

Fredrik Sommerfelt Grønvold

Modelling of Fast Beam-Ion Instability in Future Accelerators

Hovedoppgave i MIIK

Veileder: Morten Hovd

Medveileder: Lotta Mether

Januar 2022

Fredrik Sommerfelt Grønvold

Modelling of Fast Beam-Ion Instability in Future Accelerators

Hovedoppgave i MIK
Veileder: Morten Hovd
Medveileder: Lotta Mether
Januar 2022

Norges teknisk-naturvitenskapelige universitet
Fakultet for informasjonsteknologi og elektroteknikk
Institutt for teknisk kybernetikk



Kunnskap for en bedre verden

Preface

This report is the culmination of work done for the course "TTK4900 - Engineering Cybernetics, Master's Thesis" and is a continuation of the work done in the course "TTK4551 - Engineering Cybernetics, Specialization Project" during the autumn of 2022. The work was completed during my Technical studentship at the Coherent Effects and Impedance (CEI) section in CERN's Beams Department (BE) which started in January 2023. The technical student program is a CERN initiative, where students from member and associate member states travel to CERN for a period of 6 to 12 months, and not exceeding 14 months. The CEI section is a part of the Accelerator and Beam Physics (ABP) group in the BE and oversees the study of beam coherent collective effects and stability. Furthermore, the section provides expertise in the field of beam coupling impedance computation, optimization and specification.

Acknowledgements

I would first and foremost like to thank my supervisors Prof. Morten Hovd (NTNU) and Lotta Mether (CERN) for their guidance and support throughout the project.

Thank you to Lotta Mether for granting me this opportunity to come and conduct my master project at CERN. You have given me excellent assistance during this project, both on technical and theoretical matters.

Thank you to Prof. Morten Hovd for taking me on as your student already from the beginning of the specialisation project in August of 2022. Your feedback during our monthly meetings and the corrections you have provided for this report has been much appreciated.

I would also like to thank everyone in the Accelerator and Beam Physics group at CERN's Beams Department, especially the members of the Coherent Effects and Impedance section, for making my technical studentship such a wonderful experience.

Finally, I would like to thank my family and my girlfriend for all their love, support, and encouragement.

Executive summary

Future accelerators such as the compact linear collider (CLIC) are expected to be vulnerable to the fast beam-ion instability (FBII) phenomenon due to their high beam intensity and small transverse beam size. Unlike classical ion instability, FBII manifests along the length of the bunch train which means that it can arise in linear accelerators (LINAC) as well as circular accelerators. Ions are generated through the beam's interaction with the gases present in the beam duct and induce a coherent motion of the bunches which causes the beam to become unstable. The numerical macroparticle modelling tools PyECLOUD and PyHEADTAIL have previously been used to study FBII in the CLIC damping ring.

In this project the numerical modelling tools have been applied for studying the effects of residual gas pressure in the CLIC main LINAC. Simulation setups with multiple species gas composition have been tested for the CLIC damping ring and the ion generation process has been updated to handle electric field ionisation. The results from the damping ring simulations showed good alignment with theory. However, further investigation are needed to before successful simulation studies of FBII in the CLIC main LINAC can be achieved. To that end, the project has laid the foundation for continued development and future pressure studies in the CLIC main LINAC.

Table of Contents

Preface	i
Acknowledgements	ii
Executive summary	iii
List of Tables	vi
List of Figures	ix
Glossary	x
Abbreviations	xii
Symbol	xiii
1 Introduction	1
1.1 Background	1
1.2 Problem Description	4
1.3 Project Objectives	5
1.4 Specialisation Project Results and Findings	5
1.5 Thesis Structure	5
2 Theory	7
2.1 Concepts from Linear Beam Dynamics	7
2.1.1 Beam Coordinates, Trace Space, and Equations of Motion	8
2.1.2 Particle Beam Properties	10
2.1.3 Transfer Matrix and Beam Motion	13
2.2 Ionisation mechanisms	14
2.2.1 Scattering Ionisation	14
2.2.2 Electric Field Ionisation	15
2.3 Fast Beam-Ion Instability	17

2.3.1	Mitigation Strategies	19
2.4	Overview of Numerical Modelling Tools	20
2.4.1	FASTION	23
2.4.2	Python Tools	24
3	Application of Python Tools for FBII Studies	26
3.1	Simulation Setup for a Linear Accelerator	26
3.1.1	Dual Grid Setup	27
3.1.2	Multiple Gas Species Setup	29
3.2	Implementing E-Field Ionisation	29
3.3	Pressure Studies in the CLIC Main LINAC	33
3.4	Multiple Gas Species in the CLIC Damping Ring	34
4	Results	36
4.1	Electric field ionisation in PyECLOUD and FASTION	36
4.2	CLIC main LINAC simulation results	37
4.2.1	Without Electric Field Ionisation	37
4.2.2	With Electric Field Ionisation	42
4.3	Multiple gas species in CLIC DR	48
5	Discussion	49
5.1	Ion Generation in FASTION and PyECLOUD	49
5.2	Simulation Results	50
5.2.1	Results from the CLIC Main LINAC	50
5.2.2	Results from the CLIC DR	50
5.3	Some Assumptions and Approximations	51
5.4	Challenges	51
6	Future Work	54
7	Conclusion	56
	References	57
	Appendix	61
A	Linear Beam Dynamics	62
B	Example of Simulation_parameter.input file	65
C	Example of CO ₂ Cloud file	65

List of Tables

2.1	Magnetic pole contribution and functionality[1]	8
2.2	M^2 , C , and σ^{ion} for typical species when beam energy of 9 GeV [2]. . .	15
2.3	Electric field threshold for selected gas species [3].	16
2.4	Example of TWISS file for the CLIC Main LINAC	24
3.1	CLIC Main LINAC Parameters and Values	33
3.2	Gas Species and Atomic Mass Number	34
3.3	CLIC DR Parameters and Values (NOT UPDATED)	35
3.4	Pressure Combinations for Multiple Gas Species Studies	35

List of Figures

1.1	Baseline luminosities as a function of centre of mass energy for each of the four electron positron collider projects. The energy levels for the formation of the Z, W, and Higgs boson and the top quark are indicated along the FCC-ee baseline plot. Source: [4]	2
1.2	Map showing the footprint of the LHC and that of the CLIC with energy stages 380 GeV, 1.5 TeV and 3.0 TeV in the border region between Switzerland and France at the CERN site close to Geneva. Source: [5]	3
2.1	The co-moving Cartesian coordinate system for the motion of particle q in a circular accelerator.	9
2.2	The emittance is directly related to the area of the ellipse in the trace plane. Adapted from [1]	11
2.3	The upper figure shows the evolution of the betatron oscillation envelope $E(s)$. All particle trajectories lie within this envelope. The lower graph shows the emittance plot corresponding to the envelope. Although the shape of the emittance plot changes, the value of the emittance remains constant in accordance with Liouville's theorem. Adapted from [1] and [6].	12
2.4	Motion from position s_0 to s_1 on the orbit which is described by the transfer matrix	13
2.5	The ionisation area swept by the beam is the elliptical area determined by $\sigma_{x_{fion}}$ and $\sigma_{y_{fion}}$. The shaded area contains gas particles that are able to diffuse into the ionisation area during the bunch spacing.	16
2.6	The distribution of ions generated by the electric field ionisation of residual gas.	16
2.7	The shaded area contains ions which are able to reach the ionisation area	17
2.8	Diagram of a bunch-by-bunch feedback system. Adapted from [7]	20
2.9	The accelerator is divided into interaction points along the accelerator where the beam and environment is modelled. The beam dynamics from one IP to the next is done with the transfer matrix $(M(\cdot))$ defined by the accelerator's optics parameters.	21

