PyTT Manual and Notes

Mariusz Sapinski
Paul Scherrer Institut, Villigen, Switzerland

June 1, 2024

Abstract

PyTT is a python code simulating heating and cooling of a thin wire or screen during a transient heating by a beam pulse. This is a physics reference manual and user guide for the latest iteration of the code. The document contains physics considerations and calculations concerning the phenomena accompanying the thin targets exposed to high brightness particle beams. It contains various divagations and remarks which are usually not found in public documents. Hopefully it can help a potential user with the issues with the code.

1 Introduction

Thin targets are wires, stripes or foils often used in accelerators to perform measurements of beam properties. Typical application are wire scanners, which probe the transverse beam profile using a thin wire. Wire thickness is typically in the range of 10-100 μm. Other devices for which this simulations can be useful are septa magnets, beam splitters, scintillation and SEM foils, etc.

The PyTT code is used to calculate physics of thin targets interacting with particle beams. The primary goal of the code was to calculate target temperature evolution under the beam conditions. The later addition are calculations of secondary emission current and spectra of the emitted electrons and comparing them to thermionic current.

The code performs calculations in one or two dimensions, perpendicular to the beam direction, therefore it is relatively simple.

2 History

The problem of thermal behaviour of thin targets has been tackled by several researchers in various laboratories. Their calculation codes are usually 'private' [Hu, Doe, YIL+23, dBP17] and results are used internally. Therefore, PyTT is the first public code and probably represents a most complete approach to thin target thermal simulations.

The history of this code started in 2006 when Jan Koopman [1], engineer responsible for CERN wire scanners, inquired about estimation of the wire scanner temperature when scanning high-brightness LHC beams. The first version of the code, released in 2008 and documented in [Sap08], was written in C++, and was used as root-script (called WireTemp2D.C). It was used to determine wire scanner usage limits on LHC and SPS and to produce results presented in [MS08, MS09, MS10].

In 2016, during a month-long stay at J-PARC, the author converted the original root-script to python 2.7. Also, this is when the program got a name pyTT (originally starting with lowercase letter), where TT stands for Thin Targets. This was a first version of PyTT.

Around 2020 the code was rewritten in python 3 by Araceli Navarro. The changes to the code concept and structure were substantial and new features were introduced, for instance secondary electron yield. It is well described in Araceli's PhD thesis [Nav23]. This code was sometimes called PyTT2.

This document refers to this last iteration of the code, which can be found on github [AN23]. It contains numerous important changes with respect to the PyTT2, therefore it is called the third python version (on top of the very first C++ version) or PyTT3.

[1] see https://cds.cern.ch/record/1221061?ln=en

3 Main numerical concepts

The code uses internally the following concepts:

- divides the wire movement into steps, dt, which are computed in one of the following ways:

$$\Delta t = d_{wire}/v_{wire} \tag{1}$$

  or

$$\Delta t = \frac{x_{max} - x_{min}}{N_steps \cdot v_{wire}} \tag{2}$$

  where d is diameter of the wire or similar characteristic dimension if the wire is cylindrical;

- if second equation is used, the Δt must always be smaller than d_{wire}/v_{wire}
- default number of steps is 10'000, however it can also be changed in the configuration file (TBD!);
- divides the wire into small segments which are called cells or sometimes bins; the length of h of the bin should be smaller than beam size σ_{beam} in the direction of the wire;
- temperature change is calculated once per step, after calculation the summation of all energy transfers:

$$\Delta T = \Delta E/(m \cdot c_p) \tag{3}$$

  where m is mass of a cell and c_p is specific heat of the material [2]

- the applied numerical simulation method is called Finite Difference Method (FDM) and the particular scheme used in pyTT is FTCS (Forward Time Center Space) (verify).

3.1 Beam profile

The total number of protons traversing a wire N_{wire} can be expressed by:

$$N_{wire} = I_{beam} \cdot d_{wire}/v_{wire} = N_{circ} \cdot f_{rev} \cdot d_{wire}/v_{wire} \tag{4}$$

where I_{beam} is a beam current in number of particles (e.g. protons) in the beam per second. In order to calculate number of particles passing in a given simulation step by a given segment of the wire, the following is applied.

The 2D transverse profile of the beam is created by `CreateNiMatrix` function in `TempPhysicalModels` module. This function uses position vectors to generate a map of the beam transverse profile normalized to one.

