only 7% of the time (assuming a linear speedup, HLRN-II). Additionally, the 100 GByte produced by MD with 10^5 atoms and 50 ns also influences speedups, but due to the embedding in the German D-Grid performs especially well. Again filling the waiting times for processors and even running a second thread on underused cores results in meta efficiencies of >99.9% using location dependent approaches [8].

3. Management of users and jobs as well as accounting and billing

Grid infrastructures obviously provide great opportunities to access computing resources. Parallel high-performance grids now need to combine the necessary optimization process concerning the technical hardware requirements, i.e. i) CPU, ii) memory, iii) storage, and iv) communication, with the classic grid challenges in an even more advanced form: i) interaction with users to provide best technical support, ii) the management of jobs to allow meta-efficiency of the hardware >99.9% by combining trivial with real parallel jobs, iii) accounting taking into account the special capacities of parallel applications and aggregation over multi-core distributed jobs and balancing with meta-efficiency increasing “filler” jobs, and iv) billing with appropriate customer value propositions in relation to the accounting. In respect to user goals this is not trivial and requires that this integration must be done especially well to exploit the hardware optimally. Here the egoistic goals of the users conflict clearly with that of the super computer centre/grid looking for overall performance efficiency over several users. Proper combination of the management with the technical layer can here result in reaching the maximum overall efficiency >99.9% of the hardware:

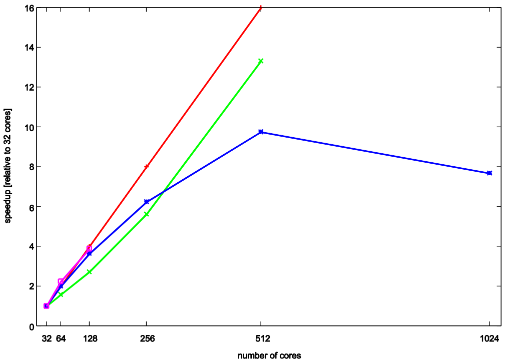

Figure 3. Speedup of an MD simulation of an entire nucleosome with 2.5x10^5 atoms at different clusters. The speedup for all clusters was normalize for 32 cores as 1. Linear scaling is under normal conditions the optimum (red). The speedup depends besides the used algorithm mainly on the hardware (HLRN-II: green; Jugene: blue; JUMP: purple) and shows quite big differences which are important for the hardware choice.

3.1. User and job management in parallel high-performance grid environments

The maximum resource efficiency in parallel high-performance infrastructures can only be reached by advanced user management, due to the heterogeneity mentioned. This involves: i) thorough information about the grid resources and their technical capabilities both for trivial and non-trivial parallel jobs, ii) education on parallelization in general and the simultaneous optimization of applications towards several dedicated hardware, and iii) technical and psychological training how trivial and real parallel jobs run in the grid and how this influences the overall meta-performance of the system. Whereas the first two require dedicated training approaches, the third is most import and often very hard to achieve: People i) have to understand trivial and real parallel approaches simultaneously, although they might only use one category at a time, and ii) they also have to understand the general concepts other people apply in the grid to adjust their own applications and job management. On a technical level, that can be shielded by dedicated scheduling systems setup by the resource provider, who then faces the same challenges internally. Nevertheless, while nearing in on the maximum efficiency, also scheduling systems get more complex and a certain application might not run optimally anymore. Thus, the will to reduce own egoistic goals in favour of overall performance increase requires altruistic behaviour, which is a complex psychological and socio-cultural challenge to be tackled by e-Human Grid Ecology [7].

For the workflow management this has severe consequences: The heterogeneity of the clusters in the grids necessitates identification of potential clusters first. Selecting the most efficient number of processes and cores also requires extensive testing. This can then be used for production and similar simulation types (e.g. nucleosomes with different DNA sequences) and the size. Thereafter, the scheduling system and the “filler” applications must be in detailed analysed and optimizations of the own and other applications have to be considered and implemented. Finally, all this information has to be internalized in a portal like system managing the running of the jobs within the parallel high-performance grid infrastructure. This system also needs to be able to monitor job progress and control in- as well as outputs to guaranty proper comparable results. With such an approach we were able to reach >99.9% resource meta-efficiency, and thus were able to optimize grid usage to its maximum.