2.10	The flowchart shows the steps done at each IP. Adapted from []	22
2.11	First few lines from a FASTION configuration file for a multispecies simulation with H ₂ , CO, CO ₂	23
2.12	The machine is modelled as a list of alternating machine and cloud objects created by PyHEADTAIL and PyECLOUD respectively.	25
3.1	The machine modelled is changed to also contain a energy update object between the machine and cloud objects created by PyHEADTAIL and PyECLOUD.	27
3.2	Nested grid setup.	28
3.3	Extract from <code>PyEC4PyHT.py</code> showing how the interface would cancel the simulation in the presence of multiple cloud files.	29
3.4	The machine object list contains a cloud object for each of the gas species included in the simulation study.	30
3.5	The flowchart shows how the ionisation mechanism is determined for the ion generation in PyECLOUD.	32
3.6	Parameters for creating the single grid setup used for FBII studies of CLIC main LINAC.	33
3.7	Parameters for creating the dual grid setup used for FBII studies of CLIC main LINAC.	34
4.1	Ion limits	36
4.2	Number of ions generated in FASTION and PyECLOUD over the length of the CLIC main LINAC.	37
4.3	The left column shows the bunch train at IP:2005 for a simulation using a single grid. The middle column shows the same simulation using a dual grid. The two simulations were run without electric field ionisation activated in PyECLOUD. The column on the right shows the bunch train for a simulation with A: 132 and P: 5.0 nTorr in FASTION.	38
4.4	The left column shows the evolution of bunch 300 along the length of the LINAC for a simulation using a single grid. The middle column shows the same simulation using dual grid. The two simulations were run without electric field ionisation activated in PyECLOUD. The column on the right shows the bunch evolution for a simulation with A: 132 and P: 5.0 nTorr in FASTION.	39
4.5	The left column shows evolution of the bunch transverse size along the length of the LINAC for a simulation using a single grid. The middle column shows the same simulation using dual grid. The two simulations were run without electric field ionisation activated in PyECLOUD. The column on the right shows the bunch transverse size evolution for a simulation with A: 132 and P: 5.0 nTorr.	40
4.6	The two plots show the maximum offset of the bunch centroid over pressure for one simulation using a single grid and one using a dual grid. The simulations were run without electric field ionisation activated in PyECLOUD.	41

4.7	The left column shows the bunch train at IP:2005 for a simulation using a single grid. The middle column shows the same simulation using a dual grid. The two simulations were run with electric field ionisation activated in PyECLOUD. The column on the right shows the bunch train for a simulation with A: 132 and P: 5.0 nTorr in FASTION.	42
4.8	The left column shows the evolution of bunch 300 along the length of the LINAC for a simulation using a single grid. The middle column shows the same simulation using dual grid. The two simulations were run with electric field ionisation activated in PyECLOUD. The column on the right shows the bunch evolution for a simulation with A: 132 and P: 5.0 nTorr in FASTION.	43
4.9	The left column shows evolution of the bunch transverse size along the length of the LINAC for a simulation using a single grid. The middle column shows the same simulation using dual grid. The two simulations were run with electric field ionisation activated in PyECLOUD. The column on the right shows the bunch transverse size evolution for a simulation with A: 132 and P: 5.0 nTorr	44
4.10	The column on the left show the maximum offset of the bunch centroid in x and y for all the gas species listed in Table 3.2 over pressure for one simulation using a single grid with electric field ionisation activated in PyECLOUD. The column on the right shows the results for the same simulation done in FASTION [8].	45
4.11	The left column shows the bunch train at IP:2005 for a simulation using a single grid, where the gas has atomic mass number of 132 and a pressure 5.0 nTorr. The column on the right shows the results for the same simulation done in FASTION [8].	46
4.12	The left column shows evolution of the bunch transverse size along the length of the LINAC for a simulation using a single grid, where the gas has atomic mass number of 132 and a pressure 5.0 nTorr. The column on the right shows the results for the same simulation done in FASTION [8].	47
4.13	The left column shows the bunch train in x and y at IP: 258 in the CLIC DR for the pressure sets 1, 2, and 3, found in Table 3.4. The middle plots pressure sets 4, 5, and 6, while column on the right shows the bunch train for pressure sets 7, 8, and 9.	48
7.1	Motion from position s_0 to s_1 on the orbit which is described by the transfer matrix	62

Glossary

Accelerator	A machine that accelerates charged particles to relativistic speeds.
Boson	A subatomic particle whose spin quantum number has an integer value.
Brightness	Number of charged particles over unit area.
Bunch	Collection of particles.
Bunch train	Collection of consecutive bunches.
c.o.m energy	Analogous with collision energy.
Coherent motion	The motion of the bunch centroid.
Collider	Type of accelerator that brings the particle beams to interaction points where they collide
Collision energy	The combined energy of the two colliding beams. This is an important parameter as it allows for the analysis of the collision outcome, since the total energy of the detected particles from the collision has to equal the collision energy through conservation law.
Emittance	The space taken up by the particles position in the position-momentum phase-space/position-angle trace space.
Hadron	Particle that consists of quarks.
Higgs factory	A particle collider that produces Higgs bosons
Intensity	Also called beam current and is the number of particles that passes through a cross-section over unit time.
Luminosity	An indicator of the number of collisions between particles in the interaction point of a particle accelerator.
Muon	Fundamental subatomic particle described by the standard model.
Quark	Fundamental subatomic particle described by the standard model.

Standard Model

A physics model that explains how the basic building blocks of matter interact, governed by four fundamental forces.

Synchrotron

An accelerator that is circular and synchronises the energy and radius.

