

New generation of LLRF and beam-based feedback stability models

C. Serrano¹, J. Byrd¹, L. R. Doolittle¹, D. Driver², G. Huang¹, Q. Llimona³, A. Queiruga², A. Ratti¹

¹Lawrence Berkeley National Laboratory, Berkeley, CA 94720

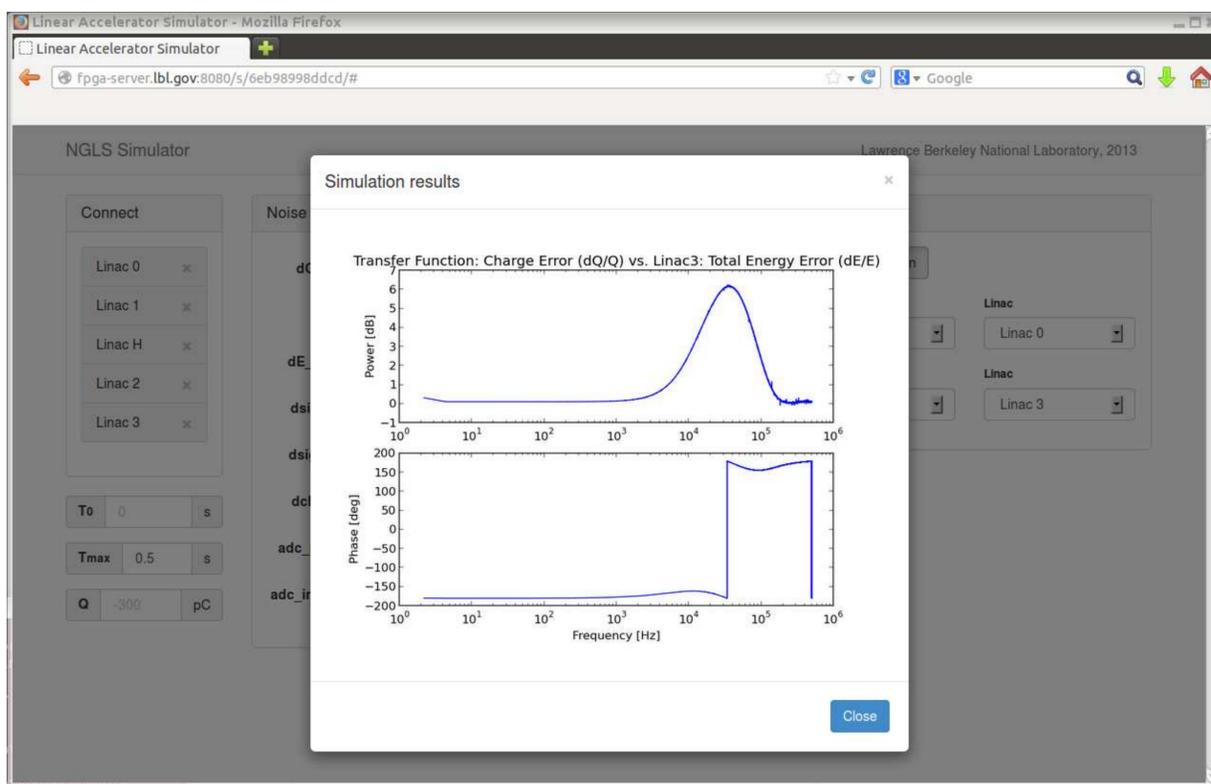
²University of California, Berkeley, CA 94720

