

# A Multiscaling Constant Lambda Molecular Dynamic Gromacs Implementation

NICOLAE GOGA<sup>1\*</sup>, STEFANIA COSTACHE<sup>2</sup>, SIEWERT MARRINK<sup>1</sup>

<sup>1</sup>University of Groningen, Faculty of Natural Science, Nijenborgh 4, 9747 AG Groningen, Netherlands

<sup>2</sup>Politehnica University of Bucharest, Faculty of Automatic Control and Computers, 313 Splaiul Independentei, Bucharest, 060042, Romania

*Molecular dynamics is one of the methods used now-a-days by the scientific community to study the property of polymers. This paper presents a new method for multiscaling molecular dynamics that combines the advantages of fine-grained and coarse grained representations. The new methodology is implemented in the package Gromacs for molecular dynamics, a world-wide used software that achieves the best performance on single processors. The article presents the methodology, its implementation in the Gromacs package for molecular simulation and performance results.*

*Keywords: polymers, molecular dynamics, Gromacs, multiscaling modeling*

Molecular dynamics indicates the general process of describing complex chemical systems in terms of a realistic atomic model, with the aim to understand and predict macroscopic properties based on detailed knowledge on an atomic scale. Molecular dynamics (MD) is one of the methods used now-a-days by the scientific community to study the properties of polymers [5, 15, 12, 16, 2]. Such methods next to other that are developed [3, 10] are used as complementary to the laboratory experiments [7, 13, 14] for advancing the knowledge in the field of plastic materials.

A molecular system can be described with a fine-grained representation that is a detailed, low-level model of it. A coarse-grained representation of a molecular system is a model where some of these fine details have been smoothed over or averaged out [17, 18]. In MD, coarse graining consists in replacing a fine grained description of a molecular system such as polyethylene, for example, with a lower-resolution coarse-grained model that averages or smooths away fine details. The advantage of using coarse grained representations is the fact that it speeds-up the simulations. The price paid is the fact that some of the results obtained have a larger error margin than in the case of using FG models or sometimes even some macroscopic phenomena under investigation are not observed in the simulations.

There are recent efforts [6, 11] to combine the advantages of the two simulations in a single multiscaling simulation. In such a simulation a molecular system has a double nature, being modeled in a coarse grained representation while the fine grain details are represented proportionally with a scaling factor  $\lambda$ . In the Christen approach,  $\lambda$  is a constant during the simulation while for Praprotnik,  $\lambda$  depends on the coarse grain coordinates of each particle. Nevertheless, all the proposed approaches have limitations. In the model proposed by Christen for constant lambda, in a pure coarse grained simulation for keeping the fine grained particles together, Christen computes (bonded) forces between the FG particles and the only ones that are not computed being the non-bonded forces between unconnected atoms. In this case, next to the addition in the complexity and computational time, pure CG is no longer pure. Praprotnik

approach for space lambda computes only forces between two particles. Its multiscaling model is applicable only for simple systems, such as butane, with only one coarse grain particle corresponding to an assembly of 4 fine grained atoms. For more complex systems with more than one CG particle, such as polycarbonate for example, systems of each movements need the computation of the forces with the contribution of 3 particles (angle forces) or 4 (dihedral forces), Praprotnik modeling cannot be applied. Moreover, Praprotnik model does not follow a very systematic approach for the force computation that usually starts from the Hamiltonian.

The MD group of the University of Groningen started a research to improve the existing methods. In this paper we summarize the theoretical approach for constant lambda that solves the problem mentioned for Christen approach and we discuss its implementation in Gromacs package for molecular simulation. In the final Section we present the related performance issues.

## Experimental part

In the beginning of this section we shall describe the multiscaling formulas related to the Newtonian movement of particles starting from the Hamiltonian and after that we shall discuss different aspects related to the temperature coupling. At the end of the section we shall present the implementation of this model in the Gromacs 4.0 package for molecular dynamics.