Abbreviations

ALS	Advanced Light Source
ATLAS	A Toroidal LHC Apparatus
BPM	Beam Position Monitor
BSM	Beyond the Standard Model
CERN	Conseil Européen pour la Recherche Nucléaire (European Organization for Nuclear Research)
CLIC	Compact Linear Collider
CMS	Compact Muon Solenoid
c.o.m	Centre of Mass
DR	Damping Ring
ESPP	European Strategy for Particle Physics
FBII	Fast Beam-Ion Instability
FCC	Future Circular Collider
HL-LHC	High Luminosity Large Hadron Collider
ILC	International Linear Collider
LHC	Large Hadron Collider
LINAC	Linear Accelerator
LURE	Laboratoire pour l'utilisation du rayonnement électromagnétique (Laboratory for the Use of Electromagnetic Radiation)
NTNU	Norwegian University of Science and Technology
PIC	Particle-in-Cell
RF	Radio Frequency
SM	Standard Model
SOLEIL	Source optimisée de lumière d'énergie intermédiaire du LURE (LURE optimised intermediary energy light source)
UHV	Ultra High Vacuum

Symbols

A	Atomic mass number [u]
B	Magnetic field
$\beta_0 = \frac{v}{c}$	Relativistic velocity
C	Coulomb
$c = 2.99793 \times 10^8$	Speed of light in vacuum [$\text{m} \cdot \text{s}^{-1}$]
ΔT_b	Bunch spacing
E	Energy
E	Electric field
eV	Electron volt
$e = 1.6021766 \times 10^{-19}$	Elemental charge [C]
e^-	Electron
e^+	Positron
$m_e = 9.10938370 \times 10^{-31}$	Electron mass [kg]
$\varepsilon_x, \varepsilon_y, \varepsilon_z$	Emittance in the horizontal, vertical and longitudinal plane
$\varepsilon_0 = 8.8542 \times 10^{-12}$	Dielectric constant [$\text{C}^2\text{J}^{-1}\text{m}^{-1}$]
F	Force
γ	Relativistic gamma function
$\gamma = \frac{1}{\sqrt{1-\beta_0^2}}$	Lorentz factor
\mathcal{H}	Hamiltonian
\hbar	Reduced Plack's constant
$k_B = 1.380649 \times 10^{-23}$	Boltzmann's constant [$\text{m}^2\text{kg} \cdot \text{s}^{-2}\text{K}^{-1}$]
$\lambda_{cur} = \frac{N}{\beta_0 c}$	Line charge density
$\mathcal{L} \propto \frac{1}{\sigma_x \sigma_y}$	Luminosity

m	Mass
m_0	Invariant mass or rest mass
N_b	Number of particles in a bunch
$p_m = 1.66055 \times 10^{-27}$	Proton mass [kg]
R	Radius
$r_p = 1.5458775 \times 10^{-18}$	Proton radius [m]
$\sigma_x, \sigma_y, \sigma_z$	Beam sizes in the horizontal, vertical and longitudinal plane
T	Temperature [K]
$u = 1.66053873 \times 10^{-27}$	Atomic mass unit [kg]
v	Velocity

1

Introduction

1.1 Background

On the 5th of July 2022, the day after the 10 year anniversary of the discovery of the Higgs boson, the Large Hadron Collider (LHC) at CERN's accelerator complex initiated its third operational run (Run 3) after 3 years of upgrades during Long Shutdown 2 (LS2) [9] [10]. The LHC immediately reached collision energies of 13.6 TeV, exceeding its own world record of 13 TeV collisions, and remains the worlds largest and most powerful particle accelerator. During Run 3, which is scheduled to last until 2025, two of the four detectors in the LHC, ATLAS and CMS, are expected to record more collisions than the previous operational runs combined [11].

After the completion of Run 3, the LS3 will usher in the upgrades necessary to transition to the High Luminosity LHC (HL-LHC). The HL-LHC will then operate with luminosity values up to 10 times the luminosity of the LHC until the early 2040s [12].

With the end of the LHC clearly on the horizon, CERN has started to prepare for a post-LHC future. This is reflected in the 2020 update to The European Strategy for Particle Physics (ESPP), which states that although the LHC and the HL-LHC will remain the primary tool for particle physics exploration in the coming decade, the highest priority for future initiatives is an electron-positron "Higgs factory" [13].

The choice of an electron-positron collider as a Higgs factory, as opposed to a hadron collider like the LHC, has to do with the formation mechanisms of the Higgs boson in the collider types. Since electrons and positrons are fundamental particles, while hadrons are composites of fundamental particles, the collisions between electrons and positrons are, from a physics point of view, considered "cleaner". An electron-positron collider would therefore further enable the mapping of interactions between the Higgs boson and other particles [13].

It is a clear consensus among the worldwide particle physics community that a Higgs factory is the path forward in the quest for finding new physics beyond the standard model (BSM). Although the standard model (SM) has so far been able to describe all phenomena in collider experiments, the existence of gravity, matter-antimatter asymmetry, dark matter,

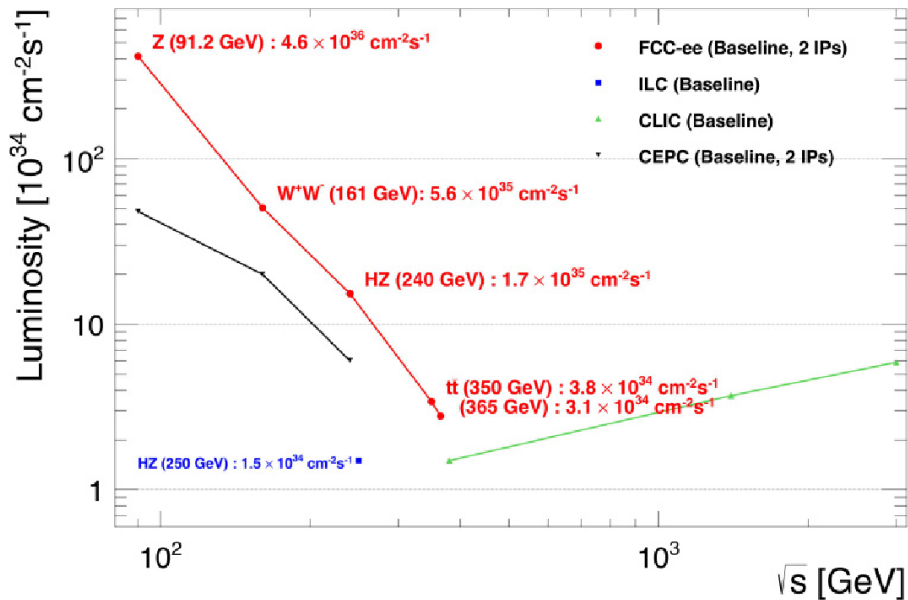


Figure 1.1: Baseline luminosities as a function of centre of mass energy for each of the four electron positron collider projects. The energy levels for the formation of the Z, W, and Higgs boson and the top quark are indicated along the FCC-ee baseline plot. Source: [4]

dark energy, and non-zero neutrino masses all indicate the existence of BSM physics [14].

There are at least four major electron-positron collider designs that are being studied or are approaching maturity across the globe [4]. The baseline luminosities for the four accelerator projects are plotted over their centre-of-mass (c.o.m) energies in figure 1.1 [4].

