Reunião Anual do INCT-FCx - 2022

De 26 a 29/Outubro de 2022,
Hotel Estância Atibainha, Nazaré Paulista/SP

# Marlics and Mclics liquid crystals simulators

Authors: **Eric K. Omori**[1] , Rafael S. Zola[1,2]

[1]*Departamento de Física, Universidade Estadual de Maringá*
[2]*Departamento de Física, Universidade Tecnológica Federal do Paraná, Campus Apucarana*

**Abstract:** In this work we present two Liquid Crystal Simulators: Marlics (Maringá Liquid Crystal Simulator) and Mclics (Monte Carlo Liquid Crystal Simulator), both softwares written in C++. Marlics is used to obtain either the system dynamics, by using the Runge-Kutta method, or the minimum energy states with the Fast Inertial Relaxation Engine (FIRE). Mclics is used to obtain minimum energy states and execute thermal quenching procedures. In both cases, achiral and chiral nematic liquid crystals can be simulated. The system solved by Marlics consists in the dynamical evolution for the Q-tensor in the Landau-de Gennes formalism, while Mclics uses different pairwise potentials and the Monte Carlo procedure, both codes for several geometries, including confined slab cells and spherical, liquid crystal droplets. Furthermore, the codes accepts custom geometries, so the user may provide a particular geometry of interest to run simulations. The programs take as input a descriptive file giving the simulations parameters and initial conditions, generating a series of different snapshots distributed in time according to the user's needs. Several initial conditions are provided to help the user, starting simulations direct to different goals. Both softwares were distributed as open-source codes, organized in class modules, which can be modified by the user base to attend their further needs. Both codes are parallelized for CPU and Cuda based GPU computing. We describe the basic ideas behind each code, and a few test cases.

**Key-words**: Landau-de Gennes Formalism, Monte Carlo Simulation, Liquid Crystals
**Support:** This work has been supported by CNPq, CAPEs and INFC-Fcx

**References:**

[1] R.F. de Souza, E.K. Omori and R.S. Zola , Computer Physics Communications 277 (2022) 108379.
[2] E. K. Omori and R. S. Zola, in production (2023).