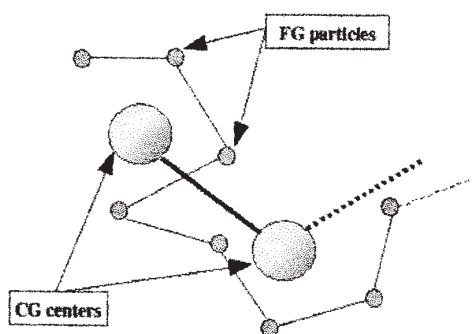


Fig. 1. FG particles and their correspondent CG centers

\* email: n.goga@rug.nl; Tel: (+31)-64-3634327

### Movement of the particles

Usually, in the existing literature for multiscale modelling, the particles at a given time are represented by their coordinates  $(R_i^{CG}, r_{ik}^{FG})$  where  $R_i^{CG}$  represent the coordinates of the CG particles and  $r_{ik}^{FG}$  represent the coordinates of the correspondent FG particles. Another possible view is to have the FG positions as relative to the correspondent CG particles. In this view, the multigrained system is rewritten as  $(R_i^{CG}, s_{ik}^{FG} = r_{ik}^{FG} - R_i^{CG})$  where  $s_{ik}^{FG}$  represent the relative positions of the FG particles corresponding to the CG particle with coordinates  $R_i^{CG}$ .

The advantage of having relative positions for the FG particles resides in the fact that relative positions imply relative velocities for the FG particles. Anticipating the discussion of the temperature coupling section, it is easy to observe that in the case of a pure CG simulation, the relative FG velocities should be zero (there is no FG nature for a pure CG simulation) and for a full FG simulation the relative velocities should have their normal values. We will come back to this in the paragraph related to temperature coupling.

As mentioned above the system is represented by

$$(R_i, s_{ik}^{FG} = r_{ik}^{FG} - R_i^{CG}) \quad (1)$$

and the constraints

$$\sum_{ik} m_{ik} * s_{ik} = 0 \quad (2)$$

where  $m_{ik}^{FG}$  represent the masses of the FG particles and the number of constraints equals the number of CG particles multiplied with 3 (the number of coordinates  $(x, y, z)$ ). Usually (also in [6]) the positions of the CG particles are computed as the positions of the centers of mass of their correspondent FG particles. For our model this is not longer the case: the movements of the CG particles and their correspondent FG complexes are correlated only through the constraints. This implies that

$$\sum_{ik} m_{ik}^{FG} * v_{s_{ik}}^{FG} = \frac{\partial 0}{\partial t} = 0 \quad (3)$$

where  $v_{s_{ik}}^{FG}$  is the relative movement to the CG center of an FG particle. This relation is needed in the computation of the kinetic energy and it has the meaning that the mass averaged relative movement of the FG particles to the correspondent CG position is always zero. The kinetic energy is given by the movements of the FG particles and it is computed as:

$$\begin{aligned} E_c &= \frac{1}{2} \sum_i \sum_{ik} m_{ik}^{FG} * (v_{ik}^{FG})^2 = \\ &= \frac{1}{2} \sum_i \sum_{ik} m_{ik}^{FG} * \left( \frac{\partial (R_i^{CG} + s_{ik}^{FG})}{\partial t} \right)^2 \end{aligned} \quad (4)$$

Taking into account (3) it results that

$$\begin{aligned} E_c &= \frac{1}{2} \sum_{ik} m_{ik}^{FG} * V_i^{CG} + \frac{1}{2} \sum_i \sum_{ik} m_{ik}^{FG} * (v_{s_{ik}}^{FG})^2 \\ &= E_c^{CG} + E_c^{FG} \end{aligned} \quad (5)$$

The formula from (5) shows that the kinetic energy is the sum of the kinetic energy of the CG particles and the relative kinetic energy of the FG particles. The other term of the Hamiltonian is given by the interaction

energies. The integration of energies is done in the following way

$$\begin{aligned} U_{mult} &= \lambda * U^{FG}(r_{ik}^{FG}) + (1 - \lambda) * U^{CG}(R_i^{CG}) \\ &= \lambda * U^{FG}(R_i^{CG} + s_{ik}^{FG}) + (1 - \lambda) * U^{CG}(R_i^{CG}) \end{aligned} \quad (6)$$

The computation of the forces for CG and FG particles are given by

$$\begin{aligned} F_{mult}^{CG} &= \frac{\partial U_{mult}}{\partial R_i^{CG}} \\ F_{mult}^{FG} &= \frac{\partial U_{mult}}{\partial s_{ik}^{FG}} \end{aligned} \quad (7)$$