The International Linear Collider (ILC) project in Japan would have a c.o.m energy of 250 GeV, at which the Higgs-strahlung process is the dominant Higgs production process [15][16]. The 2020 update to the ESPP states that in the eventuality of a timely realisation of the ILC, then the European particle physics community would want to collaborate [13].

The Chinese Circular Electron Positron Collider (CEPC) project is expected to start construction in 2026 to coincide with the 15th Chinese 5 year plan [17]. The 100 km circumference circular collider would be able to provide collisions in the energy range of 90 GeV to 250 GeV, enabling the study of Higgs, Z, and W bosons [4]. The tentative operation is set to 10 years after commissioning in 2035, of which 7 years will be spent on Higgs exploration [17][18].

The remaining two collider projects are CERN's Future Circular Collider (FCC) and Compact Linear Collider (CLIC). The FCC is a proposed collider with a circumference of roughly 100 km. The FCC project proposes an electron-positron collider as the first stage (FCC-ee) to start operation in the mid 2040s, with a hadron collider as the second stage (FCC-hh) to start operation in the 2070s.

The FCC-ee, would have a collision energy range of 88 GeV to 365 GeV. Much like the CEPC, the FCC-ee would be able to study Z, W, and Higgs bosons at energies of 91 GeV,

161 GeV, and 240 GeV respectively. However, the FCC-ee would also be able to explore the top quark, through top quark pair production around the threshold of 340-365 GeV [4]. In accordance with the ESPP 2020 update guideline of operating a proton-proton collider at the highest achievable energy, the FCC-ee tunnel is designed to host the subsequent FCC-hh, a hadron collider able to reach collision energies of 100 TeV [4].

Financial and feasibility studies of the FCC projects are currently ongoing and is expected to be presented at the next update to the ESPP [19].

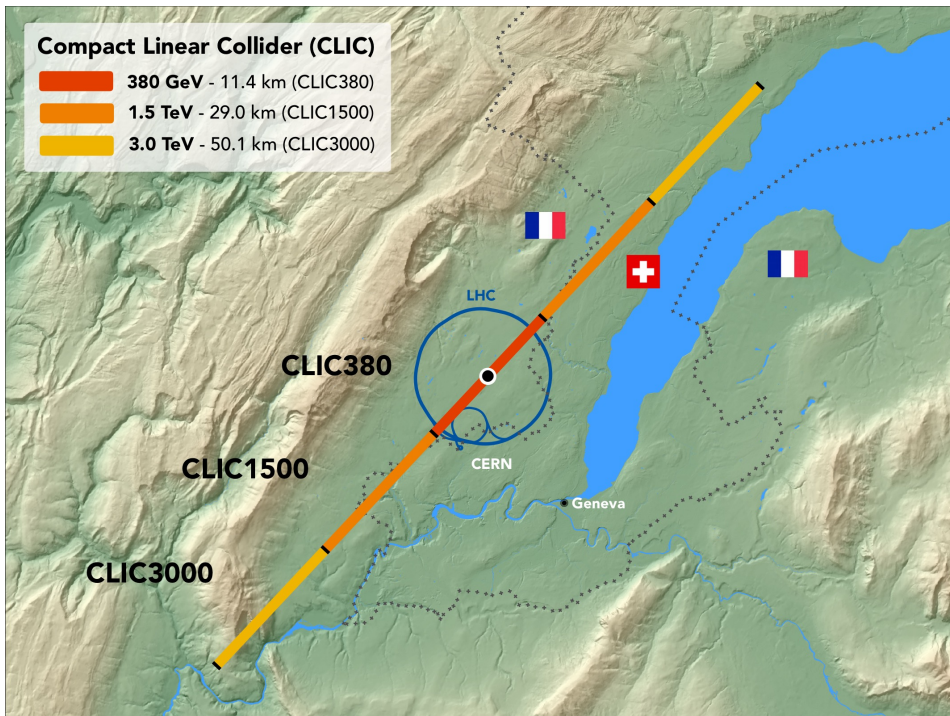


Figure 1.2: Map showing the footprint of the LHC and that of the CLIC with energy stages 380 GeV, 1.5 TeV and 3.0 TeV in the border region between Switzerland and France at the CERN site close to Geneva. Source: [5]

The proposed CLIC is a linear collider that could potentially start construction in 2026 [20]. The main advantage of a linear collider is that it does not exhibit energy loss through synchrotron radiation. However, since the beam has to be accelerated to the target energy over one pass, as opposed to multiple turns, the accelerator has to either be very long or have a significantly large acceleration gradient to achieve the collision energies needed. Hence, the CLIC will utilise a novel two beam system to generate an acceleration gradient of 100 MV per meter in its main linear accelerators (LINAC), which significantly exceeds the acceleration gradient in the LHC of 5 MV per meter [21].

The collider would be built in three stages, with each new stage providing higher collision energy. The first stage could start construction already in 2026, with the first collisions happening in 2035 [22]. The accelerator site would be slightly over 11 km long and would

be able to reach energies of 380 GeV. The second stage would increase the site length to 29 km and reach energies of 1.5 TeV. The third and final stage would push the collision energies even further to 3.0 TeV and have a footprint of over 50 km in length [20]. Figure 1.2 shows the footprint of each stage in comparison to the LHC.

1.2 Problem Description

Although electron-positron colliders are better suited for Higgs exploration, the probability that a collision results in the formation of a Higgs boson is still very small. Only one collision in the LHC out of one billion resulted in the formation of the Higgs boson [23]. Therefore, a key factor in determining the performance of a particle collider is the achievable luminosity. Luminosity is a measure of the interaction probability of two colliding beams and is proportional to the frequency at which the beams are fired at each other (f), the number of particles in the beams (N_1, N_2), and the transverse beam size (σ_x, σ_y) [1]:

$$L \propto \frac{fN_1N_2}{\sigma_x\sigma_y} \quad (1.1)$$

assuming equally sized beams, i.e.: $\sigma_{1x} = \sigma_{2x}$, $\sigma_{1y} = \sigma_{2y}$, $\sigma_{1z} = \sigma_{2z}$, which is often the case for beams travelling around the same machine.

For electron-positron colliders, a limiting phenomenon for achieving high luminosity are ion induced instabilities. The ions are generated from residual gases ionised by the electron beam through electric field ionisation or collisions [2]. If the ion buildup is significant, its electromagnetic field will alter the electromagnetic beam environment which can cause the beam to become unstable. The beam instability could either lead to growth of the beam cross section, which reduces the luminosity, or the total loss of the beam [24].

In circular machines, the ion buildup occurs over multiple turns around the accelerator [25]. However, in some cases the ion buildup can manifest along the length of the electron beam in a phenomenon called fast beam-ion instability (FBII) [26]. The generated ions receive a kick from the first bunch in the bunch train forming the electron beam. The ions drift until they receive a kick from the subsequent bunch, trapping the ions and causing them to oscillate around the beam. In the meantime, new ions are generated by each passing bunch, leading to the ion density being significantly larger towards the tail (end) of the beam as opposed to the head (front) [26]. Therefore, FBII is considered a coupled-bunch head-tail instability and must be taken into consideration for both circular and linear accelerators.

