Improvements on Monte Carlo Scintillation Simulations

Gustavo Valdiviesso
Unifal-MG Brazil

Light Detection In Noble Elements - LIDINE 2024

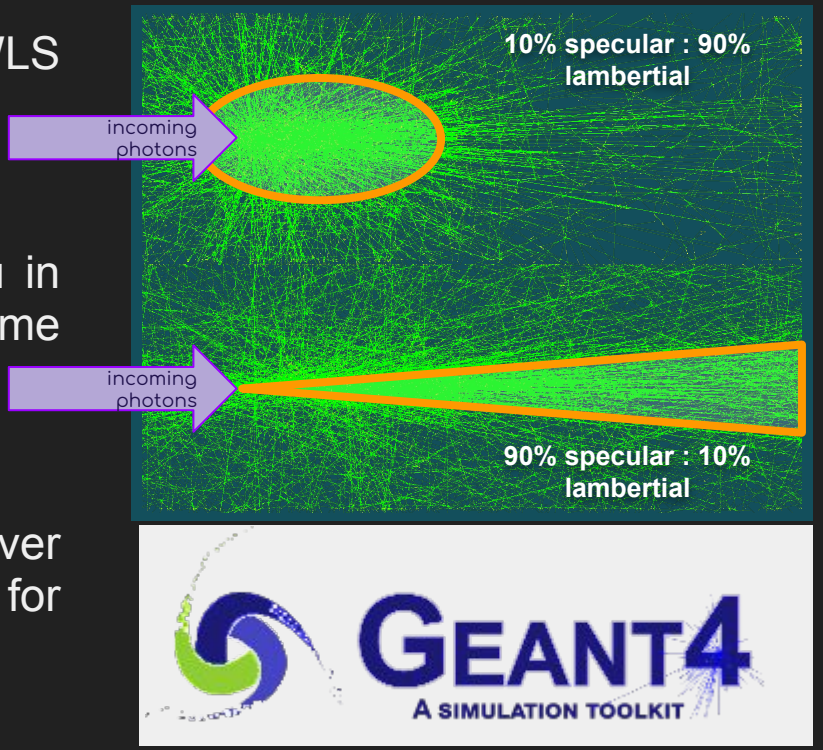
August 28, Instituto Principia, São Paulo, Brazil

Summary

- What is this presentation all about?
- Monte Carlo light production
- Geant4 WLS class
- Pitfalls with the "black box" approach
- Identifying and Correcting biases
- What comes next?
- Conclusions

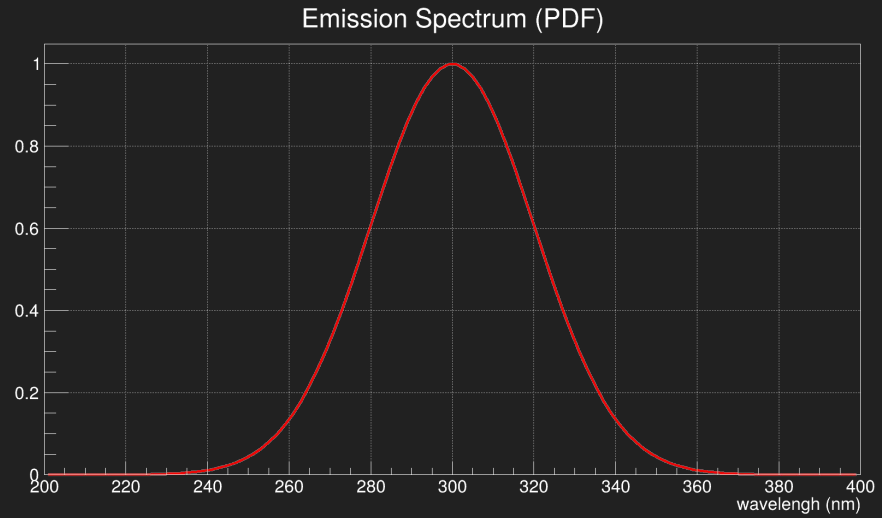
What is this presentation all about?