4 Physics processes

Here we describe the relevant physics processes. Some of them, as the beam heating due to energy loss in interaction with material electrons, are well known. Others, like RF heating, are described only qualitatively, or discussing the examples, because general description is difficult or impossible.

4.1 Beam heating

Energy deposited by the beam in the material depends on the interaction of the beam particles with material electrons. It is commonly expressed as dE/dx. This can be taken, for instance, from PSTAR NIST database, calculating Bethe-Bloch formula or using simulation tools like Geant4.

To calculate the energy left by a single particle traversing the center from dE/dx (tabulated or from Bethe-Bloch) of wire with diameter d is:

$$\Delta E = -\langle \frac{dE}{dx} \rangle \cdot \rho \cdot d \tag{5}$$

[2] In some versions of PyTT the temperature changes were summed, not the energies. Although algebraically equivalent, this has been found to lead to much less stable simulations. The reason is not yet understood.

where ρ is material density. Substituting ΔE in formula 3 by the above formula we get:

$$\Delta T_{beam} = \frac{-\langle dE/dx \rangle \cdot \rho \cdot d}{m \cdot c_p} = \frac{-\langle dE/dx \rangle \cdot d}{V \cdot c_p} \tag{6}$$

where V is the cell volume. As we see the density of the material cancels out in the temperature increase Equation 6.

The Equation 5 is valid for a foil but for a wire crossing the beam an additional correction factor, due to circular cross-section of the wire, should be taken into account. From purely geometrical considerations this factor should be $\pi/4$.

4.1.1 Geant4 simulations

Charged, heavy particles deposit the energy in the material mainly due to the interaction with its electrons. This process is well described by Bethe-Bloch equation:

$$-\langle \frac{dE}{dx} \rangle = \frac{4\pi}{m_e c^2} \frac{nz^2}{\beta^2} (\frac{e^2}{4\pi\epsilon_0})^2 \left[ln \frac{2m_e c^2 \beta^2}{I(1-\beta^2)} - \beta^2 - \frac{\delta^2}{2} \right] \tag{7}$$

The $\langle \frac{dE}{dx} \rangle$ is an average rate at which a material absorbs the kinetic energy of a charged particle. It depends on charge of the impacting particle z, its velocity β and electron density of the material n. It is expressed in $[MeV \cdot g/cm^2]$ because it is normalized to the material density. This allows for simple computation of stopping power for mixed-material media. In order to get total energy deposited in a target the Equation 5 should be used.

Validity of Bethe-Bloch formula extends from about 0.5 MeV to TeVs. This is why the PSTAR database uses fitting-formulas are used which are based on experimental stopping power data for proton energies below 0.5 MeV. One should note that the Bethe-Bloch does not take into account the particle interaction with nucleus, either electromagnetic or in form of nuclear reactions. It also does not take into account the production of δ-electrons which can carry away significant amount of energy deposited by the protons. For very thin targets, where the number of interactions with electrons is small, Vavilov or Symon theory are used to calculate energy loss distribution.

Modern Monte-Carlo tools can be used to calculate energy deposit in thin targets. Geant4 and FLUKA simulations have been reported multiple times (ref). The results presented here were obtained using Geant4.10.4 assuming the following definition of wire material:

```
G4Material* wiremat = new G4Material("Graphite", 6, 12.01*g/mole, 2.1*g/cm3);
```

The simulation step has been limited to 0.05 µm. Figure 1 shows a comparison of Bethe-Bloch with density correction, PSTAR values (which overlap with Bethe-Bloch) and results of Geant4 run for proton beam in energy range 80 MeV - 20 GeV. The Geant4 simulations are done using `QGSP_BERT` or `EM Standard Physics option 4` physics lists. They both displayed very similar results. The beam definition is a pencil beam passing exactly through the wire centre. For each data point 1 million protons have been simulated. Results were binned into a histogram and a mean value has been calculated. Because of the cylindrical shape of the wire cross section the factor of $\pi/4$ is applied [3]. As the PSTAR database contains only graphite with density of 1.7 g/cm^3, factor 2.1/1.7 is used.

The discrepancy is usually explained by δ-electrons carrying away significant portion of deposited energy [Sap08]. It is much smaller at low energy than at high. Figure 2 presents the dependence of the correction coefficient on the beam energy and on wire thickness. The energy dependence is somehow complex and requires further investigation. The wire thickness dependence has been fitted with power function and exponent is of the order of 0.03.