The CLIC collaboration is working towards presenting a readiness report by the next ESPP update around 2025-26 [27]. The accelerator's high intensity beams will have a small transverse bunch size and short bunch spacing, thus it is particularly vulnerable to the onset of FBII [28][29]. Although feedback systems have been documented to mitigate the instability in circular accelerators, this solution is not viable for linear machines [30]. Hence, vacuum specifications have to be established to ensure optimal operation of the accelerator.

Furthermore, due to the high voltages present in the CLIC's main LINAC, the accelerator can experience vacuum breakdown. The breakdowns cause local areas of increased pressure inside the accelerator. It is of great interest to the CLIC collaboration to study the

impact of these local pressure spikes in relation to the vacuum limits imposed by FBII on the machine.

1.3 Project Objectives

The main objective of this master project is to establish vacuum specifications for the CLIC main LINAC with respect to the fast beam-ion instability, using CERN's Python modelling tools. This objective is part of a larger effort towards the development of the CLIC, which is described in section 1.2. The main objective has been broken down into three minor objectives.

1. Simulation of fast beam-ion instability in a linear accelerator. The Python tools were designed for synchrotrons and have not been used for linear accelerators.
2. The implementation of electric field ionisation into the Python tools. Although the feature is present in the C-based modelling tool, CERN is moving away from and it is considered decommissioned. This feature does not exist and has to be integrated into the Python tools.
3. Testing of multi-species feature in the numerical Python tools. Although multi-species cloud simulations have been done in build-up studies, it has not been tested in combination with instability studies.

1.4 Specialisation Project Results and Findings

In the autumn of 2022, a specialisation project was conducted through the course "TTK4551 Engineering Cybernetics, Specialization Project" as a precursor to the master project. The aim of the specialisation project was to become familiar with the modelling tools employed by CERN, and prepare for the completion of the master project.

The modelling tool FASTION was used to study the phenomenon of fast beam-ion instability in the CLIC's main LINAC. It was shown that the instability manifests for heavier gas species like CO₂, Kr, and Xe. Furthermore, it was shown that although lighter species like H₂ and CO do not induce instability on their own, their inclusion with CO₂ yields an increased maximum centroid offset compared to CO₂ on its own [8].

The specialisation report discussed some of the shortcomings of FASTION and CERN's objective of moving away from monolithic tools. In addition, the report presented a path forward which included the objectives presented in section 1.3.

1.5 Thesis Structure

Chapter 1 has presented the background and problem which motivates the thesis objectives listed in section 1.3. Results from the specialisation project were also summarised in section 1.4.

The rest of the thesis is structured to present the applied methodologies and results from studies undertaken in pursuit of the thesis objectives.

The relevant physics is presented in Chapter 2. Linear beam dynamic concepts such as beam size, emittance and transfer matrix is covered in section 2.1. The beam dynamics will first be expressed for a single charged particle travelling in a circular accelerator, before being extended to a particle ensemble. It should be noted that the same approach can be applied to linear machines.

Section 2.2 will present the gas ionisation mechanisms, scattering and electric field ionisation, while section 2.3 introduces the fundamentals of the fast beam ion instability phenomenon. The concept of ion trapping and the instability growth rate will be introduced. Mitigation strategies such as active feedback and beam environment will be discussed briefly.

The fundamental approach of numerical modelling of FBII will be presented in section 2.4. The section also gives an overview of the numerical modelling tools used at CERN for studying FBII.

Chapter 3 will present how the Python modelling tools were applied for studying FBII in the CLIC main LINAC. This includes the linear accelerator setup, the implementation of multiple gas species, and the integration process of ion generation through electric field.

Chapter 4 will present the results from the testing and simulations outlined in chapter 3. The results presented will be discussed in chapter 5. The chapter will also cover the approximations and assumptions made in the modelling process.

Finally, an overview of potential improvements and future work will be given in chapter 6 before the conclusion in chapter ??.

2

Theory

This chapter will outline and review concepts central to understanding the fast beam-ion instability phenomenon, and how the modelling tools are built and treat particle systems. Some of these concepts were originally presented in the specialisation report [8]. For a deeper exploration of the topics in section 2.1, readers are referred to [1] and [6]. Readers are also referred to [31] and [32] for the topics covered in section 2.3 and section 2.4 respectively.

2.1 Concepts from Linear Beam Dynamics

Central to accelerator physics is the equation for the Lorentz force acting upon a charged particle travelling in an electromagnetic field [1]:

$$\vec{F} = e \left(\vec{E} + \vec{v} \times \vec{B} \right) = \dot{\vec{p}} \quad (2.1)$$

The acceleration of particles in an accelerator is done by electric fields oscillating at frequencies of MHz or GHz in metallic chambers called radio-frequency (RF) cavities [33] [1]. After leaving the RF cavity, the particles travel around the accelerator structure. The physical fundamentals of steering and focusing of the particle beam as it travels around the accelerator is called beam optics. Although from (2.1) one can infer that electric fields can be used for this purpose, there are technical limitations in achieving the necessary electric field strength [6]. Consequently the steering and focusing of particles is done by magnetic fields [1]. For large machines like the LHC, these fields are generated by superconducting magnets.

At any point along the circular orbit, the centrifugal force on a charged particle has to be balanced by the Lorentz force.

$$evB = \frac{mv^2}{\rho} \quad (2.2)$$

where m is the mass of the particle and ρ is the radius of the orbit. Inserting the momentum of the particle $p = mv$ yields:

$$\frac{p}{e} = B\rho \quad (2.3)$$

which gives rise to the definition of the magnetic rigidity $B\rho$. The magnetic rigidity can be used to characterise circular accelerators, since its value directly indicates the achievable beam momentum [6].

Through the Taylor expansion of the vertical magnetic field in the vicinity of the particle trajectory, the total magnetic field acting on the charged particle can be seen as the sum of contributions from magnetic multipoles:

$$B_y = B_{y0} + \frac{dB_y}{dx}x + \frac{1}{2!}\frac{d^2B_y}{dx^2}x^2 + \frac{1}{3!}\frac{d^3B_y}{dx^3}x^3 + \dots \quad (2.4)$$

Linear beam dynamics concerns itself with the first orders of magnetic field contributions, namely dipoles and quadrupoles. The different types of multipoles, their contributions, and their uses are listed in Table. 2.1.

