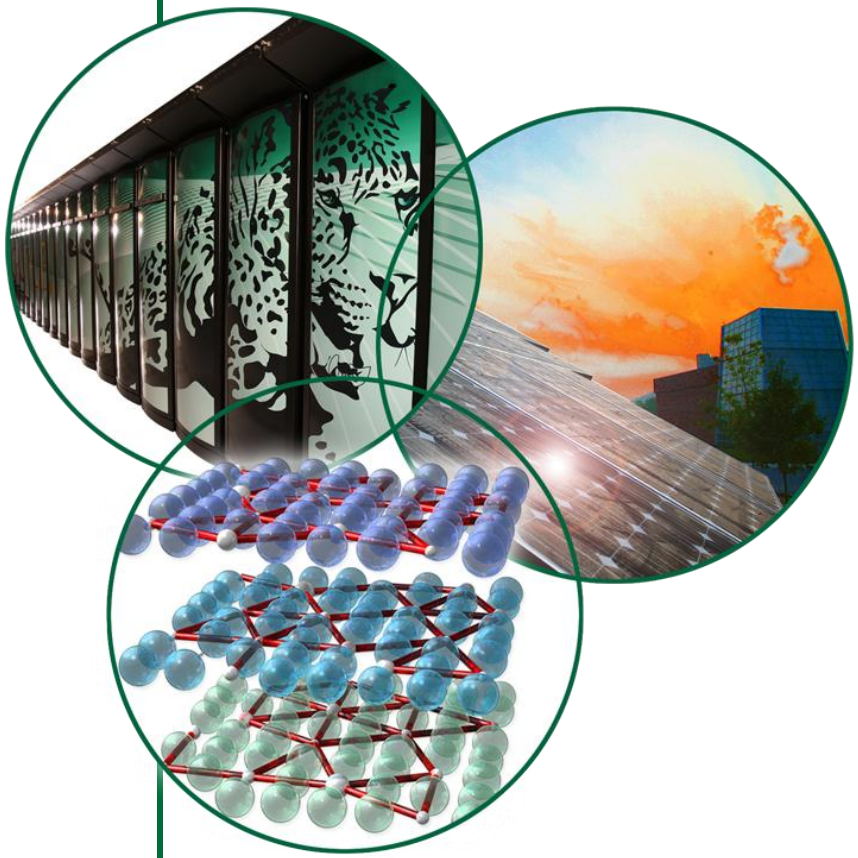
Physics Improvements in SNS XAL

Christopher Allen
ORNL



U.S. DEPARTMENT OF ENERGY

OAK RIDGE NATIONAL LABORATORY
MANAGED BY UT-BATTELLE FOR THE DEPARTMENT OF ENERGY

Outline

- Background and Overview
- Enhancements
- Upgrades
- Verification
- Know Issues

Overview

XAL is composed of three major systems

1. General tools including a GUI package
2. Machine Hardware Representation, hierarchical and introspectively
3. A self-synchronizing machine simulator known as the ***XAL Online Model***

- **Here we present some recently added features of the online model**

NOTE:
This presentation should really be titled “Physics Improvements in Open XAL.” Most of the upgrades to the XAL Online Model were performed while on sabbatical at J-PARC. Although some upgrades came from the SNS repository, most of the SNS repository is still legacy systems.

Background

XAL Online Model Architecture

- Based upon Element/Algorithm/Probe design pattern originally put forth by N. Malitsky.
- As such it is capable of simulating (efficiently) a variety of different beam aspects using the same machine model
  - Single particle (centroid)
  - Transfer matrix
  - Single phase-plane Twiss parameter
  - Fully-coupled RMS envelope
  - (Multi-particle is possible, but impractical – too CPU intensive)

Enhancements: Envelope Simulation

- **Emittance growth due to phase spread**
  - **Due to the finite length of a beam bunch the RF phase of an accelerating gap varies at the beam passes through it**
  - **This in an increased beam emittance for all three phase planes**
  - **This is not a space-charge effect. The online model still considers only the linear part of space charge forces.**

Modeling the emittance growth effect requires the evaluation of Bessel functions. As such, a Bessel function package was implemented and placed in the XAL tools package

Enhancements: Envelope Simulation

Back-Propagation of RMS Envelope Probes

- Given Twiss parameters downstream, then can be propagated to their corresponding values upstream
  - Emittance growth cannot be used because of its nonlinear nature.
  - Invoked using the method Scenario#runBackwards().