Table 1. Time to calculate 1ns for different systems on different clusters with best speedup.

MD system	Cluster	Nodes	Time [h]
DNA (8.5x10^4 atoms, SMD)	svahe	32	36:13
	HLRN-II	64	12:05
Nucleosome (2.5x10^5 atoms, MD)	HLRN-II	512	3:20
	Jugene	512	2:54
	Jump	128	7:12
Nucleosome (7x10^5 atoms, SMD)	HLRN-II	512	21:01

3.2. Accounting and billing

The accounting and billing in parallel high-performance grids also needs elevated strategies, due to i) the information aggregation over different cores, and ii) the more complex accounting and price structure. During the waiting time for the necessary amount of cores, small parallel or trivial filler jobs can be executed. This is very similar but more complex than for online and distributed grid visualization approaches [2]. For billing, this suggests a variety of opportunities, where the price can be coupled to the accounting via an integrative model with weighted scenarios. Vice versa also the prizing and thus the business model can be used to put pressure on the user to optimize his application especially concerning the provider goal to run the resource at the optimum. This might be also a way to ease the challenge of technical and psychological training while optimally combining trivial with nontrivial parallel jobs.

Conclusion – beyond the “yes-we-can”

In the life-science and health-care sectors demanding IT and especially large scale grid resources are needed to analyze huge and complex systems. We showed here, how the presence of large grid clusters within grid infrastructures allows now parallel high-performance grid computing efficiently and thus combines the benefits of dedicated super-computing centres and grid infrastructures. Therefore, hardware wise the i) CPU, ii) memory, iii) storage, and iv) communication, must be considered in detail and combined optimally with advanced and dedicated i) interaction with users, ii) the management of jobs, iii) accounting, and iv) billing. Thus, we combined classic with parallel high-performance grid usage, while increasing the efficiency of IT resource providers. This is beyond the mere “yes-we-can” a great opportunity for the life-science and health-care sectors and allows reaching the maximum resource efficiency.

Acknowledgements

Our special thanks go to Karsten Rippe for stimulating discussions, and the Erasmus Medical Centre, the BioQuant Centre/German Cancer Research Centre, the German MediGrid and German D-Grid Initiative, the Erasmus Computing Grid, the Supercomputing Centre Karlsruhe (SCC; grant ChromDyn), North German Supercomputing Alliance (HLRN; mvb00007), and the Supercomputing Centre Jülich for infrastructure access. This work was partly supported by the Bundesministerium für Bildung und Forschung (BMBF) under grants # 01 KW 9602/2 (Heidelberg 3D Human Genome Study Group, German Human Genome Project), # 01 AK 803 A-H (German

MediGRID), # 01 IG 07015 G (German Services@MediGRID), # 01 G 07016 A (MedInfoGRID), and # 01 IG 07014 (D-Grid Integration Project).

References

[1] T. Solomonides, M. Hofmann-Apitius, M. Freudigmann, S. C. Semler, Y. Legré, and M. Kratz, *Healthgrid research, innovation and business case - Proceedings of HealthGrid 2009*. IOS Press, Amsterdam, ISBN 978-1-60750-027-8, 2009.

[2] F. Dickmann, M. Kaspar, B. Löhnhardt, N. Kepper, F. Viezens, F. Hertel, M. Lesnussa, Y. Mohammed, A. Thiel, T. Steinke, J. Bernarding, D. Krefting, T. A. Knoch, and U. Sax. Visualization in health grid environments: a novel service and business approach. *Stud. Health Technol. Inform.* **147** (2009), 150-159.

[3] T. A. Knoch *Approaching the three-dimensional organization of the human genome: structural-, scaling- and dynamic-properties in the simulation of interphase chromosomes and cell nuclei long–range correlations in complete genomes, in vivo analysis of the chromatin distribution construct conversions in simultaneous co–transfections*. Ruperto Carola University, Heidelberg, Germany, and TAK-Press, Dr. Tobias A. Knoch, Mannheim, Germany, ISBN 3-00-009960-3, 2002.

