CERN

Low energy electron modelling in PyECLOUD

Eric Wulff

Acknowledgements: Thanks to Giovanni Iadarola for helpful discussions and input.

Outline

Introduction

In this presentation we will study two of the available secondary emission modules in PyECLOUD [1]. Specifically we will study and compare the standard secondary emission module,

- `sec_emission_model_ECLOUD.py`

with the secondary emission module for more accurate treatment of low energy electrons,

- `sec_emission_model_accurate_low_ene.py`.

Theory of secondary electron emission

Secondary electron emission

- Is described by δ, the secondary electron yield (SEY)

$$\delta = \frac{I_{emit}}{I_{imp}}$$

- δ depends on the energy of the impacting electrons as well as the angle of incidence
- δ can be divided into components,

$$\delta = \delta_{elas} + \delta_{true} \ (+\delta_{rediff})^1$$

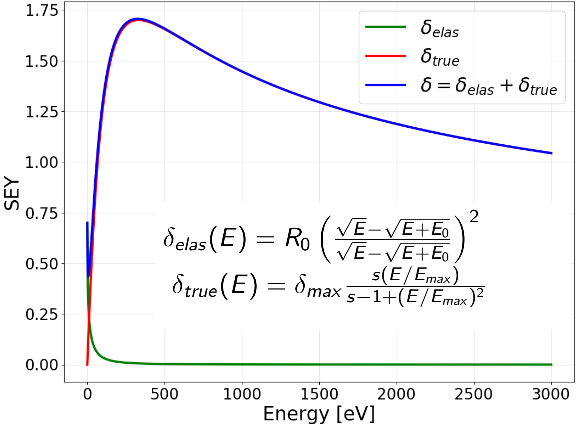
[1] Not used in PyECLOUD but sometimes by others, e.g. Furman and Pivi [2]

The SEY Curve

Let's have a closer look at the SEY components of PyECLOUD.

The SEY Curve

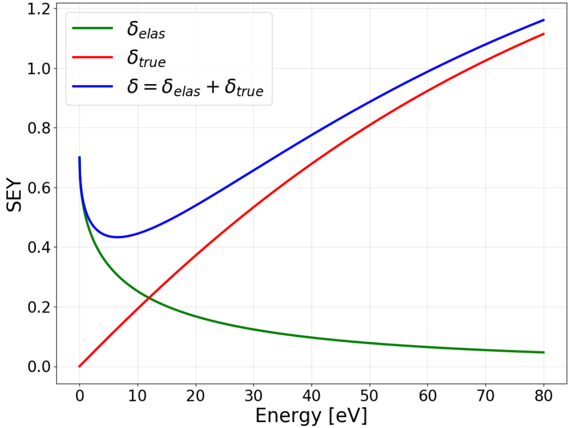
For high electron energies δ_{true} makes up most of δ and the contribution from δ_{elas} is negligible.



The SEY Curve

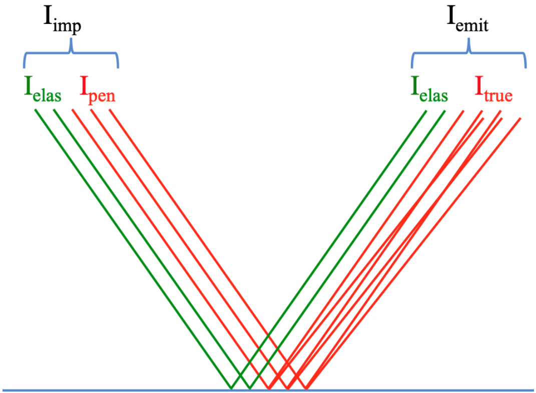
For low electron energies δ_{elas} plays an important role.

Consequently, a difference in treatment of the elastic SEY component mainly affects low energy electrons.