³Universitat Pompeu Fabra, Barcelona, 08018, Spain

Abstract

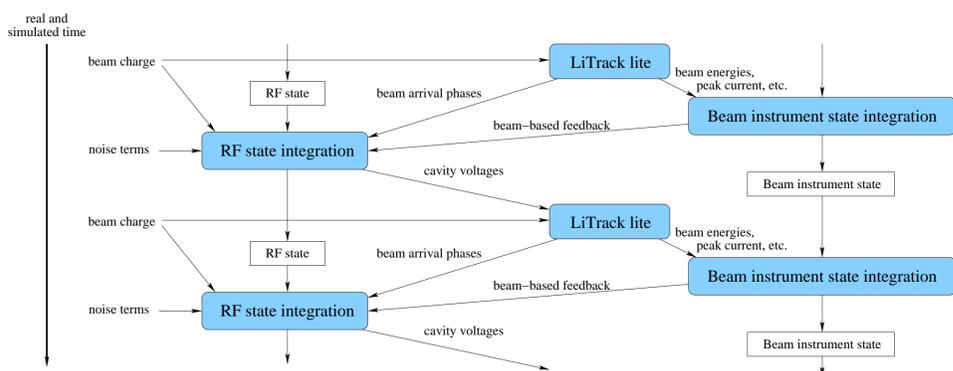
All RF groups in the accelerator community use modeling to some extent for LLRF development, and LBNL has traditionally invested a significant effort in this area for a number of reasons. It is possible to solve many issues in the control algorithms in a comfortable software environment, often well before the availability of production systems. Software and hardware designs can be combined in simulation, easing the transition to final hardware system integration. Over time we have built a solid base of generic models which can be applied to a variety of machines and configuration schemes. The low-level implementation of the models has been re-implemented and extended to include elements in beam-based feedback loops and optimized to improve computation speed, memory usage, scalability and flexibility. The way they are configured and interact with high-level software has been greatly improved to provide clean APIs and a convenient configuration scheme accessible from user-friendly tools such as GUIs and websites.

User Interface



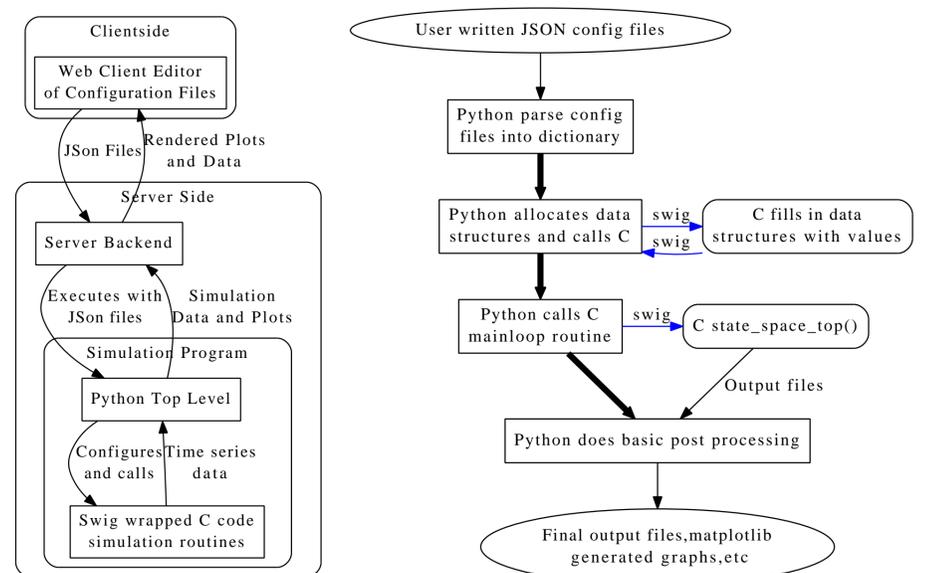
- ▶ Convenient, web-based user interface to mask back-end and configuration details.
- ▶ JavaScript based,
- ▶ Creates JSON configuration files from user input and runs the simulator back-end,
- ▶ Handles configuration file, result and plot storage for easy retrieval,
- ▶ Runs Hash on configuration file (including git commit ID of simulator source) and skips running simulation if it has been run before.

Simulation data flow



- ▶ Integrated particle longitudinal tracking simulator with LLRF and beam-based feedback models,
- ▶ Full integration of instrumentation models, along with LLRF controller and different noise sources,
- ▶ Any part of the system can be enabled/disabled and configured to match any Linac configuration.

Simulation engine architecture



- ▶ High-performance: 30 seconds for 1.2M steps,
- ▶ Documented, unit tests, and Python interface.