From (6) it results that the formulas of the forces are computed in the following way:

$$\begin{aligned} F_{mult}^{CG}(R_i^{CG}) &= \lambda * \sum_{ik} F_{ik}^{FG}(R_i^{CG} + s_{ik}^{FG}) + (1 - \lambda) * F^{CG}(R_i^{CG}) \\ F_{mult}^{FG}(s_{ik}^{FG}) &= \lambda * F_{ik}^{FG}(R_i^{CG} + s_{ik}^{FG}) \end{aligned} \quad (8)$$

Once the forces are computed, the velocities and positions for both FG and CG particles can be computed in every simulation step by using a scheme similar with, for example, the Verlets scheme [1, 8]. The part that remains to be solved is the one related to the constraints. The constraints can be rephrased in the following way: for every simulation step the position of the center of mass (CM) of an FG sub-system of particles should be equal with the position of the correspondent CG particle. The position and the velocity of the CM can be computed in the following way.

$$\begin{aligned} R_i^{CM} &= \frac{1}{M_i} * \sum_{ik} m_{ik}^{FG} * (R_i^{CG} + s_{ik}^{FG}) \\ V_i^{CM} &= \frac{1}{M_i} * \sum_{ik} m_{ik}^{FG} * (V_i^{CG} + v_{s_{ik}}^{FG}) \end{aligned} \quad (9)$$

where  $M_i$  is the mass of the CG particle that is equaling the sum of the FG particles masses. It can well be the case that, after the computation of the new CG positions and FG relative positions, the CM differs from CG. Solving the constraints results in

$$\begin{aligned} s_{ik}^{FG} &= s_{ik}^{FG} + (R_i^{CG} - R_i^{CM}) \\ v_{s_{ik}}^{FG} &= v_{s_{ik}}^{FG} + (V_i^{CG} - V_i^{CM}) \end{aligned} \quad (10)$$

The next problem that needs to be solved is the temperature coupling, or how the relative velocities take into account the scaling of the forces. This will be discussed in the following section.

### Temperature Coupling

When varying  $\lambda$ , parameter that gives the degree of multiscale (or FG and CG natures), the phenomena that are present are the following:

- the velocities of the CG particles are not influenced by the scaling with  $\lambda$ . There will be always a move of the CG particles (that represents the centers of mass of the correspondent FG particles), movement that is present for full CG or FG simulations, or mixed ones;
- the relative velocities are influenced by the scaling with  $\lambda$ . At full CG simulation ( $\lambda$  is 0), because there is no FG presence, the relative velocities should be 0 while, for a full FG simulation ( $\lambda$  is 1), the relative velocities

should have their normal value. For a  $\lambda$  between 0 and  $\lambda$  the relative velocities should have a value that is proportional with the  $\lambda$ -scaling of the forces.

For modeling 2), let us concentrate on the case of a pure CG simulation ( $\lambda$  is 0). Having no FG nature means to have no forces acting between the FG particles and no relative velocities. The scaling of the forces with 0 solves the first condition, namely no forces acting between FG particles. Moreover, if there is no initial momentum present, having no force will maintain the relative velocity to zero. But in the computation of the relative velocities the influence of the reference temperature is always taken into account. If there will be an initial momentum for the FG particles and the temperature is maintained as usual for both CG and FG particles, because there is no force between the FG particles to regulate their movement, the FG velocities might increase and in few steps the system can explode. Similar discussion can be done for small values of  $\lambda$ . For solving this problem our solution is to couple the CG and the FG representations to two separated temperatures buffers and to scale the reference temperature for the FG buffer with  $\lambda$ . This means that for a pure CG simulation the reference temperature for the FG details is 0 and in this way the relative velocities are scaled to 0 even if there is an initial momentum. In this way it can be shown that the Boltzmann factor is the same for both FG and CG representations. Having this approach will imply that the computation of the bonded forces between particles are not needed for pure CG, as in the case of Christen approach.