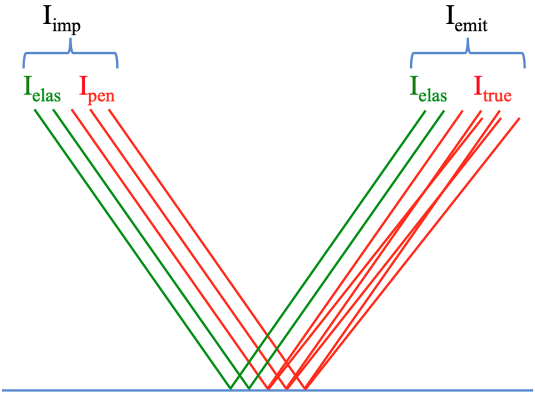
The SEY components

- $\delta = \frac{I_{emit}}{I_{imp}}$
- $\delta_{elas} = \frac{I_{elas}}{I_{imp}}$
- $\delta_{true} = \frac{I_{true}}{I_{imp}}$



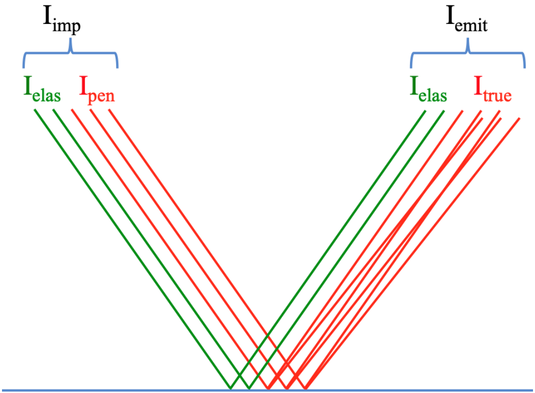
The SEY components

- $\delta = \frac{I_{emit}}{I_{imp}} = \delta_{elas} + \delta_{true}$
- $\delta_{elas} = \frac{I_{elas}}{I_{imp}}$
- $\delta_{true} = \frac{I_{true}}{I_{imp}}$



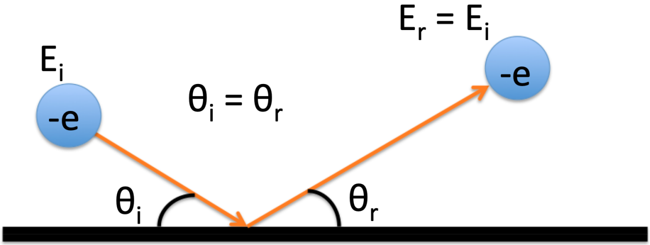
The SEY components

- $\delta = \frac{I_{emit}}{I_{imp}} = \delta_{elas} + \delta_{true}$
- $\delta_{elas} = \frac{I_{elas}}{I_{imp}}$
- $\delta_{true} = \frac{I_{true}}{I_{imp}}$
- $\hat{\delta}_{true} = \frac{I_{true}}{I_{pen}}$
  "number of emitted electrons per non-elastic event"



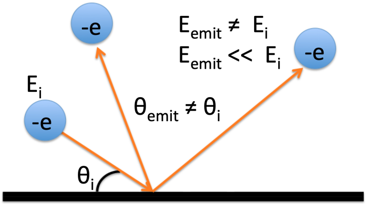
Elastic collision events

- Energy does not change
- Angle of incidence equals angle of reflection



True secondary collision events

- True secondary electrons are generated at various angles and various energies



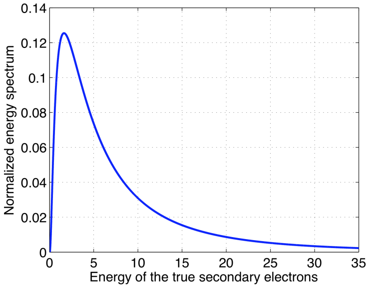
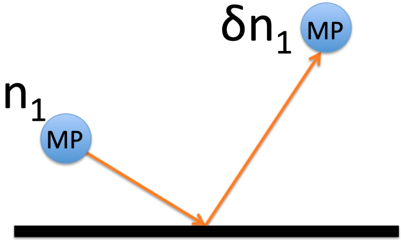


Figure: G. Iadarola [3]

The macroparticle approach

The macroparticle approach

- Due to computational limitations we cannot track individual electrons
- Instead we use macroparticles (MPs), each representing many electrons
- We rescale the MP size instead of adding or removing electrons



Comparison of models and implementation

Comparison of models and implementation

From here on we will refer to the standard module as `ECLOUD` and the module for more accurate low energy electron modelling as `ACC_LOW`.