Similar simulations have been performed with FLUKA and the results were consistent with Geant4.

4.2 RF heating

The wire target is in machine vacuum and it has to cross the beam. Therefore, it is a subject of the RF fields generated by the beam and leaking from the RF-cavities of the accelerator. Figure 4 shows

[3] Another Geant4 simulations where, instead of pencil beam, a distribution uniform across the wire diameter is used, confirmed validity of this assumption for 34 µm wire.

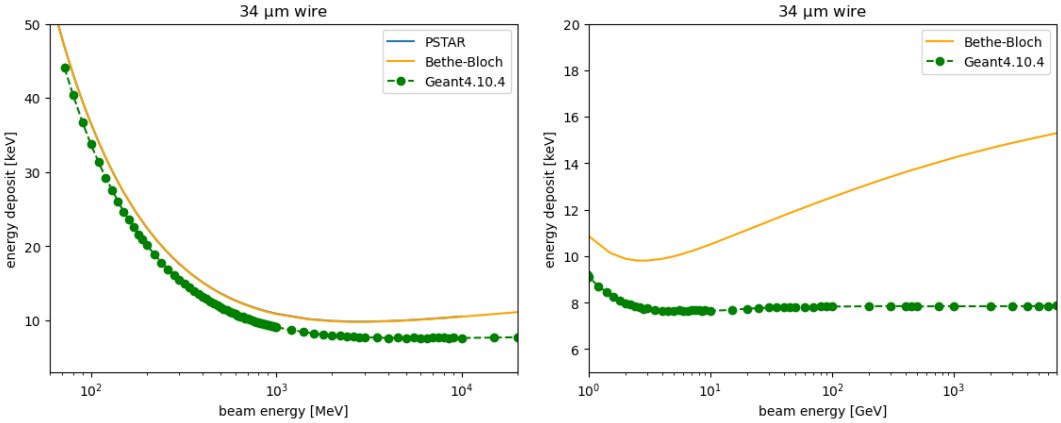

Figure 1: Comparison of Geant4 simulations with Bethe-Bloch formula for low beam energy (left plot) and for energies above 1 GeV (right plot).

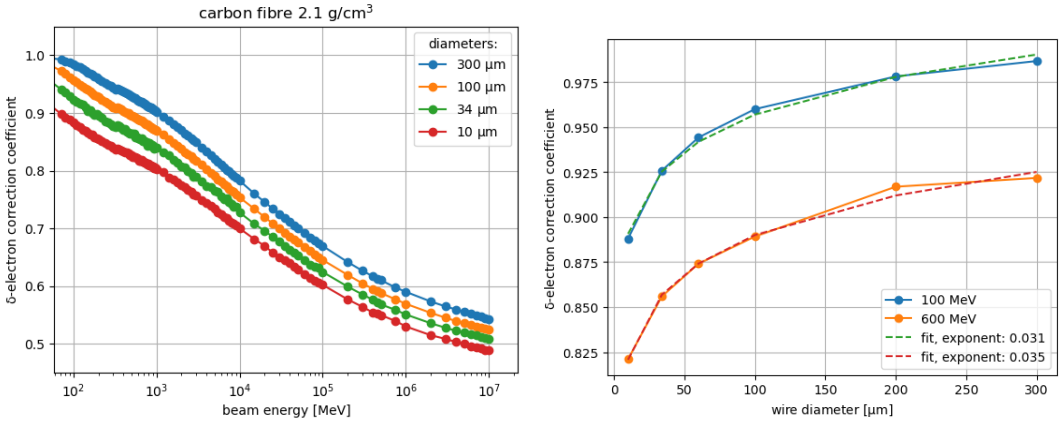

Figure 2: Correction coefficient describing the fraction of energy deposited in the wire as a function of beam energy (left plot) and as a function of wire radius (right plot).

proton energy	Bethe-Bloch [keV]	Geant4 [keV]	ratio
72 MeV	46.897	44.231	0.943
100 MeV	36.551	33.840	0.926
600 MeV	12.664	10.840	0.856
1 GeV	10.862	9.114	0.839
450 GeV	13.702	7.841	0.527
7 TeV	15.291	7.869	0.515

Table 1: Energy deposited in a carbon wire of 34 µm diameter and 2.1 g/cm^3 density calculated using standard Bethe-Bloch formula with density correction and Geant4 for chosen proton energies.