[4] T. A. Knoch, M. Lesnussa, N. Kepper, H. Eussen, and F. G. Grosveld. The GLOBE 3D Genome Platform: towards a novel system-biological paper tool to integrate the huge complexity of genome organization and function. *Health Technol. Inform.* **147** (2009), 105–116.

[5] W. F. van Gunsteren, D. Bakowies, R. Baron, I. Chandrasekhar, M. Christen, X. Daura, P. Gee, D. P. Geerke, A. Glättli, P. H. Hünenberger, M. A. Kastenholz, C. Oostenbrink, M. Schenk, D. Trzesniak, N. F. A. van der Vegt, and H. B. Yu Biomolecular Modeling: Goals, Problems, Perspectives. *Angewandte Chemie* **45** (2006), 4064-4092.

[6] S. Jhunjhunwala, M. van Zelm, M. Peak, S. Cutchin, R. Riblet, J. van Dongen, F. G. Grosveld, T. Knoch, and C. Murre. The 3D structure of the Immunoglobulin Heavy–Chain locus: implications for long–range genomic interactions. *Cell* **133(2)** (2008), 265–279.

[7] T. A. Knoch, V. Baumgärtner, L. V. de Zeeuw, F. G. Grosveld, and K. Egger e-Human Grid Ecology - understanding and approaching the inverse tragedy of the commons in the e-Grid society. *Stud. Health Technol. Inform.* **147** (2009), 269-276.

[8] T. A. Knoch, M. Göker, R. Lohner, A. Abuseiris, and F. G. Grosveld. Fine-structured multi-scaling long-range correlations in completely sequenced genomes--features, origin, and classification. *Eur. Biophys. J.***38(6)**, 757-779.

[9] N. Kepper, D. Foethke, R. Stehr, G. Wedemann, and K. Rippe Nucleosome geometry and internucleosomal interactions control the chromatin fiber conformation. *Biophys. J.* **95** (2008), 3692-3705.

[10] K. Rippe, A. Schrader, P. Riede, R. Strohner, E. Lehmann, and G. Längst, DNA sequence- and conformation-directed positioning of nucleosomes by chromatin-remodeling complexes. *Proc. Natl. Acad. Sci. USA* **104** (2007), 15635-15640.

[11] K. Luger, A. W. Mäder, R. K. Richmond, D. F. Sargent, and T. J. Richmond Crystal structure of the nucleosome core particle at 2.8 Å resolution. *Nature* **389** (1997), 231-233.

[12] J. C. Phillips, R. Braun, W. Wang, J. Gumbart, E. Tajkhorshid, E. Villa, C. Chipot, R. D. Skeel, L. Kale, and K. Schulten Scalable molecular dynamics with NAMD. *J. Comput. Chem.* **26** (2005), 1781-1802.

[13] A. D. Case, T. E. Cheatham, I. I. I. T. Darden, H. Gohlke, R. Luo, K. M. Merz Jr., A. O. Onufriev, C. Simmerling, B. Wang, and R. J. Woods The Amber biomolecular simulation programs. *J. Comput. Chem.* **26** (2005), 1668-1688.

[14] W. D. Cornell, P. Cieplak, C. I. Bayly, I. R. Gould, K. M. Merz Jr., D. M. Ferguson, D. C. Spellmeyer, T. Fox, J. W. Caldwell, and P. A. Kollman A second generation force field for the simulation of proteins, nucleic acids, and organic molecules. *J. of the Am. Chem. Soc.* **117** (1995), 5179-5197.

[15] B. Hess, C. Kutzner, D. van der Spoel, and E. Lindahl GROMACS 4: Algorithms for Highly Efficient, Load-Balanced, and Scalable Molecular Simulation. *J. Chem. Theory Comput.* **4** (2008), 435-447.

[16] A. D. MacKerell Jr., N. Banavali, and N. Foloppe Development and current status of the CHARMM force field for nucleic acids. *Biopolymers* **56** (2000), 257-265.

[17] Wedemann, G., and Langowski, J. (2002). Computer simulation of the 30-nanometer chromatin fiber. Biophys J *82*, 2847-2859.

[18] R. Stehr, N. Kepper, K. Rippe, and G. Wedemann, The Effect of Internucleosomal Interaction on Folding of the Chromatin Fiber. *Biophys. J. 95* (2008), 3677-3691.