Comparison of models and implementation

The two modules differ in terms of

- Deciding event type of each collision event (elastic or true secondary)
- Rescaling the MPs

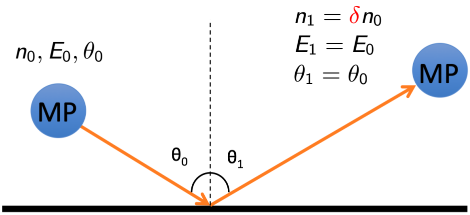
The `ECLOUD` module

The ECLOUD module

Elastic

$P(elas) = \frac{I_{elas}}{I_{emit}} = \frac{\delta_{elas}}{\delta}$

probability that an emitted electron comes from an elastic event



The ECLOUD module

Elastic

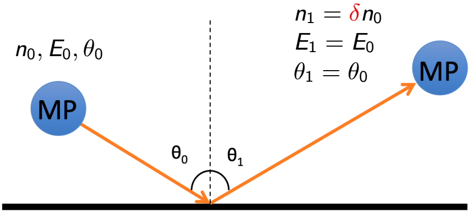
$P(elas) = \frac{I_{elas}}{I_{emit}} = \frac{\delta_{elas}}{\delta}$

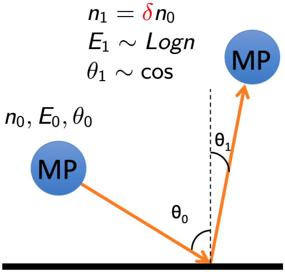
probability that an emitted electron comes from an elastic event

True secondary

$P(true) = \frac{I_{true}}{I_{emit}} = \frac{\delta_{true}}{\delta}$

probability that an emitted electron comes from a true secondary event





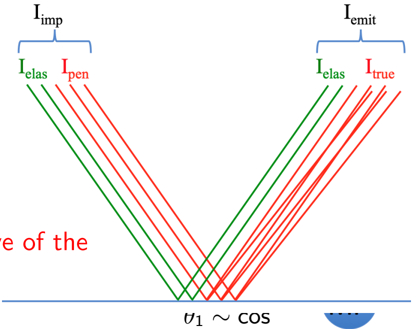
The ECLOUD module

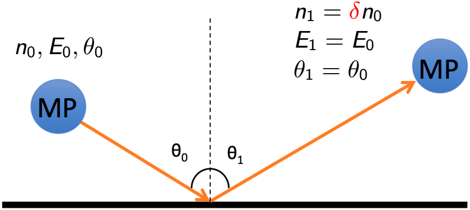
Elastic

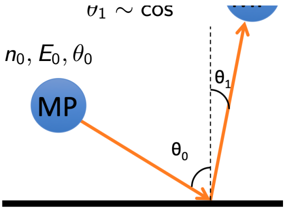
$P(elas) = \frac{I_{elas}}{I_{emit}} = \frac{\delta_{elas}}{\delta}$

probability that an emitted electron comes from an elastic event

Probabilities are defined from a perspective of the emitted electrons

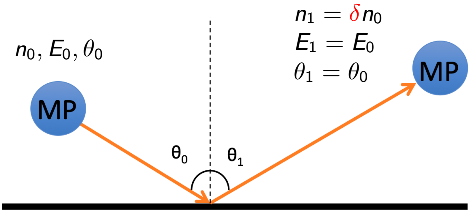




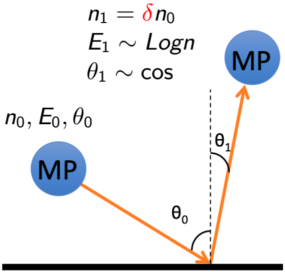


The ECLOUD module

Note: Since we rescale elastically scattered MPs, energy conservation is not respected on single events if $\delta > 1$.



(Averaged over all events, energy conservation is respected)