TABLE 2.1
MAGNETIC POLE CONTRIBUTION AND FUNCTIONALITY[1]

Multipole	Contribution	Effect
Dipole	B_{y0}	Steering
Quadrupole	$\frac{dB_y}{dx}$	Focus
Sextupole	$\frac{d^2B_y}{dx^2}$	Chromatic compensation
Octupole	$\frac{d^3B_y}{dx^3}$	Field errors or field compensation

2.1.1 Beam Coordinates, Trace Space, and Equations of Motion

Consider the charged particle q travelling parallel to the ideal path of a reference particle around a circular accelerator in Fig. 2.1. The location of q in relation to the reference particle can be described by the Cartesian coordinate system $(\hat{e}_x, \hat{e}_y, \hat{e}_z)$, whose origin follows the orbit of the reference particle along the ideal path.

The arc length parameter s is measured along the ideal orbit starting from a defined position. This position is often the injection point on the accelerator or a symmetry point on the synchrotron [6]. When the Cartesian coordinate system moves along the ideal path of the reference particle it is rotated about the vertical axis, thus the coordinate system is dependent on s [6]. Since the position s is uniquely defined for any time t , the beam dynamics can be expressed in terms of the accelerator position s [1] [8].

The particle q can be described by the coordinates $x(s)$ and $y(s)$ in the co-moving coordinate system as its transverse displacement from the ideal orbit, and $z(s)$ as is its

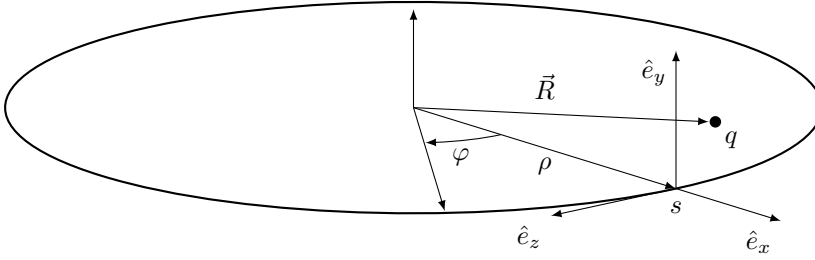


Figure 2.1: The co-moving Cartesian coordinate system for the motion of particle q in a circular accelerator.

longitudinal position with respect to the reference particle [34]. The change in transverse displacement is defined by the angular displacement [6]:

$$x(s)' = \frac{dx(s)}{ds} = \tan(\alpha_x) \approx \alpha_x \quad (2.5a)$$

$$y(s)' = \frac{dy(s)}{ds} = \tan(\alpha_y) \approx \alpha_y \quad (2.5b)$$

The coordinates and angular displacements, together with the relative momentum offset from the momentum of the reference particle, $\delta = \Delta p/p_0$, form the trace space:

$$\left[\begin{pmatrix} x \\ x' \end{pmatrix} \quad \begin{pmatrix} y \\ y' \end{pmatrix} \quad \begin{pmatrix} z \\ \delta \end{pmatrix} \right] \in \Gamma \quad (2.6)$$

which fully characterises the particle q . For a particle with very small deviation in momentum from the reference particle, the trace space can be identified with the phase space (u, p_u) through the relation between transverse angular displacement and transverse momentum [6] [35]:

$$p_u = \beta_r \gamma_r m_0 c \cdot u' \Rightarrow u' \approx \frac{p_u}{p_0}, \quad p_0 \gg p_u, \quad u = x, y \quad (2.7)$$

Treatment of the beam dynamics requires solving the equation of motion. The equations are found by first introducing the position vector which gives the location of q with respect to the centre of the accelerator orbit:

$$\vec{R} = (\rho + x(s)) \cdot \hat{e}_x + y(s) \cdot \hat{e}_y \quad (2.8)$$

The Lorentz force from the magnetic fields causes a change of position vector \vec{R} , which leads to a change in momentum:

$$\dot{\vec{p}} = \gamma_r m_0 \frac{d^2 \vec{R}}{dt^2} = e (\vec{v} \times \vec{B}) \quad (2.9)$$

The following assumptions enable the linear transverse equations of motion to be derived directly from (2.9) [6]:

- A particle is moving with constant longitudinal velocity and $v_z \gg |v_x|, |v_y|$.
- The orbit curvature varies slowly such that $\rho' = \rho'' = 0$.
- The design orbit is in the x, z plane and the vertical unit vector is independent of s , i.e. $\hat{e}'_y = 0$.
- Transverse displacements are small compared to the radius of the orbit, $x, y \ll \rho$.
- There is no coupling between transverse and longitudinal motion and $R_z = R'_z = R''_z = 0$.
- The deviation from the reference momentum is very small $\Delta p \ll p_0$.

Applying these assumptions to (2.9) yields:

$$\gamma_r m_0 \left(v_z \frac{\rho}{r(s)} \right)^2 \bar{R}'' = e v_z (-B_y \hat{e}_x + B_x \hat{e}_y) \quad (2.10)$$

Only considering linear magnetic field contributions from dipole and quadrupole, and neglecting non-linear terms leads to the transverse linear equations:

$$x''(s) + \left(\frac{1}{\rho^2(s)} - k(s) \right) \cdot x(s) = \frac{1}{\rho(s)} \frac{\Delta p}{p} \quad (2.11a)$$

$$y''(s) + k(s) \cdot y(s) = 0 \quad (2.11b)$$

The nature of the transverse motion is oscillatory and is referred to as betatron oscillation. The full derivation of (2.11a) and (2.11b) can be found in [6], [1] and appendix A.

2.1.2 Particle Beam Properties

The particle beam is divided into packets of N_b particles called bunches. Each of the particles have an associated set of coordinates $(\vec{p}, \vec{q})_{N_b}$ [34]:

$$(\vec{p}, \vec{q})_{N_b} \equiv (x_1, x'_1, y_1, y'_1, z_1, \delta_1, \dots, x_{N_b}, x'_{N_b}, y_{N_b}, y'_{N_b}, z_{N_b}, \delta_{N_b}) \quad (2.12)$$

When N_b has a high order such as 10^{14} , it can be easier to describe the bunch in terms of the particle distribution $\psi(\vec{q}, \vec{p})$. In many cases, the distribution can be treated as a Gaussian probability density function, from which the bunch intensity N_b , centroid position $\langle u \rangle$, and size σ_u are defined according to [34]:

$$N_b = \int \psi(\vec{q}, \vec{p}) d\vec{q} d\vec{p} \quad (2.13a)$$

$$\langle u \rangle = \int u \cdot \psi(\vec{q}, \vec{p}) d\vec{q} d\vec{p} \quad (2.13b)$$

$$\sigma_u^2 = \int (u - \langle u \rangle)^2 \cdot \psi(\vec{q}, \vec{p}) d\vec{q} d\vec{p} \quad (2.13c)$$

where $u = x, y$. Thus, the transverse size in the horizontal and vertical plane is the standard deviation of the particle distribution and the centroid position of the particle bunch is the mean of the particle distribution.

Another important property of the bunch is the emittance, which is related to the area occupied by the particles in the trace plane, see Fig. 2.2 [32]. An important characteristic of the emittance is given by Liouville's theorem which states that the emittance is constant for a non-accelerating beam.