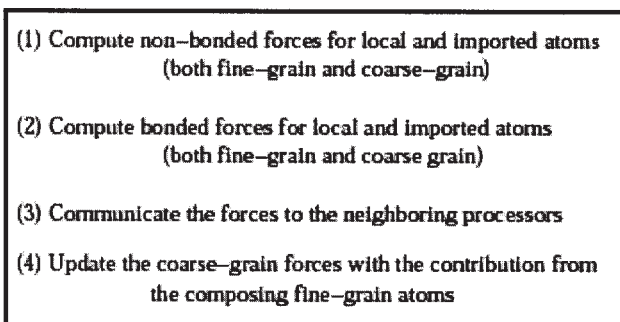
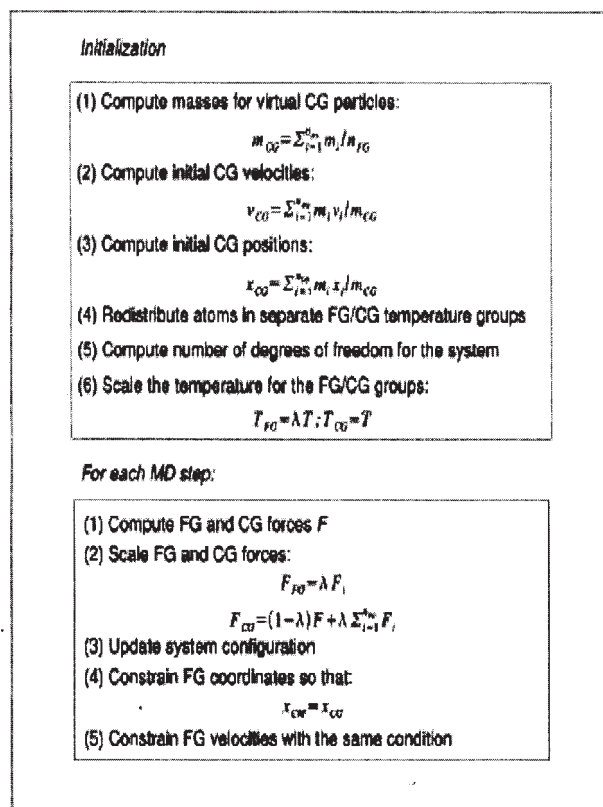
#### Implementation in Gromacs 4.0

GROMACS1 (GRONingen MACHine for Chemical Simulations) is an extensive, well-established and free software package used in Molecular Dynamics simulations [8]. It is developed in C, using MPI (Message Passing Interface) [9] for the parallel model implementation, and it is mainly used under Linux. It has been developed in the early 1990s at Groningen University and the actual version GROMACS 4.0 contains a high degree of parallelism. Nowadays, there are several comparable (mostly commercial) systems, e.g. Charmm, Tinker, NAMD and Gromos; however, GROMACS achieves the best performance (on single processors, see [4]). GROMACS is widely used (by more than 150 universities, research institutions and companies all over the world), e.g. for the simulation of biological processes at the cellular level, polymers, the design and testing of detergents and pharmaceutical drugs etc. The natural solution for implementing the developed multiscale methodology was to do it in Gromacs. This will also assure at the end a large community of users.

When implementing the constant lambda multiscale method in Gromacs we should take in consideration that MD simulations are very resource consuming: running time is months in average and they run on multiple processors. As mentioned earlier, we choose to implement this model in Gromacs version 4.0 that shows a better scalability than the previous versions of Gromacs. In the next figure we represent the main algorithm.

The computation of the forces is detailed in the following figure.

We used two types of storages for the two representations: one in which FG and CG parameters were kept in two separated files (topologies in Gromacs) and another solution in which the configuration of the



FG and CG parameters was kept in one file. Both solutions have their own advantages and disadvantages. As we will see in the next section, having one topology for both representations speeds up the simulation. The disadvantage is that some FG and CG parameters such as cut off or reaction field must be the same for both representations, which is not always desirable. Therefore, the implementation with two separated topologies, even if it might slow down the execution, might be still more desirable for some simulations because it assures two separated sets of parameters.

#### Results and discussions

For investigating the scalability of the implementation, the measurements were performed on an IBM Blue Gene/L architecture with all the simulations ran in the Co-processor mode.

We simulated two systems: one of butane having 13500 FG atoms and 3375 CG particles and one of Hexadecane with 8192 FG atoms and 2048 CG particles. The simulated time per step is 1 fs (1000 femtoseconds = 1 picoseconds), the temperature was kept constant at 300 K and we varied the  $\lambda$  parameter.

First, we analyzed the difference of performance between the multiscale model, using two separate simulation spaces, and the model with a single simulation space. The results can be viewed in table 1. For choosing an appropriate distribution of the processors for the FG



and CG simulations, we have taken in consideration the design of the CG model that we used. In the end, we kept the fine-grain/coarse-grain ratio to 4:1, meaning that for one processor that handles the coarse-grain simulation, four processors are allocated for the fine-grain part. However, even with a proper ratio, having two simultaneous simulations is not so efficient due to the overhead imposed by the global communication between the two simulations and the two topologies for the same system.