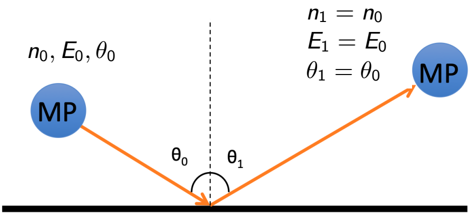
The ACC_LOW module

The ACC_LOW module

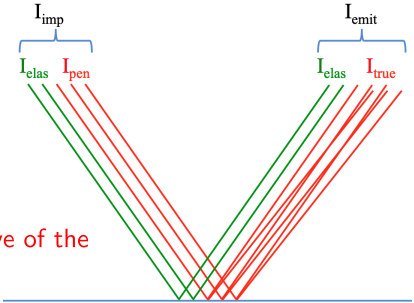
Elastic

$P(elas) = \frac{I_{elas}}{I_{imp}} = \delta_{elas}$

probability that an impacting electron will become elastically scattered



The ACC_LOW module



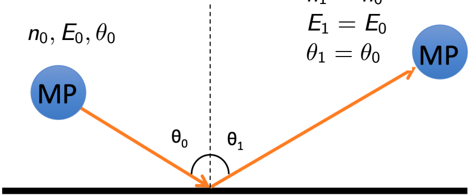
Elastic

$$P(elas) = \frac{I_{elas}}{I_{imp}} = \delta_{elas}$$

probability that an impacting electron will become elastically scattered

Probabilities are defined from a perspective of the impacting electrons

$n_1 = n_0$
$E_1 = E_0$
$\theta_1 = \theta_0$



The ACC_LOW module

Elastic

$$P(elas) = \frac{I_{elas}}{I_{imp}} = \delta_{elas}$$

probability that an impacting electron will become elastically scattered

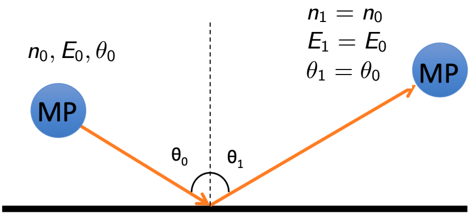
True secondary

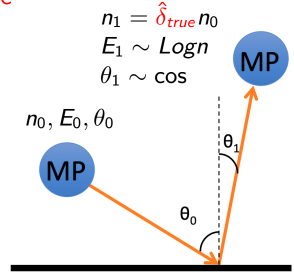
$$P(true) = 1 - P(elas) = 1 - \delta_{elas}$$

probability that an impacting electron will be in a true secondary event

Probabilities are defined from a perspective of the impacting electrons

Note: $\hat{\delta}_{true} \neq \delta$

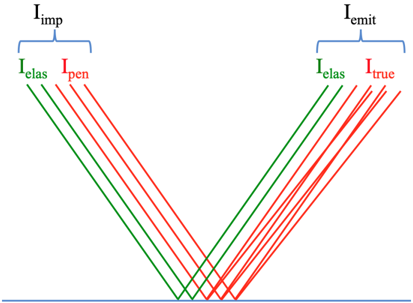




What is $\hat{\delta}_{true}$?

What is $\hat{\delta}_{true}$?

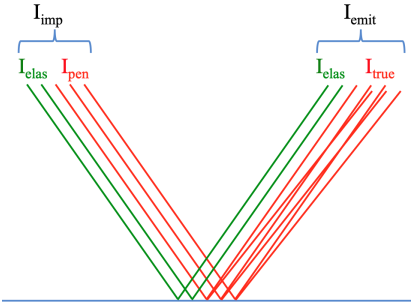
- $\hat{\delta}_{true}$ is the SEY per penetrated current, I_{pen}
- I_{pen} is the fraction of I_{imp} not elastically scattered



What is $\hat{\delta}_{true}$?

- $\hat{\delta}_{true}$ is the SEY per penetrated current, I_{pen}
- I_{pen} is the fraction of I_{imp} not elastically scattered

$$\hat{\delta}_{true} = \frac{I_{true}}{I_{pen}} = \frac{I_{true}}{I_{imp} - I_{elast}}$$

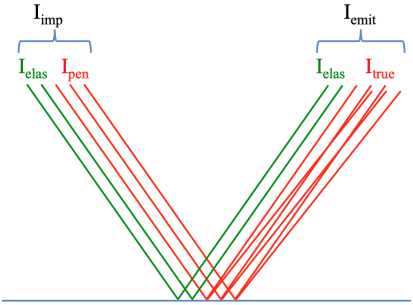


What is $\hat{\delta}_{true}$?