Enhancements: PMQ Modeling

- Permanent Magnet Quadrupoles beamline elements are now modeled
  - The current implementation is crude
    - Beamline element are not independent in this case, field dependent upon beamline position (not element position)
    - Violates the Element/Algorithm/Probe architecture
  - I have a possible solution
    - Model PMQ as a collection of `IdealQuadrupole` and `PmqFringeField` elements

Upgrades: Space Charge

The space charge mechanism was upgraded removing known bugs and limitations

- **Procedure works for any beam orientation and has been verified with Trace3D and IMPACT**

Given h, $\boldsymbol{\sigma}$, γ_0 {

- From γ_0 form the Lorentz transform $\mathbf{L}_0$.
- Form the translation matrix $\mathbf{T}_0$.
- Compute the rotation matrix $\mathbf{R}_0$.
- Compute the space-charge generator matrix in the beam frame $\mathbf{G}_{sc0}$ from $\boldsymbol{\sigma}$.
- Compute space charge transfer matrix $\boldsymbol{\Phi}_{sc}(h)$ from $\mathbf{G}_{sc0}$, $\mathbf{R}_0$, $\mathbf{T}_0$, and $\mathbf{L}_0$.

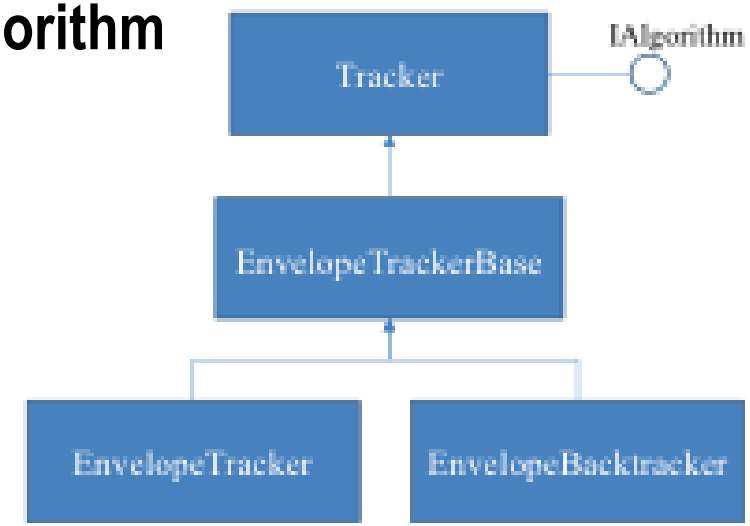
}

Outline of space-charge algorithm

Upgrades: Envelope Algorithms

The architecture and computational techniques of the RMS envelope tracking algorithms was upgraded and improved.

- **Due to the abundance of different tracking algorithms for envelope probes the the class hierarchy was streamlined for improved maintenance and readability.**
- **The algorithms used are fully symplectic.**
- **There is an adaptive step-sizing algorithm**



Upgrades: Javadoc

- Significant addition were made to the Javadoc associated with the XAL online model.
  - Implementation details, technical explanations, and theoretical background now available in online documentation
  - HTML extensively used for readability

Verification:

Extensive verification of the XAL online model was done against the envelope code (Trace3D) and the multi-particle simulation code (IMPACT)

- **RF Gap model:**
  - **Corrected longitudinal normalization error**
- **Emittance growth model: (has multiple models)**
  - **Can reproduce Trace3D exactly with uniform distribution**
  - **Can follow IMPACT transversely with Gaussian model**
- **Space charge effect:**
  - **Can exactly reproduce Trace3D**
- **Bending Magnet:**
  - **Still have unexplained issues for SNS, but not J-PARC ??**

Issues: Phase Slip

The XAL online model can simulate the effects of variable bunch arrival times, which is particularly convenient for building tuning applications for accelerating structures

- **All the necessary computations are there,**
- **However, the implementation of this feature is clumsy and brittle**
  - **Unnecessary dependencies**
  - **Violations of the Element/Algorithm/Probe architecture**
  - **A "common block" structure is used**

Some of the implementation shortcomings are the result of the current online model "lattice generator" which destroys the hierarchical structure of the accelerator hardware representation. We lose information about which accelerating gap belongs to which tank.

Issues: Lattice Generator

- The *lattice generator* has long been a weak point of the XAL online model
  - Strips accelerator hierarchy
    - Final machine model is flat, order but no structure
  - Unnecessarily adds another level of model complexity
    - "Intermediate elements" bind to hardware and to modeling elements
  - All beamline element bindings are hard-coded
    - Labor intensive as developers must learn large code blocks
    - All element additions require rebuilding the entire model

The preferred replacement mechanism would be soft-code configurable, that is, configuration via database or file (I prefer XML). The file could define the device binding and the parameters with which to synchronize.

Summary

- Open XAL has the most advanced, up-to-date version of the online model
  - Documented, tested, and verified
- There are still important tasks to complete
  - Lattice generator
  - Phase slip calculation refactoring
  - PMQ refactoring