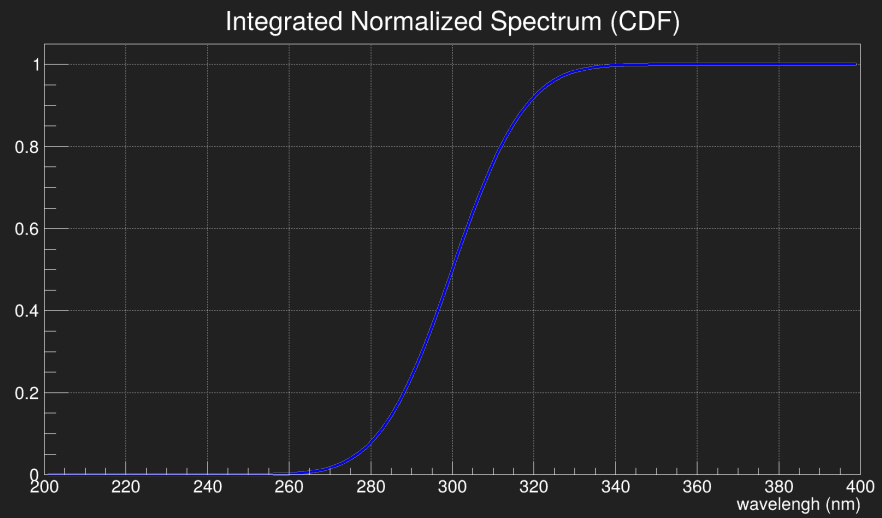
- I've been working with scintillation and WLS processes using G4 for the past 5 years.
- I'd like to share some learnings with you in the hopes that you don't make the same mistakes I've made.
- Consider this a cautionary tale about never using software as "black boxes" for generating scientific information.



Monte Carlo light production

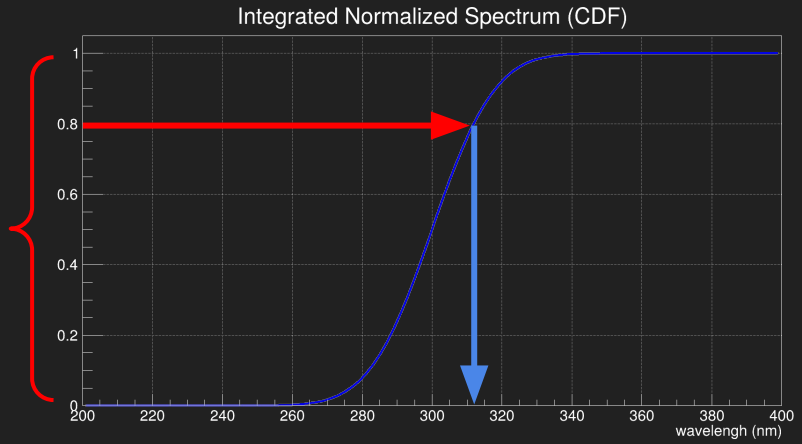
- By treating the material's scintillation spectrum as a probability density function (PDF) we obtain the cumulative distribution function (CDF)