**Table 1**

COMPARISON BETWEEN THE TIMINGS OBTAINED FOR MULTISCALING WITH ONE AND TWO TOPOLOGIES

Nr. procs (4:1)	Single Simulation (ms/step)	Coupled simulations (ms/step)
5	27.25	25.97
10	14.7	15.26
20	7.86	12.38

Next, we shall discuss how multiscaling a simulation at two levels of details affects the overall performance of the MD simulator. We ran a normal fine-grain simulation on the HD system, using the original Gromacs MD simulator, and a multiscale one with additional coarse-grain particles, using our modified MD simulator.

**Table 2**

DIFFERENCE OF PERFORMANCE BETWEEN A NORMAL AND A MULTISCALE SIMULATION FOR THE HEXADECANE SYSTEM

Nr. procs	Normal simulation (ns/day)	Multiscale simulation (ns/day)
1	1.96	1.47
4	7.03	5.35
8	14.12	10.70
16	28.00	20.64
24	37.00	28.00
32	53.16	37.80
128	160.00	122.60

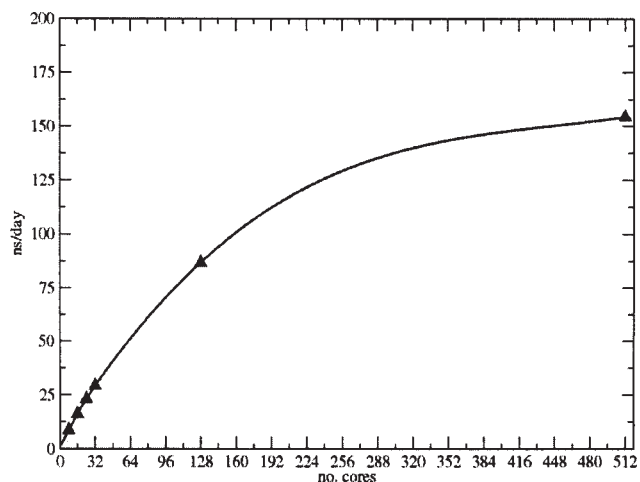


Fig. 2. Performance of the multiscale model (single topology) for a Butane system

As noticed in table 2, the overhead imposed by the increased volume of data communicated (information about coarse-grain particles must also be known) and

the computational steps required by the multiscale simulation makes itself noticed and produces a performance degradation up to 25%, but even so, the multiscale simulation manages to scale nicely up to the maximum number of processors that we have used. This is quite expected because having a simulation at two separate levels of representation leads to a topology with an increased number of particles (25% particles in plus) and interactions between them. Also, we have to take in consideration the correction and computation steps required by our multiscale simulation.

Finally in figure 2 we present the scaling up to a couple of hundred processors of our multiscale model for the Butane system. The butane system reports a linear scalability till 128 processors. After 200 processors the gain is not that much for this size of butane system. (The final simulated run for this system was for 512 processors).

## Conclusions

Molecular dynamics (MD) is one of the methods used now-a-days by the scientific community to study the properties of polymers. A fine-grained description of a molecular system is a detailed, low-level model of it. A coarse-grained description is a model where some of this fine detail has been smoothed over or averaged out. Recently there are efforts to combine the advantages of the two simulations in a single multiscaling simulation in which a system is representing as having double nature, FG and CG, and the integration of the FG nature is done according to a scaling factor  $\lambda$ . The models presented in the literature have limitations.

For solving these limitations we proposed a special representation in which the FG positions are computed as relative to the correspondent CG centers. Based on this representation we show how to compute the Newtonian equations of motions for a multiscaling algorithm with constant lambda. In this paper we discussed the implementation of this algorithm in the Gromacs package for molecular simulations, world-wide used software started in the MD group of Groningen that also achieves the best performance in the world for the runs on single processors. We choose to implement it in the last version 4.0 that has a good scalability over the number of processors used.

The runs on the IBM Blue Gene/L system showed a linear scalability till around 128 processors, the gain decreasing after 200 processors for a system of butane consisting of 13500 FG atoms and 3375 CG particles. For a simulated HD multiscaling system, the losses compared with a system just in a FG representation were lower than 25%. For the solution in which FG and CG parameters were kept in two separated topologies, the best ratio for the computational nodes turned out to be 4 processors for FG computations compared with 1 processor per CG computation.