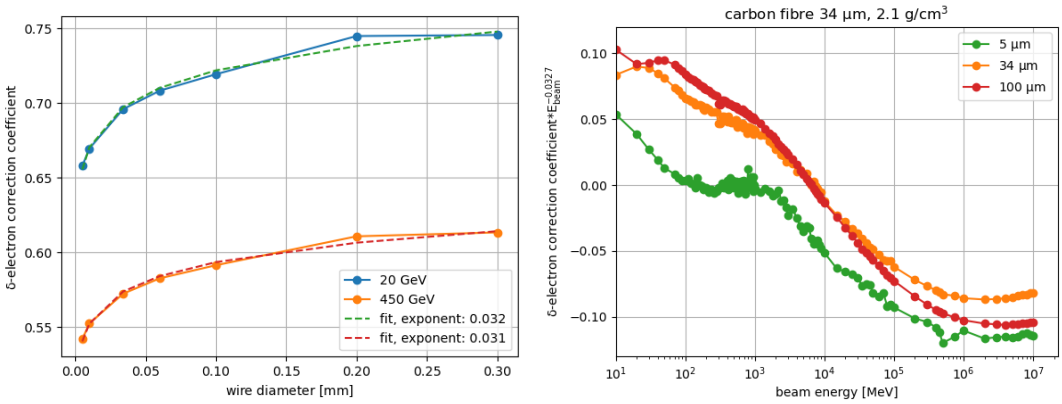

Figure 3: Left plot: correction coefficient describing the fraction of energy deposited in the wire as a function of wire radius for high energy beams. Right plot shows attempt to find regularity in correction coefficient energy dependence.

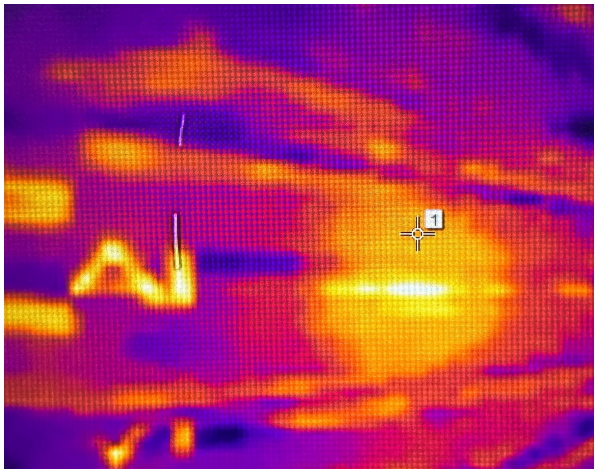

Figure 4: RRL wires during the scan in PSI Main Ring glowing due to RF coupling. The picture was taken with FLIR AX8 infrared camera.

image of the inside of PSI Main Ring Cyclotron vacuum chamber during RF commissioning without the beam. Three carbon fibers, one vertical and two at $\pm 45°$ are seen glowing in the left part of the image. The ends of the wire seem to glow stringer than the centre.

Calculation of RF coupling with the wakefield generated by LHC beam was reported in [MS08].

4.3 δ electrons

4.4 Radiative cooling

Radiative cooling is a process that involves the emission of thermal radiation from a surface. The spectrum of this radiation is described by a Planck law:

$$u(\nu, T) = \frac{8\pi h\nu^3}{c^3} \frac{1}{exp(\frac{h\nu}{k_B T}) - 1} \tag{8}$$

The total power removed from a hot body by radiative cooling is described by Stephan-Boltzmann

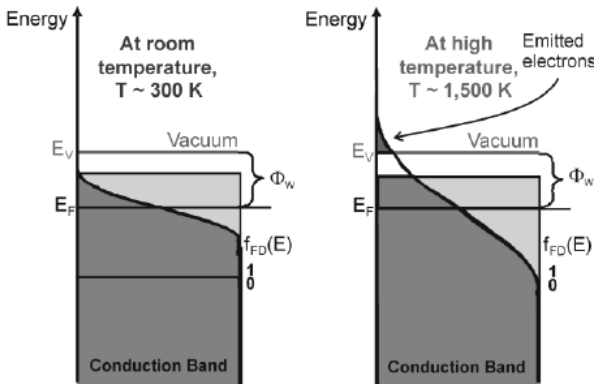

Figure 5: Illustration of thermionic emission mechanism [Tuo23].