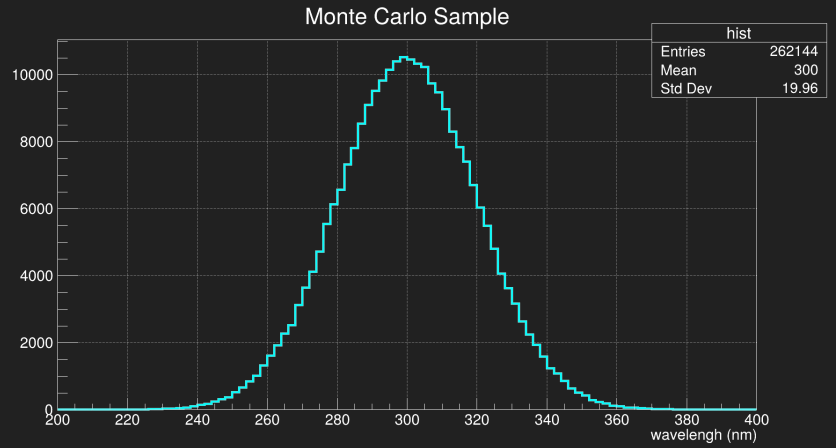




Monte Carlo light production

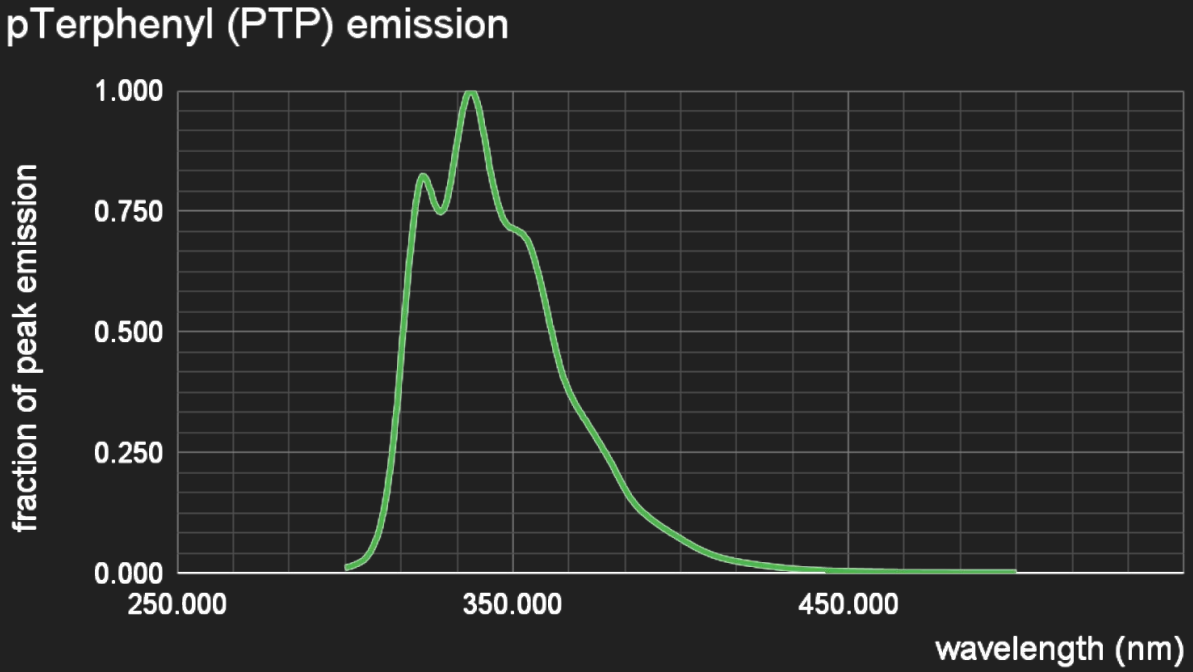
- From a uniform random throw, the CDF (actually its inverse) returns a corresponding random wavelength (or energy).
- The resulting value represents our MC photon.