- $\hat{\delta}_{true}$ is the SEY per penetrated current, I_{pen}
- I_{pen} is the fraction of I_{imp} not elastically scattered

$$\hat{\delta}_{true} = \frac{I_{true}}{I_{pen}} = \frac{I_{true}}{I_{imp} - I_{elast}}$$

$$= \frac{\delta_{true} I_{imp}}{(1 - \delta_{elas}) I_{imp}}$$

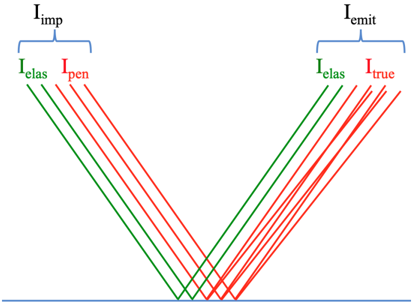


What is $\hat{\delta}_{true}$?

- $\hat{\delta}_{true}$ is the SEY per penetrated current, I_{pen}
- I_{pen} is the fraction of I_{imp} not elastically scattered

$$\hat{\delta}_{true} = \frac{I_{true}}{I_{pen}} = \frac{I_{true}}{I_{imp} - I_{elast}}$$

$$= \frac{\delta_{true} I_{imp}}{(1 - \delta_{elas}) I_{imp}} = \frac{\delta_{true}}{1 - \delta_{elas}}$$



Comparison of simulation output

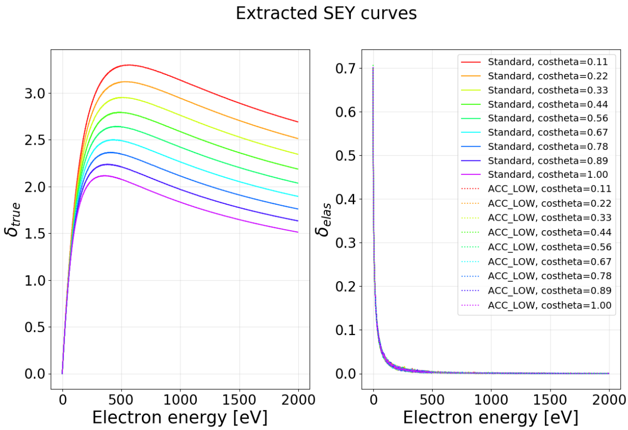
Simulation setup

Simulations using the `ECLOUD` and the `ACC_LOW` secondary emission modules were carried out with the following parameters.

- 450 GeV beam energy
- $2 \cdot 10^{11}$ protons/bunch
- SEY parameter $\delta_{max} = 2.0$
- Circular drift tube with 44 mm diameter

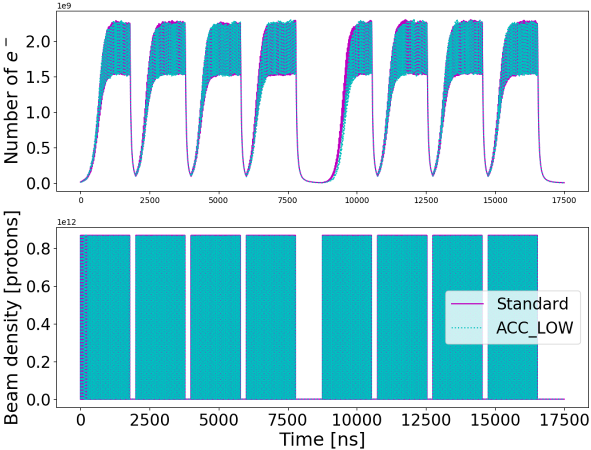
Extracted SEY Curves

As expected, the extracted SEY curves from the two modules are the same.



Electron cloud build-up

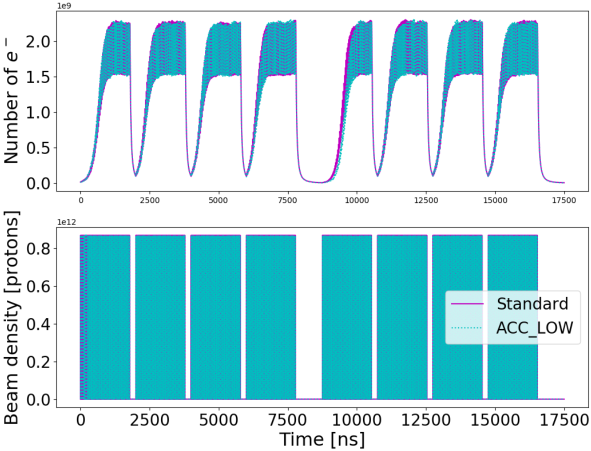
At first glance the simulation results look very similar.