law:

$$\Delta E(T) = \frac{8\pi h\nu^3}{c^3}(T^4 - T_0^4) \tag{9}$$

(correct)

4.5 Thermionic emission

Thermionic emission is a complex phenomena which takes place when the material reaches high temperatures. A fraction of the electrons, present in the material, are not bound to atoms and behave as a free gas. The density of those electrons in metals is very large allowing them to be good conductors. Graphite is also a good conductor. The electrons are fermions and their energy spectra follow follow the Fermi-Dirac distribution:

$$n_i = \frac{1}{1 + exp(\frac{\epsilon_i - \mu}{k_B T})} \tag{10}$$

where μ is Fermi level or chemical potential. The thermionic emission takes places if some of the electrons at high temperatures reach energies allowing for escape from the material to the vacuum. This is illustrated in Figure 5 from [Tuo23]. The energy difference between Fermi level E_F and escape energy (vacuum level) is called work function Φ_W.

The emitted current density is described by Richardson-Dushmann formula:

$$J_{th} = \lambda_R A_R T^2 exp(-\Phi_W/kT) \tag{11}$$

Where A_R is Richardson constant equal to $4\pi mk^2 q_e/h^3$=1.20173×10^6 A m^{-2} K^{-2}. λ_R is a material-dependent coefficient between 0 and 1.

Removal of the most energetic electrons from the material is a very efficient cooling mechanism. Total energy loss is expressed by the following equation:

$$\Delta E_{th} = A_{rad} J_{th}(\Phi_W + 2kT) \tag{12}$$

where A_{rad} is total radiative surface.

4.5.1 Kinetic energy of thermionic electrons

Emitted electrons have a very small kinetic energy E_k. Equation 1.96 from [Mod84] gives:

$$j_e(E_k) \propto E_k \cdot exp(-E_k/kT) \tag{13}$$

Figure 6 shows thermionic electron spectra for various temperatures of the emitting carbon fibre. The energies are low, non-relativistic, therefore a classical formula $E_k = mv^2/2$ can be used to calculate electron velocities. This is shown in Table 2, where dynamic properties of the electrons with most probable kinetic energy are shown. At 2000 K the speed of the electrons is much higher than wire

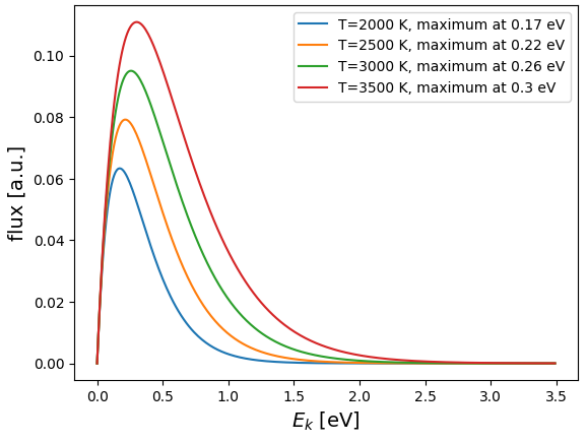

Figure 6: Kinetic energy distribution of thermionic electrons for various temperatures of the emitter.

temperature [K]	E_k [eV]	v_e [m/s]	β_{rel}
2000	0.17	244540	0.00082
2500	0.22	278187	0.00093
3000	0.26	302422	0.00101
3500	0.30	324852	0.00108

Table 2: Thermionic electron properties at various target temperatures.

speed, therefore the scanner movement can be neglected and the calculations can be conducted as in a stationary approach.

4.5.2 Space charge limited thermionic emission

At high thermionic currents, the amount electrons emitted to the vicinity of the wire form a cloud dense enough, to create, at the wire surface, electric field which reduces the amount of emitted electrons. This effect, well known to the designers of high-power electron guns, is called space-charge limited thermionic emission.

Here we assume a steady-state (quasistatic) conditions:

- $j_{th} = const.$ - thermionic current density is constant, because scan speed is much slower than the velocities of the thermionic electrons;
- the wire temperature is constant.

The electrons are assumed to be emitted equally in all directions perpendicular to the wire sequence. Therefore, electrons emitted at a given moment t_i, constitute a cylinder with center lying in the wire center. The radial electric field generated by this "cylinder of charge" at the surface of the wire is:

$$E_r = \frac{\rho r_w}{2\epsilon_0} \tag{14}$$

where:

- ρ is charge density in the cylinder,
- r_w is the wire radius,
- ϵ_0 is the vacuum permittivity.