Monte Carlo light production

- The concept remains the same when applied to real scintillation spectra

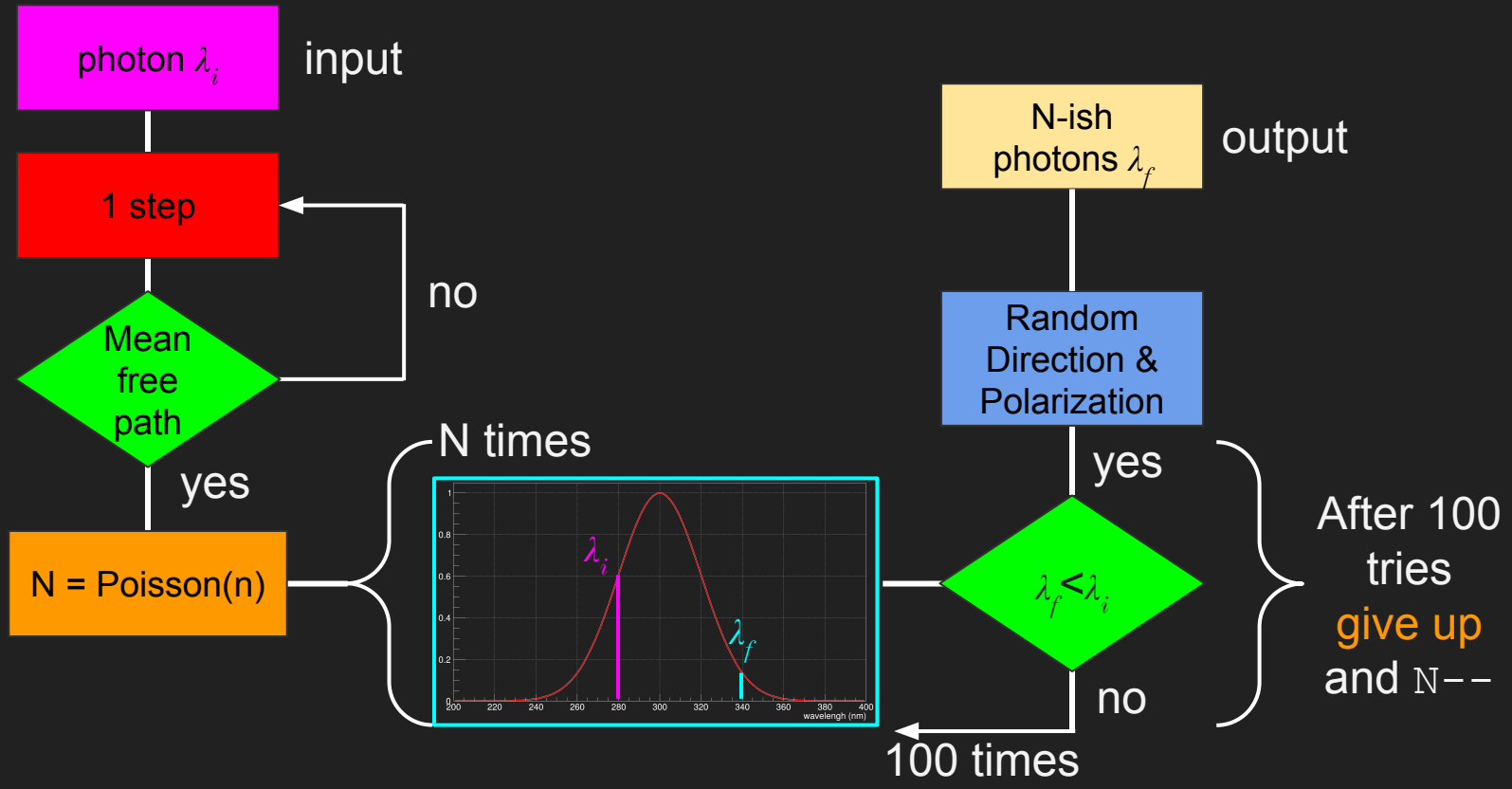


G4OpWLS class

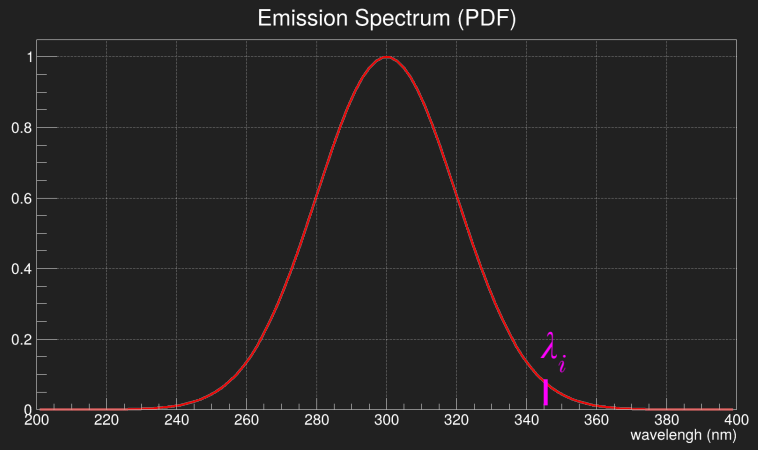
- Takes care of generating an Optical Photons (G4Particle) sampled from a given spectrum.
- It's used both for Scintillation and Wavelength-shifting.
- It's 11 years old and hasn't seen any methodological updates during this time.

```
////////////////////////////////////////////////////////////////////////
// Optical Photon WaveLength Shifting (WLS) Class Implementation
////////////////////////////////////////////////////////////////////////
//
// File:        G4OpWLS.cc
// Description: Discrete Process -- Wavelength Shifting of Optical Photons
// Version:     1.0
// Created:     2003-05-13
// Author:      John Paul Archambault
//              (Adaptation of G4Scintillation and G4OpAbsorption)
// Updated:     2005-07-28 - add G4ProcessType to constructor
//              2006-05-07 - add G4VWLSTimeGeneratorProfile
//
////////////////////////////////////////////////////////////////////////

#include "G4OpWLS.hh"
#include "G4ios.hh"
#include "G4PhysicalConstants.hh"
#include "G4SystemOfUnits.hh"
#include "G4OpProcessSubType.hh"
#include "G4Poisson.hh"
#include "G4OpticalParameters.hh"
#include "G4WLSTimeGeneratorProfileDelta.hh"
#include "G4WLSTimeGeneratorProfileExponential.hh"

//....oooOOOOOooo........oooOOOOOooo........oooOOOOOooo........oooOOOOOooo......
G4OpWLS::G4OpWLS(const G4String& processName, G4ProcessType type)
  : G4VDiscreteProcess(processName, type)
{
  WLSTimeGeneratorProfile = nullptr;
  Initialise();
  SetProcessSubType(fOpWLS);
  theIntegralTable = nullptr;

  if(verboseLevel > 0)
    G4cout << GetProcessName() << " is created " << G4endl;
}
```