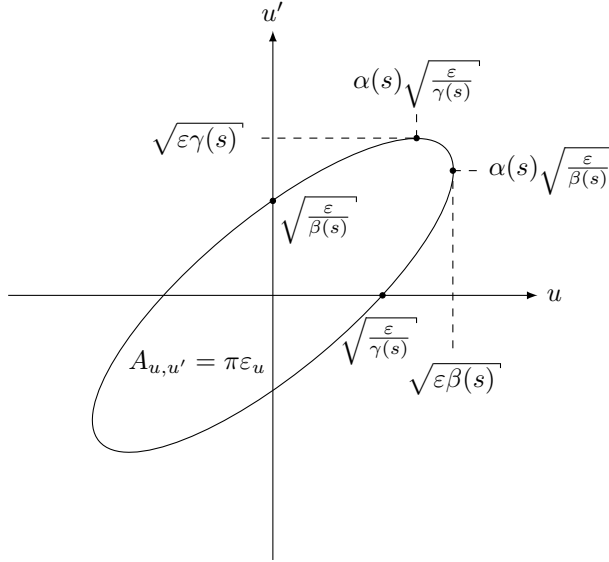


Figure 2.2: The emittance is directly related to the area of the ellipse in the trace plane. Adapted from [1]

The emittance is given by [1]:

$$\epsilon_u = \gamma(s)u^2(s) + 2\alpha(s)u(s)u'(s) + \beta(s)u'^2(s), \quad u = x, y \quad (2.14)$$

where $\alpha(s)$, $\beta(s)$, and $\gamma(s)$ are s -dependent optics functions related to the Courant-Snyder parameters or Twiss parameters α , β , and γ . The beta function $\beta(s)$, also called the amplitude function, describes the maximum local amplitude of the betatron oscillations, while $\alpha(s)$ describes the slope of the beta function [1] [32]. The three functions are related through the Courant-Snyder conditions:

$$\gamma(s)\beta(s) - \alpha(s)^2 = 1, \quad \alpha(s) = \frac{-\beta(s)'}{2} \quad (2.15)$$

The relationship between the beta function and emittance can be seen in Fig. 2.3. The figure also shows how the emittance can change shape on the trace space plane, while remaining constant.

The emittance can further be described in a statistical manner by the transverse bunch size and standard deviation of the angular displacement:

$$\varepsilon_u^2 = \sigma_u^2 \sigma_{u'}^2 - (\sigma_u \sigma_{u'})^2, \quad u = x, y \quad (2.16)$$

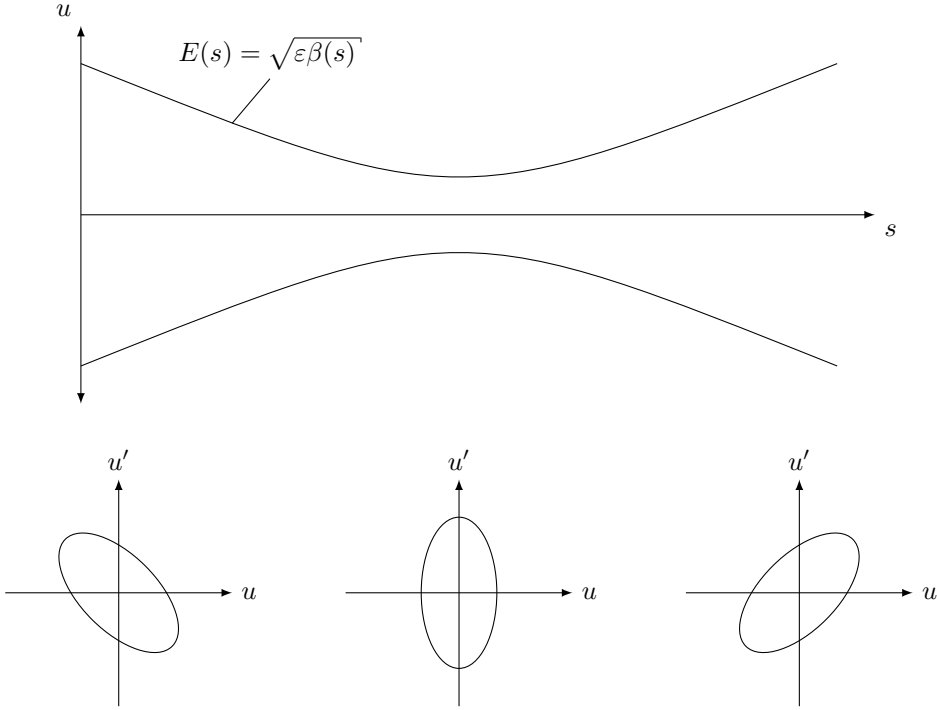


Figure 2.3: The upper figure shows the evolution of the betatron oscillation envelope $E(s)$. All particle trajectories lie within this envelope. The lower graph shows the emittance plot corresponding to the envelope. Although the shape of the emittance plot changes, the value of the emittance remains constant in accordance with Liouville's theorem. Adapted from [1] and [6].

However, accelerating the particles causes a compression of the angular displacement, resulting in a decreasing emittance [6]. Thus Liouville's theorem does not hold for an accelerating beam. Scaling the emittance with the Lorentz factor γ_r and the relativistic speed of the particles β_r gives rise to the normalised emittance, which is constant even for an accelerating beam:

$$\varepsilon_{u,norm} = \beta_r \gamma_r \varepsilon_u \quad (2.17)$$

Combining (2.16) and (2.17) gives the relation between normalised emittance and the transverse bunch size:

$$\varepsilon_{u,norm}^2 = \gamma_r^2 \beta_r^2 \left(\sigma_u^2 \sigma_{u'}^2 - (\sigma_u \sigma_{u'})^2 \right), \quad u = x, y \quad (2.18)$$

2.1.3 Transfer Matrix and Beam Motion

For modelling a synchrotron machine, it is convenient to have the particles return to their initial position after one revolution around the accelerator structure. Mathematically, this is expressed as applying a periodic condition $K(s) = K(s + C)$ to (2.11a), where C is the circumference of the synchrotron. The resultant equation is known as Hill's equation [1], [32]:

$$x(s)'' + K(s)x(s) = 0 \quad (2.19)$$

whose general solution is given by the initial conditions of x and x' [32]:

$$x(s) = ax(s_0) + bx'(s_0) \quad (2.20a)$$

$$x'(s) = cx(s_0) + dx'(s_0) \quad (2.20b)$$

Expressing the general solution on matrix form gives rise to the transfer matrix M :

$$\begin{pmatrix} x \\ x' \end{pmatrix} \Big|_s = \begin{pmatrix} a & b \\ c & d \end{pmatrix} \begin{pmatrix} x \\ x' \end{pmatrix} \Big|_{s_0} = M(s|s_0) \begin{pmatrix} x \\ x' \end{pmatrix} \Big|_{s_0} \quad (2.21)$$