The charge density can be expressed as:

$$\rho = \frac{Q_i}{2\pi R_i \Delta R l} = \frac{j_{th} l 2\pi R_i \Delta t}{2\pi R_i \Delta R l} = \frac{j_{th}}{v_e} \tag{15}$$

where l is length of the wire or the analysed wire segment. Therefore, the

4.6 Conductive cooling

4.7 Sublimation

4.8 Secondary electron emission

SE current
    cooling effect

4.9 Benchmarking

The code has been benchmarked against other, established codes and against experiments. One of the first benchmarks against a different code was an effort by Roberto Rocca (CERN) who checked the temperature evolution against ANSYS code. The report of this work was never published, but the results were consistent in the temperature range below thermionic emission threshold.

A series of experimental benchmarks were done by Araceli Navarro and documented in her PhD thesis [Nav23]. This work focused on CERN Linac4 SEM grids and the main observable was secondary emission current.

Manon Boucard in her MSc thesis [Bou23] compared the secondary and thermionic currents measured in PS Long Radial Probe with PyTT (v2) simulations, most of the time finding a good agreement.

Finally, another validation against other simulation packages has been done by Abdelrahman Abouelenain who compared PyTT (v2) to COMSOL [Abo23].

5 Code structure

The `PyTT` code directory contains the following subdirectories:

- BeamDescription/
- Doc/ (rename to Documentation or move to HelpFolder)
- EneDepData/
- HelpFolder/
- Literature/ (split to subjects, eg. RF coupling)
- MaterialInfo/
- Modules/
- Output/
- ParticleInfo/
- QtInterface/
- Simulations/
- Tests/

Block diagram of Modules!

6 Usage

The simulation GUI can be launched by:

```
> python3 MAIN.py
```

Then a configuration file can be loaded using the upper left field (`Parameters File`). After uploading the configuration file, all the fields of the GUI are filled up. The next step is to correct some fields if needed and then launch the simulation by pressing the bottom right button (`Simulate`). The simulation progress can be observed at the terminal used to launch the GUI. At the end of the simulation the GUI crashes, however three files are saved in `Output` directory:

- `InputFileUsedForSimulation.txt`
- `MaxTempVSTime.txt`
- `IntensityVSTime.txt`

The simulation can be launched without GUI by calling:

```
> python3 MAIN.py Simulation/ConfigurationFile.txt
```

where the configuration file must be prepared in advance (see example in the next Section). This is a preferred way to do simulations (now, May 2024). The process produces file `Last.txt` which contains all crucial results together. To analyse this file, use for instance `plot_MaxT.py` script.

6.1 Configuration file

```
# ---------------- Input File for PyTT_3.0.0  ----------------- #
#      There must be at least one space between name and value  #
# ------------------------------------------------------------ #


# ------------ Define Beam ------------- #
Particle: Proton # From ParticleInfo/Proton.txt
Energy: 123     # [MeV] maybe needed to compute dEdx
BeamType: Gaussian
sigx: 1.6337e-3 # [m] beam sigma along the scan
sigy: 1.4016e-3     # [m] vertical beam sigma
x0: 0.0     # [m] beam at the centre
y0: 0.0     # [m]
tpulse: 1     # [s] Always needed
BCurrent: 1.5e-3 # [A] units from Qt interface (1.8e-3)  1.8 mA
Npart: -        # Either number of particles with frequency (synchrotron)
frec: 1
# ---------------- Define Detector ----------------- #
Material: Carbon
DetType: WIRESCAN
# ------------ Parameters for WIRESCAN ------------- #
WIRESCAN_Plane: Horizontal
WIRESCAN_IniPos: -2.96e-2   # [m]
WIRESCAN_EndPos:  2.96e-2   # [m]
WIRESCAN_wWidth: 33e-6      # [m] wire diameter
WIRESCAN_wLength: 2e-2      # [m] wire length considered in simulation
WIRESCAN_wRes: 2.5e-4       # [m] wire segment size
WIRESCAN_Type: 1            # legacy method
WIRESCAN_wSpeed: 0.0297     # [m/s] wire speed
WIRESCAN_Npulses: 1         # when wire crosses beam multiple times
# ------------ Temp Simulation Parameters -------------- #
TempSIM: Yes
T0: 1030                    # inital and environmetal temp
dtPulse: 5e-4
dtCooling: 5e-4
EnableParameterVariation: Yes
RadiativeCooling: Yes # Crucial in typical temp range
ThermoionicCooling: Yes # Crucial for very high temperatures
ConductiveCooling: No # Small imact
SublimationCooling: No      # Special case
# --------------- Int Simulation Parameters ------------ #
IntSIM: Yes
mu: 0.0    # fraction of electrons stopped in targe (eg. H- beam)
eta: 0.0   # fraction of protons stopped in the target
BSp: 0.0   # fraction of backscattered protons
BSe: 0.0   # fraction of backscattered electrons (eg. H- beam)
# ---------- Debugging -------------------------------- #
Debug: None                 # options: None, Beam, Edep, Thermionic
```