G4OpWLS class



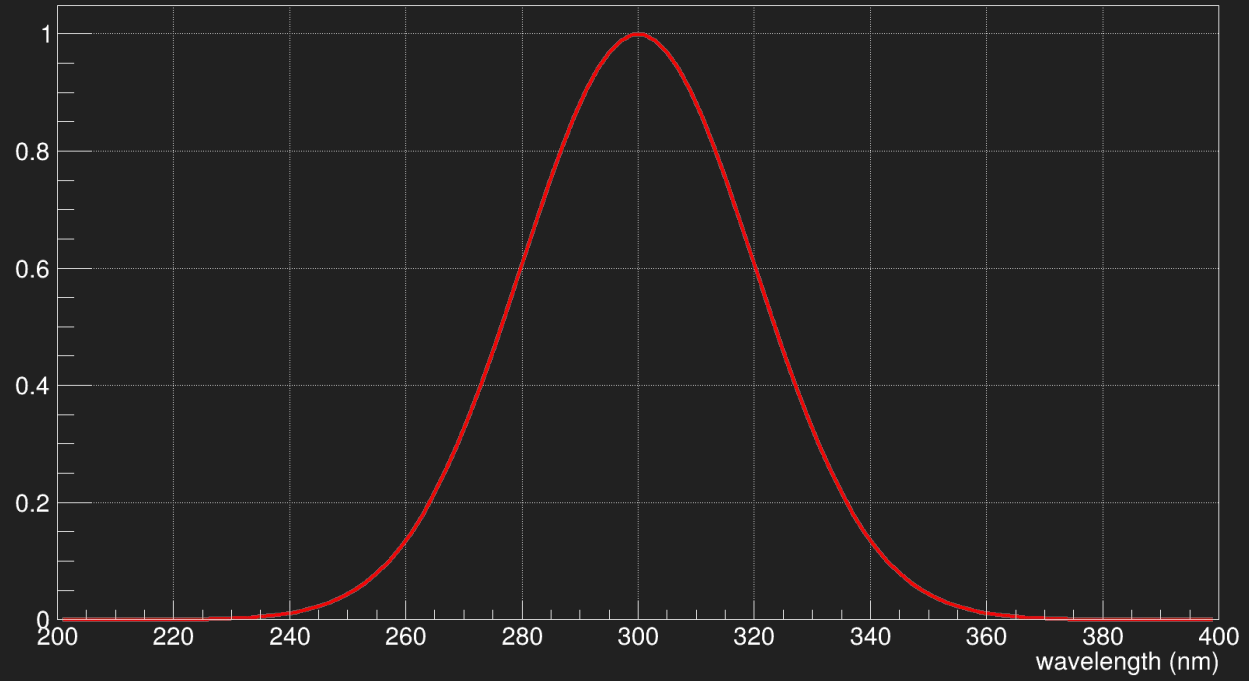
Pitfalls of a "black box" approach



- Is energy really conserved?
  - YES, on a photon by photon basis
  - NO, if n > 1
- What happens when photons with longer wavelengths are absorbed?
  - Longer wavelengths have a cut-off when $p<10\%$
  - Effectively N ≤ n

