

Implementing BNL WLS Optical Model in RAT

Chao Zhang

History

- ❑ Lindsey first implemented the new model in Geant4, and used it in the WbLS NSRL run analysis
- ❑ Lindsey ported the code into RAT and made a pull request <https://github.com/rat-pac/rat-pac/pull/20>
- ❑ Andy Mastbaum took over, made a few changes and made a new pull request <https://github.com/rat-pac/rat-pac/pull/45>
 - Code compiled fine, waiting for validation from someone (since 2015/9)

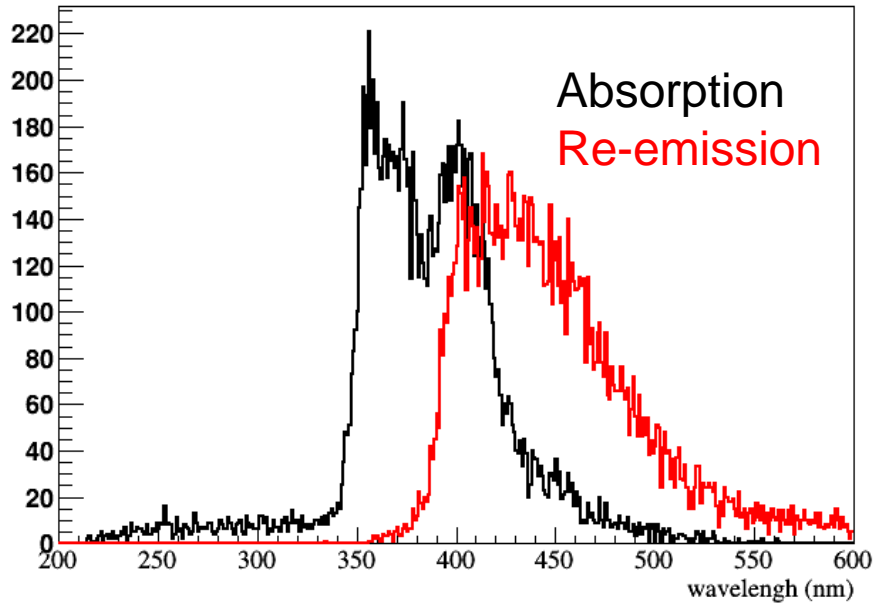
Validation

- ❑ Modified the “cylinder” experiment to use a wbls material with the following new optical properties
 - **QUANTUMYIELD**: re-emission probability vs wavelength
 - **WLSCOMPONENT**: dummy
 - **WLSABSLENGTH**: absorption length vs wavelength
 - **WLSTIMECONSTANT**: time constant
- ❑ Switch to use the bnl wls model in the .mac file
 - **/PhysicsList/setOpWLS bnl**

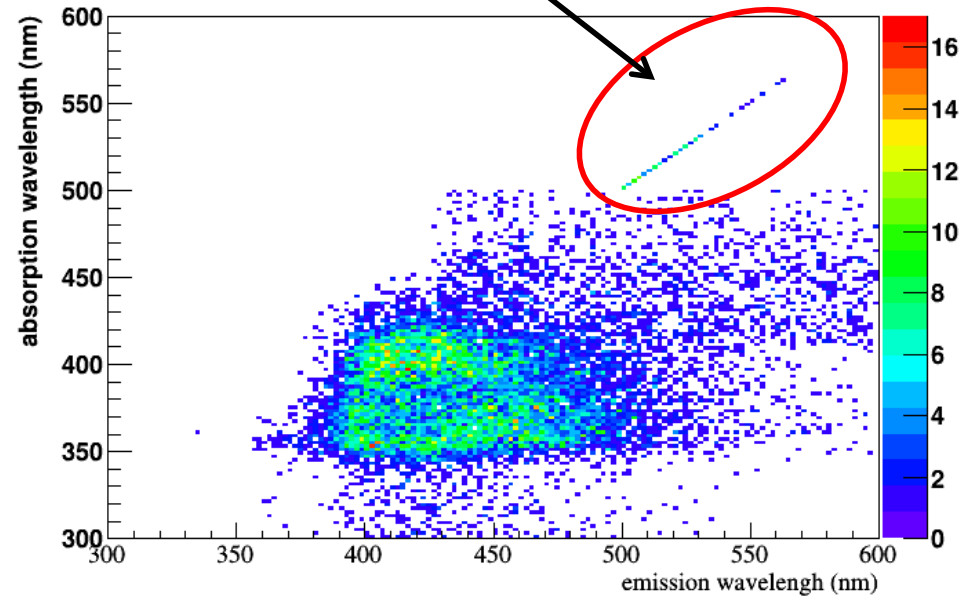
Validation

- ❑ Model switched successfully, but cannot find the excitation-emission matrix root file
 - BNLOpWLS::SetExEmData: Warning: Could not find Ex/Em data file for BNLOpWLS model
- ❑ **Fixed:** in BNLOpWLSBuilder.cc
 - `std::string thePath = std::string(dataPath) + "/" + matrixPath;`

Validation



The excitation wavelength in the root file ends at 500 nm, it then assumed same emission wavelength



- ❑ I recorded the absorbed and reemitted photon wavelength in BNLOpWLSBuilder.cc
- ❑ Looks fine. Final output root file is also fine.

Conclusion

- ❑ The code is validated and can be merged in with a minor fix

```
diff --git a/src/physics/BNLOpWLSBuilder.cc
b/src/physics/BNLOpWLSBuilder.cc
index 09a713e..7454678 100644
--- a/src/physics/BNLOpWLSBuilder.cc
+++ b/src/physics/BNLOpWLSBuilder.cc
@@ -18,7 +18,7 @@ void BNLOpWLSBuilder::ConstructProcess() {

    char* dataPath = getenv("GLG4DATA");
    if (dataPath != NULL) {
-       std::string thePath = std::string(dataPath) + matrixPath;
+       std::string thePath = std::string(dataPath) + "/" + matrixPath;
        theBNLOpWLSProcess->SetExEmData(thePath);
        RAT::info << "BNLOpWLSBuilder::ConstructProcess: Using data from "
                   << thePath << newline;
```