ExTASY: Extensible Tools for Advanced Sampling and analYsis

CHE-12614

Cecilia Clementi, Shantenu Jha, Mauro Maggioni, Iain Bethune, Charles Laughton, Ben Leimkuhler, Panos Parpas

RICE · RUTGERS · epcc · The University of Nottingham · The University of Edinburgh · Imperial College London

Light-weight toolkit to enable ensemble-based simulations and their integration with dynamic analysis capabilities and ultra-large timestep integration methods.

Three Critical Requirments, one Toolkit!

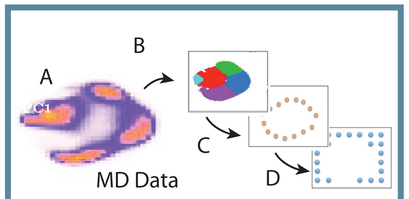
ExTASY Control ↔ **Analysis Tools**

ExTASY Control ↕ **Ensemble Methods**

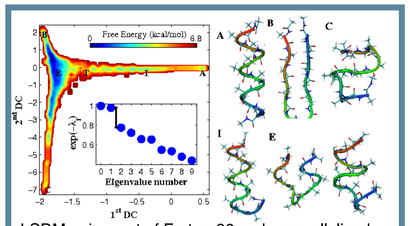
ExTASY Control ↕ **MIST-modified MD Engine**

Need first-class support for dynamic and adaptive execution

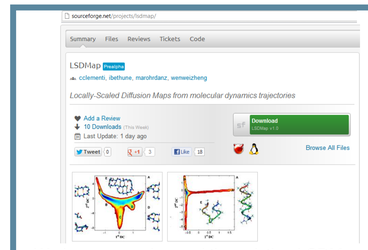
We are working on the development and application of software that uses adaptive dimensionality reduction methods to analyze and improve the sampling of conformational space in molecular simulations (particularly, but not limited to, biomolecular systems).



Reimplementation of the core PCAZIP methodology in a flexible Python-based package is nearly complete. Trajectory data from every common MD package can now be imported and analyzed. The package in its prototype form is being applied to two on-going research projects.



LSDMap is a set of Fortran90 codes parallelised with MPI for the calculation of Locally Scaled Diffusion Maps. The eigenvectors of the diffusion matrix can be used to form a small set of reaction coordinates describing the behaviour of a high-dimensional data set such as an equilibrium MD

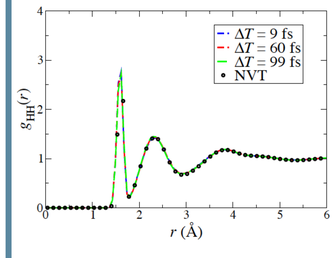


We have re-engineered the existing LSDMap analysis software into a single flexible executable, ready to be incorporated as part of the ExTASY toolkit. The software has been released to the community.

These tools will plug in to the overarching ExTASY framework, complementing work packages from the other contributors, to provide flexible, high-performance workflows.

Need advanced Integration/ MTS for increased time scales

We are modifying integration routines with the goals of accelerating configurational sampling and enhancing domain motion using collective variables. An important way forward is to relinquish strict Hamiltonian dynamics and to utilize collective variable techniques (e.g. LSDMap, see above) as the basis for novel on-the-fly integration techniques.

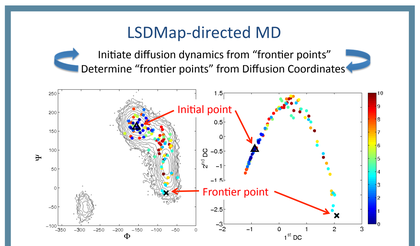


We have developed an aggressive configurational sampling algorithm that combines stochastic techniques with isokinetic multiple time-stepping methodology, with enhanced robustness and numerical stability compared to fully deterministic alternatives.

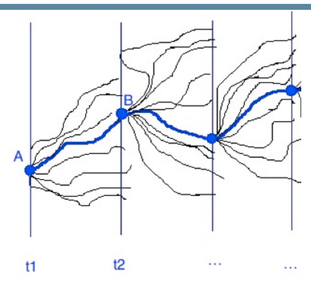
On-the-fly capability requires interaction between MIST-MD and analysis tools to request, update and report collective coordinates.

Need to support large-ensembles of kernels

Simulating many independent or loosely-coupled replicas of a system can provide a linear increase in the rate of exploration of conformational space, and, therefore, a linear increase in the rate of sampling rare events, and, consequently, more rigorous quantification of the reliability and significance of observations.

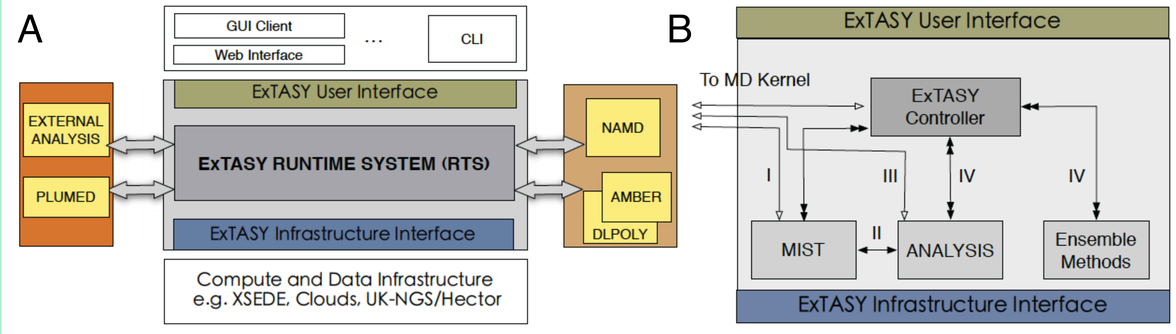


When an ensemble of short trajectories is simulated and locally analyzed by LSDMap, the point with the largest diffusion distance is the "frontier point", from which a new swarm of trajectories may be initiated. Iteration of this procedure forces the system to move away from a conformational minimum and significantly increases the probability to overcome barriers.



Multiple (thousands) of trajectories are run simultaneously for short intervals of times, and restarted on the basis of information gathered by the results from the analysis of the previous

Ensemble as a fundamental abstraction: enabling thinking and programming (Ensemble-API) in terms of "ensembles"



(A) Design of ExTASY. The modular elements that are developed or adopted are: the ExTASY Runtime, the ExTASY interfaces to the user and the infrastructure, and plugins to community codes and analysis routines. (B) Architecture for the ExTASY Runtime, which defines the main components and the control flow between them. Closed double-headed arrows represent communication between components internal to ExTASY; arrows with open-heads represent communication from an ExTASY component to the MD Engine. ExTASY uses SAGA-based Pilot-abstractions to provide fundamental support for ensembles, and SAGA to implement the infrastructure interface to different back-end systems.

Software Design and Objectives:

- Abstraction based, application driven software development
- Design of inter-component interfaces and user-facing APIs
- Python will support packaging as a software library
- Extend off-the shelf libraries and components where possible
- Support for well established kernels (eg NAMD)