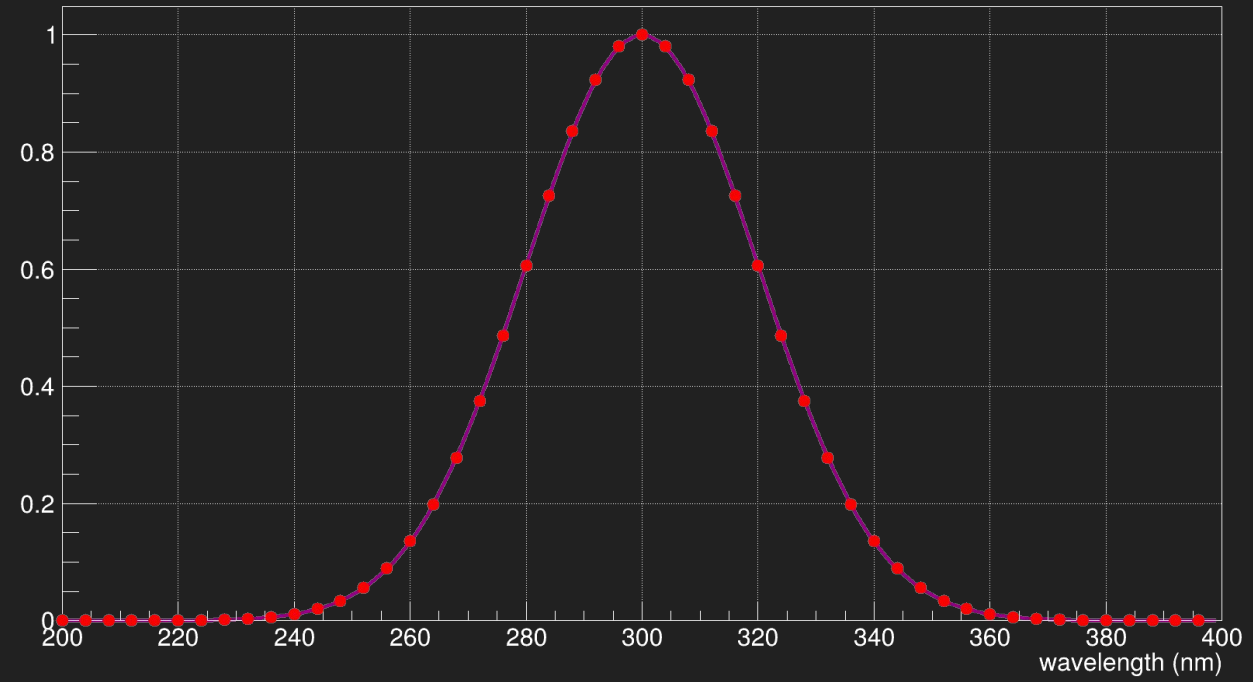
Identifying and Correcting biases

- We often think of our input spectrum as something continuous…



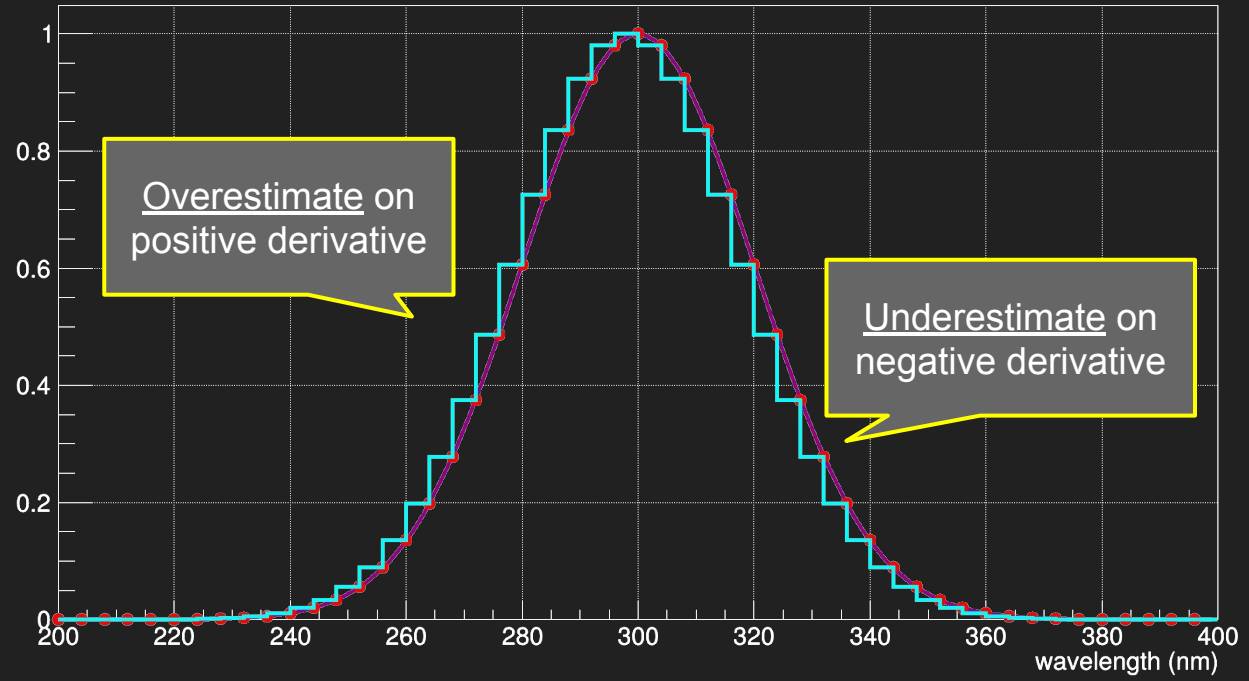
Identifying and Correcting biases

- But it actually isn't!