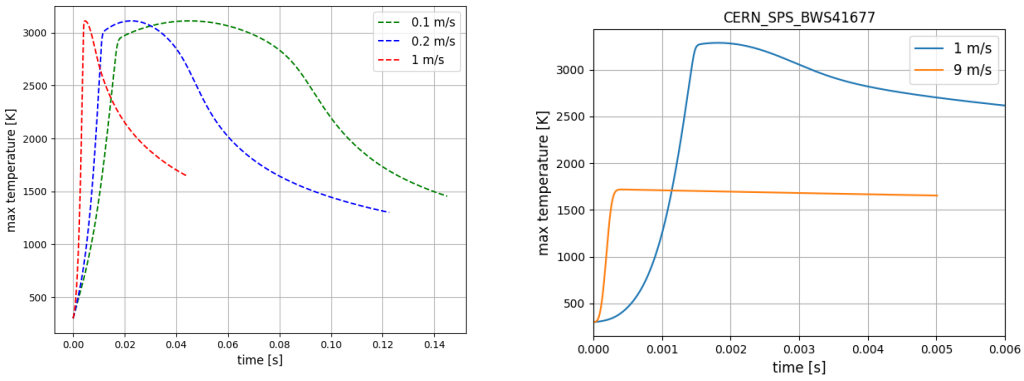

Figure 7: Two simulations of various wire scanners in SPS. The left one is made to compare with Fig. 10 of [Sap08]. The right one uses a recent set of parameters and high-speed scanners.

```
# ------------------------------------------------- #
```

The above file is for version 3.0.0 of the code for wire scanner simulation case.

6.2 Examples of simulations

Over the years, various versions of the code were used for simulation of various beam devices. Historically the first simulations were done for CERN SPS wire scanners, to understand the results of the wire damage tests [MS10]. It was also applied to LHC scanners, J-PARC grids and scanners, CERN Lina4 SEM-grids and various devices at PSI (SEM-grids, wire scanners, beam splitter). Below we present basic results of the following cases:

- SPS wire scanners
- PSI long radial probe
- PSI wire scanner in p-channel

In Table 3 the main simulation parameters of those cases are shown.

Table 3: Main parameters of the simulations.

simulation name	**beam energy** [MeV]	**beam current [mA]**	σ_H **[mm]**	σ_H **[mm]**	**material**	**diam.** [μm]	**wire speed** [m/s]
SPS 2008	450 GeV	139.3	1.63	0.65	CF	30	0.1,0.2,1
SPS BWS41677	450 GeV	150	0.46	0.54	CF	33	1,9
PSI RRL	123 MeV	1.5	1.63	1.4	CF	33	0.03

6.2.1 SPS Wire scanner

Two SPS wire scanner simulations are presented here. The first one is a repetition of the simulation done in 2008 which was using C++ code. The old results are shown on Fig. 10 in [Sap08] and new ones in the left panel of Fig 7 . Quite good agreement is found between this simulation from 2008 and the current one (2024).

The second simulation is a simulation of particular wire scanner number 41677. The beam parameters are different in this location and also faster wire scanner movement is used, as a fast scanner is now installed in that location. The reason for this investigation was a bug found in previous version of `PyTT` code, which produced high temperatures for fast-moving wire scanners.

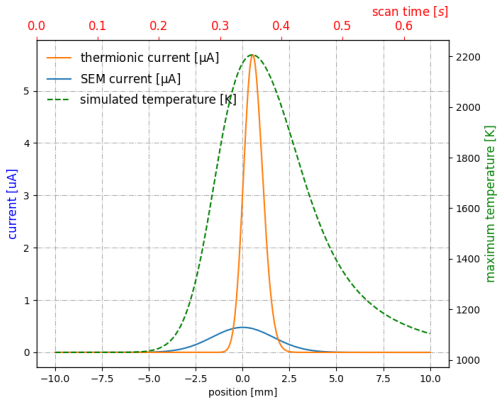

Figure 8: Simulation of thermal behaviour of PSI, Main Ring Radial Probe (RRL).