Electron cloud build-up

At first glance the simulation results look very similar.

Let's zoom in on the first part of the second train.

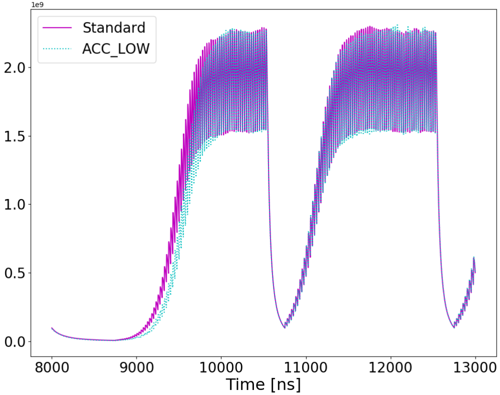


Electron cloud build-up

The buildup is slightly faster in the standard `ECLOUD` model compared to the `ACC_LOW` model.

Will this have and effect on the heat load?

Number of electrons in chamber



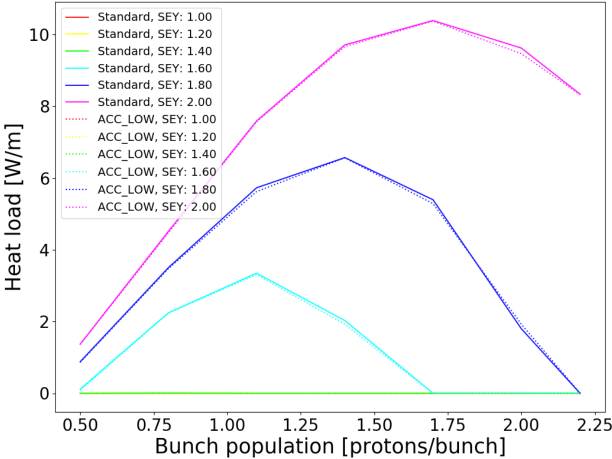
Heat load

How does the produced heat load differ between the two models?

Heat load

How does the produced heat load differ between the two models?

There are no, or only very small, differences.



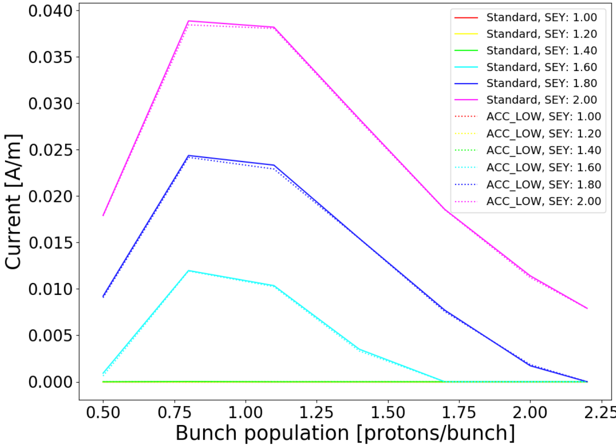
Electron current

Let's also have a look at the current impacting on the beam screen.

Electron current

Let's also have a look at the current impacting on the beam screen.

Again, no, or very small, differences.



Summary

- We have studied and compared the secondary emission modules `sec_emission_model_ECLOUD.py` and `sec_emission_model_accurate_low_ene.py` in PyECLOUD.
- There is no difference in the δ_{true} or δ_{elas}.
- The electron cloud buildup is slightly slower with the `ACC_LOW` module.
- The differences in simulation output of observables like beam screen heat load and current are negligible.

References

Giovanni Iadarola.
PyECLOUD.
https://github.com/PyCOMPLETE/PyECLOUD, 2018.

M. A. Furman and M. T. F. Pivi.
Probabilistic model for the simulation of secondary electron emission.
Phys. Rev. ST Accel. Beams, 5:124404, Dec 2002.

Giovanni Iadarola.
Electron Cloud Studies for CERN Particle Accelerators and Simulation Code Development.
PhD thesis, Università degli Studi di Napoli Federico II, Napoli, Italy, and European Organization for Nuclear Research (CERN), Geneva, Switzerland, March 2014.

CERN