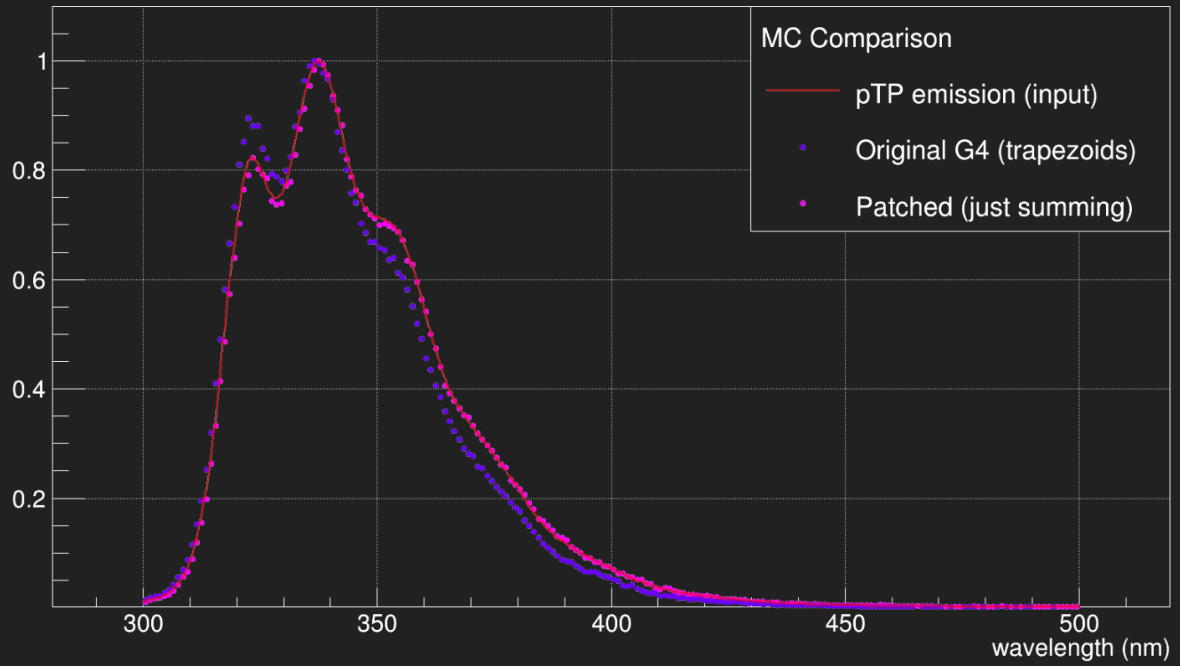
Identifying and Correcting biases

- And G4WLS integrates using trapezoids!



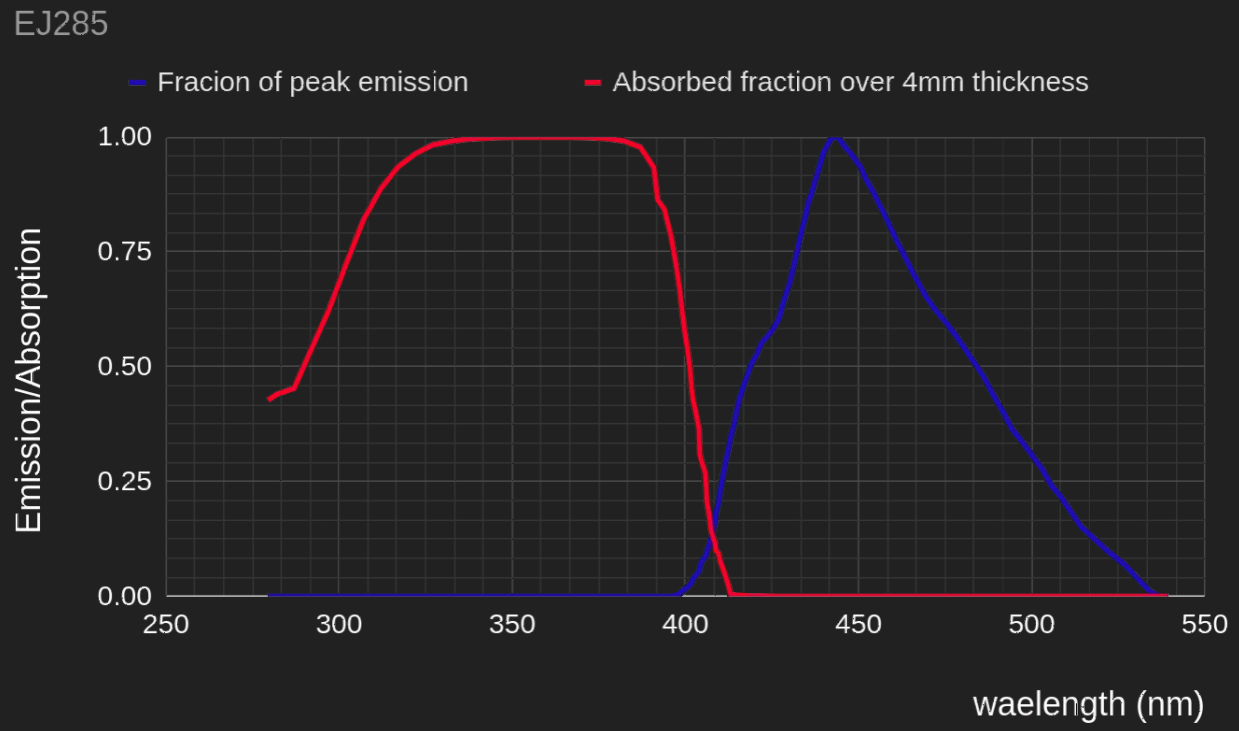
Identifying and Correcting biases

- Effect can be corrected with a better integral method.



Identifying and Correcting biases

- What if absorption and emission overlap each other?