6.2.2 PSI Long Radial Probe in Main Ring

Long Probes are types of wire scanners which move across all, or almost all orbits of a cyclotron. The long probe in PSI Main Ring is called RRL. It travels across radius of the machine, which is 2.5 meters and crosses about 182 orbits. Because of very long travel distance and the required precision, the speed is only 3 cm/s.

During the acceleration the beam energy increases and its transverse size decreases. In addition the beam in a cyclotron undergoes betatron osciallations as in other machines. Therefore, it is not easy to predict which orbit leads to the strongest heating of the wire. In case of RRL it has been found that orbit 16, where beam has an approximate energy of 123 MeV, often gives the highest thermionic peak.

Detailed analysis of the beam profile and simulation of thermal effects can be found in [Bou23, BS23]. Here the simulations are repeated with the last version of the code and shown in Fig. 8. It can be comapred to Fig. 6.10 of [Bou23].

Contents

References

[Abo23] Abdelrahman Abouelenain. *Beam wire scanner at CERN: Simulation of behavior of carbon nanotube wires with protons at high energies*. Msc project report, Universite de Franche-Comtè, Besancon, October 2023.

[AN23] M. Sapinski A. Navarro. pyTT. https://github.com/ClaudePP/PyTT, 2023. [Online; accessed 19-July-2023].

[Bou23] Manon Boucard. *Thin Targets in Extreme Conditions: Probing High-Brightness Hadron Beams*. Msc thesis, EPFL Lausanne, September 2023.

[BS23] Manon Boucard and Mariusz Sapinski. Dealing with thermionic emission in wire scanners based on secondary electron emission. *JACoW*, IPAC2023:THPL150, 2023.

[dBP17] D.W. Posthuma de Boer and A. Pertica. Thermal Simulations of Wire Profile Monitors in ISIS Extracted Proton Beamline 1. In *Proc. of International Beam Instrumentation Conference (IBIC'16), Barcelona, Spain, Sept. 13-18, 2016*, number 5 in International Beam Instrumentation Conference, pages 786–788, Geneva, Switzerland, Feb. 2017. JACoW. doi:10.18429/JACoW-IBIC2016-WEPG59.

[Doe] Rudolf Doelling. personal communication.

[Hu] Martin Hu. personal communication.

[Mod84] A. Modinos. *Field, Thermionic and Secondary Electron Emission Spectroscopy*. 1984.

[MS08] T. Kroyer M. Sapinski. Operational limits of wire scanners on lhc beams. *Beam Instrumentation Workshop, May 2008*, 2008.

[MS09] A. Guerrero J. Koopman E. Metral M. Sapinski, B. Dehning. Carbon fiber damage in accelerator beam. *proceedings of DIPAC 2009, CERN-BE-2009-028*, 2009.

[MS10] A. Guerrero M. Meyer T. Kroyer M. Sapinski, B. Dehning. Carbon fiber damage in particle beam. *46th ICFA Advanced Beam Dynamics Workshop on High- Intensity and High-Brightness Hadron Beams, Morschach, Switzerland, MOPD61*, 2010.

[Nav23] Araceli Navarro. *Understanding Secondary Emission Process and Beam Matter Interactions for Optimization of Diagnostic Wire Grid System in Particle Accelerators*. Phd thesis, Universita Politecnica de Catalunya, March 2023.

[Sap08] M. Sapinski. Model of carbon wire heating in accelerator beam. *CERN AB-2008-030-BI report*, 2008.

[Tuo23] Tomi Tuohimaa. *Liquid-Jet-Target Microfocus X-Ray Sources: Electron Guns, Optics and Phase-Contrast Imaging*. PhD thesis, Royal Institute of Technology, Stockholm, Sweden, 04 2023.

[YIL+23] Emi Yamakawa, Yoshihiro Ishi, Alan Letchford, Shinji Machida, Alex Pertica, David Posthuma de Boer, and Tomonori Uesugi. Study of Single Wire Scanner Monitor for FETS-FFA Test Ring. *JACoW*, IBIC2023:WEP019, 2023.