As future work we plan to extend this model for pressure coupling and constraint solving for FG and CG representations. The model should be extended also for a multiscaling space modeling in which the scaling factor  $\lambda$  depends on the positions of the CG particles. We plan to implement all this extensions in Gromacs 4.0 and to do performance analysis for the implementations.

*Acknowledgments* We want to thanks Herman Berendsen, Alex de Vries, Andrzej Rzepiela for helping us developing the model, useful discussions and support in the implementation in Gromacs 4.0.

## Bibliography

1. BERENDSEN, H. J. C., *Simulating the Physical World* (Cambridge University Press, Cambridge, UK, 2007)
2. BHARADWAJ R.K., BERRY R.J., FARMER B.L., POLYMER, 41, 19, 2000 , p. 7209
3. BOLCU D., STANESCU, M.M CIUCA, I., JIGA, G., GHEORGHIU, H., *Mat. Plast.*, 45, nr. 2, 2008, p. 137
4. BOWERS, J. K., CHOW, E., XU, H., DROR, R.O., EASTWOOD, M. P , GREGERSEN, B. A., KLEPEIS, J. L., KOLOSSVARY, I., MORAES, M. A., SACERDOTI, F. D. , SALMON, J. K., SHAN, Y., SHAW, D. E., Scalable algorithms for molecular dynamics simulations on commodity clusters. In SC'06: Proceedings of the 2006 ACM/IEEE conference on Supercomputing, 2006, p. 84
5. BULACU, M. Molecular dynamics studies of polymer chains. Phd thesis, University of Groningen, 2007
6. Christen, M. and Christen, W. van, *J. Comput. Chem.* 124(15), 154106 (2006)
7. CRACIUNESCU, O., LUNGU, M., ZARNESCU, O., GASPAR, A., MOLDOVAN, L. *Mat. Plast.*, 45, nr. 2, 2008, p. 163
8. \*\*\* Gromacs manual, available online at [www.gromacs.org](http://www.gromacs.org)
9. \*\*\* MPI - Message Passage Interface Standard available online at <http://www.mpi-forum.org>
10. POPESCU, D., DOBRESCU, R., *Mat. Plast.*, 45, nr. 2, 2008, p. 158
11. PRAPROTNIK, M., K. PRAPROTNIK, K., SITE, L. D., *Physical Review E* 75, 017701 (2007)
12. SMITH, J.S., BEDROV, D. SMITH D., *Composites Science and Technology*, 63, 11, 2003, p.1599
13. STELIAN, V., VLAD, A., *Mat. Plast.*, 45, nr. 2, 2008, p.172
14. SUNEL, V., POPA, M., STOICAN, A.D., POPA, A.A., UGLEA, C.V. *Mat. Plast.*, 45, nr. 2, 2008, p.149
15. YANG,H., SHENG,L.S., QIAN, H., YANG, Y., ZHANG, X., SUN, C., *Polymer*, 45, 2, 2004, p. 453
16. YU, K., LI, Z., SUN, J. *Polymer Structures and Glass Transition, Macromolecular Theory and Simulations*, 10, 6, 2001, p. 624
17. PAUN, V.-P., *Mat. Plast.* , 43, nr.1, 2006. p.57
18. PAUN, V.-P., *Mat. Plast.* , 57, nr.2, 2006. p.221

Manuscript received: 8.11.2008

## Polymers Processing in Engineering - PPE 2009



### THE 2<sup>ND</sup> INTERNATIONAL CONFERENCE ON POLYMERS PROCESSING IN ENGINEERING - PPE 2009

#### THE 2<sup>ND</sup> INTERNATIONAL CONFERENCE ON POLYMERS PROCESSING IN ENGINEERING PPE 2009

**Polymer Processing in Engineering (PPE) Conference** held every two years aims to gather academic researchers and industrial partners involved in the field of polymers

**PPE 2009** will cover all of the important areas in the field, from state-of-the-art research and development to characterization, fabrication, technology development, numerical modeling and many new and emerging applications of polymeric materials. Sections related to specific topics of the conference will be introduced by a Keynote Lecture.

HOME

Background & Scope

Organizers & Committees

Conference Topics

Important Dates

Abstract Submission

Final Program

Sponsor & Exhibitor

Location & Accommodation

Contact Information