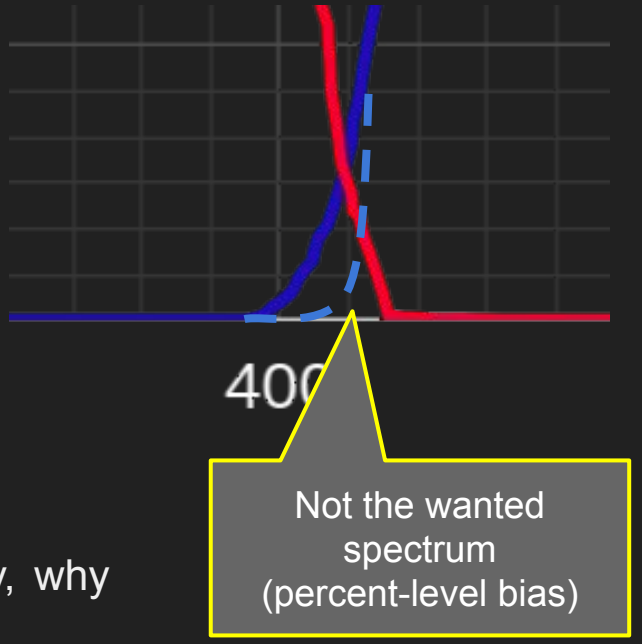


Identifying and Correcting biases

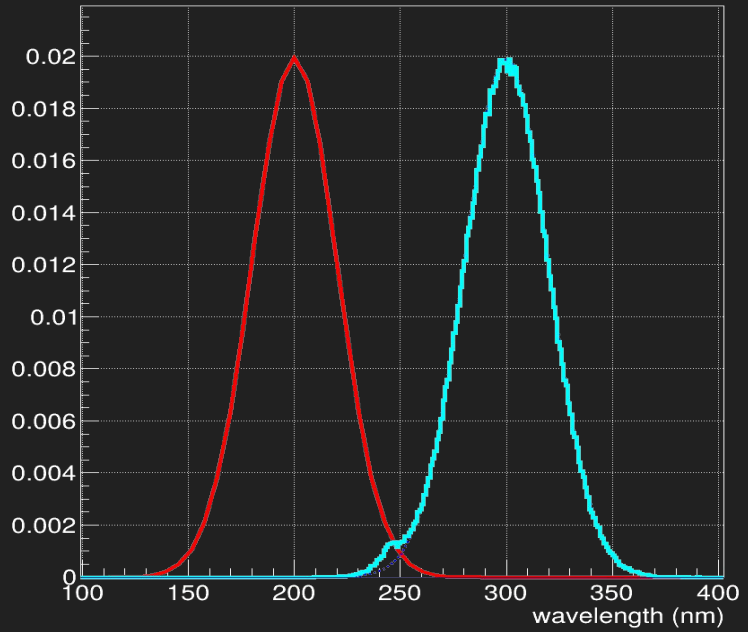
- What if absorption and emission overlap each other?
- The emission algorithm assumes uniform absorption!
- The resulting emission spectrum is different in this region:

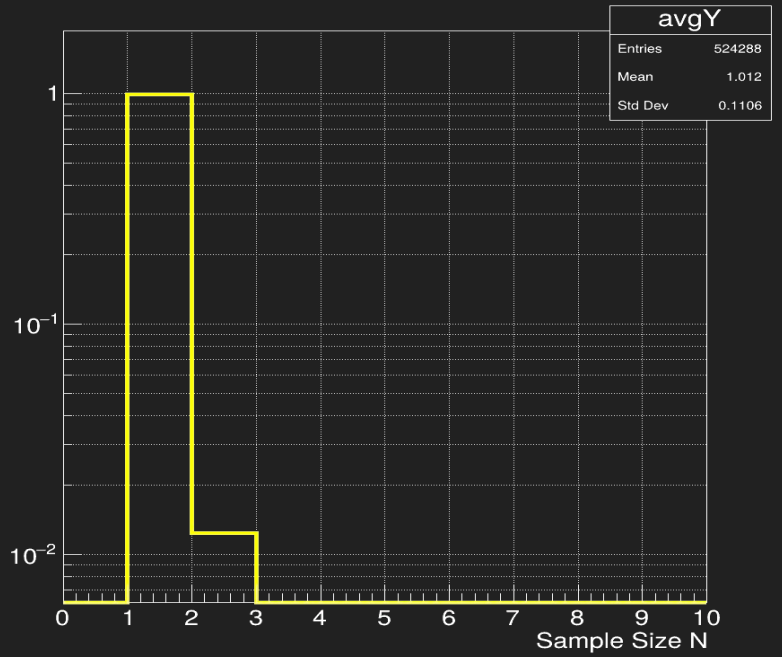
$$p(\lambda_f) \rightarrow p(\lambda_f | \lambda_f < \lambda_i)$$

- Given that conservation of energy isn't ensured anyway, why keep the condition $\lambda_f < \lambda_i$?



What comes next?





Conclusions

- Never assume your software does what you think it does (no black-boxes)
- G4WLS has more than one source of percent-level biases (they accumulate)
- Open source means it's our responsibility as well.
- github.com/gustavogx/G4WLS