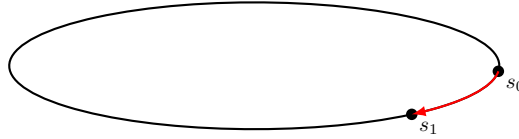


Figure 2.4: Motion from position s_0 to s_1 on the orbit which is described by the transfer matrix

From (2.21) it is possible to see that the transfer matrix can be used to calculate the particle motion from point s_0 to point s_1 in Fig. 2.4. Furthermore, the propagation from s_0 to s_n can be decomposed into a string of matrix multiplications:

$$M(s_n|s_0) = M(s_n|s_{n-1}) \cdots M(s_2|s_1) M(s_1|s_0) \quad (2.22)$$

The solution to (2.19) can be found by applying Floquet's theorem, which yields [1]:

$$x(s) = \sqrt{\frac{\beta_{s_1}}{\beta_{s_0}}} [\cos \Delta\Psi(s) + \alpha_{s_0} \sin \Delta\Psi] x_0 + \sqrt{\beta_{s_1} \beta_{s_0}} \sin \Delta\Psi(s) x'_0 \quad (2.23a)$$

$$\begin{aligned} x'(s) = & \frac{1}{\sqrt{\beta_{s_1} \beta_{s_0}}} [(\alpha_{s_0} - \alpha_{s_1}) \cos \Delta\Psi(s) - (1 + \alpha_{s_0} \alpha_{s_1}) \sin \Delta\Psi] x_0 \\ & + \sqrt{\frac{\beta_{s_0}}{\beta_{s_1}}} [\cos \Delta\Psi(s) - \alpha_{s_1} \sin \Delta\Psi(s)] x'_0 \end{aligned} \quad (2.23b)$$

where $\Delta\Psi$ is the betatron oscillation phase. The transfer matrix in (2.21) can therefore be written as [32]:

$$M(s_1|s_0) = \begin{pmatrix} \sqrt{\frac{\beta_{s_1}}{\beta_{s_0}}} & 0 \\ -\frac{\alpha_{s_1}}{\sqrt{\beta_{s_1}}} & \frac{1}{\sqrt{\beta_{s_1}}} \end{pmatrix} \begin{pmatrix} \cos \Delta\Psi & \sin \Delta\Psi \\ -\sin \Delta\Psi & \cos \Delta\Psi \end{pmatrix} \begin{pmatrix} \frac{1}{\sqrt{\beta_{s_0}}} & 0 \\ \frac{\alpha_{s_0}}{\sqrt{\beta_{s_0}}} & \sqrt{\beta_{s_0}} \end{pmatrix} \quad (2.24)$$

The same process can be done to (2.11b) for the vertical plane. Thus, the propagation of the multi-particle ensemble through an accelerator can be described by the transfer matrix in (2.24). The optics functions for a machine are often known and the transfer matrix is therefore often used in numerical modelling of beam dynamics, which will be discussed in section 2.4.

2.2 Ionisation mechanisms

After vacuum pump-down the pressure level in the beam duct of an accelerator can reach an Ultra-High Vacuum (UHV) of less than 10^{-9} mbar [2]. Although very low, the residual gases left in the beam duct at UHV can interact with the beam and cause unwanted perturbations. One of these interactions is the ionisation of the residual gas through the loss of atomic electrons. The ionisation process is dominated by either scattering or field ionisation. Both mechanisms are covered in this section.

2.2.1 Scattering Ionisation

In some cases a particle will transfer enough energy to overcome the required ionisation energy when colliding with the residual gas, in a process called scattering ionisation. The collision rate for a single particle is given by [2]:

$$\frac{1}{\tau_{col}} = n_{gas} \cdot \sigma^{col} \cdot \beta c \quad (2.25)$$

where n_{gas} is the density of the residual gas species as described by the ideal gas law:

$$n_{gas} = \frac{P_m}{k_B T} \quad (2.26)$$

where k_B is the Boltzmann constant, T is the temperature in Kelvin, and P_m is the associated pressure of the gas species m . The ionisation rate of a single particle is obtained directly from (2.25) by replacing the collision cross section σ^{col} with the ionisation cross section σ^{ion} . The ionisation cross section is given by [2]:

$$\sigma_m^{ion} = 4\pi \left(\frac{\hbar}{2\pi m_e c} \right)^2 \cdot (M^2 \cdot x_1 + C \cdot x_2) \quad (2.27)$$

where the parameters M^2 and C are dependent on the gas species, see Table. 2.2, \hbar is the reduced Planck constant, and

$$x_1 = \beta_r^2 \cdot \ln \left(\frac{\beta_r^2}{1 - \beta_r^2} \right) - 1 \quad (2.28a)$$

$$x_2 = \beta_r^2 \quad (2.28b)$$

Since (2.25) gives the collision rate for one particle, the ionisation rate per bunch is directly dependent on the bunch intensity (N_b) [28].

$$\frac{dN_{ion}}{ds} = \frac{N_b}{k_B T} \cdot \sum_m P_m \sigma_m^{ion} \quad (2.29)$$

The initial distribution of the ions after generation will take the same shape as the bunch that created them, i.e. a Gaussian bunch will generate a Gaussian ion distribution[2].

TABLE 2.2
 M^2 , C , AND σ^{ion} FOR TYPICAL SPECIES WHEN BEAM ENERGY OF 9 GeV [2].

Gas Species	M^2	C	σ^{ion} [MBarn]
H ₂	0.5	8.1	0.3
H ₂ O	3.5	32.3	1.7
CO	3.7	35.1	1.9
CO ₂	5.75	55.9	3.0

2.2.2 Electric Field Ionisation

A bunch of charged particles has an associated electric field which is dependent on the bunch intensity and transverse size. In some cases, the bunch generates a sufficiently large electric field to instantaneously ionise the residual gas in its vicinity [3]. The electric potential generated by the beam at the location (x, y) from the bunch centre is given by the Bassetti-Erskine formula [36]:

$$E_y + iE_x = \frac{N_b e}{2\epsilon_0 \sqrt{\pi} r} \left[w \left(\frac{x + yi}{r} \right) - e^{\left(-\frac{x^2}{2\sigma_x^2} - \frac{y^2}{2\sigma_y^2} \right)} w \left(\frac{\frac{x\sigma_y}{\sigma_x} + \frac{y\sigma_x}{\sigma_y} i}{r} \right) \right] \quad (2.30)$$

where ϵ_0 is the permittivity of vacuum, and $r = \sqrt{2(\sigma_x^2 - \sigma_y^2)}$, and $w(\cdot)$ is the complex error function:

$$w(z) = e^{-z^2} \left[1 + \frac{2i}{\sqrt{\pi}} \int_0^z e^{t^2} dt \right] \quad (2.31)$$

The electric potential in x and y can therefore be expressed as the imaginary and real part of the electric potential in (2.30).

The probability of a gas particle with ionisation potential ξ [eV] to be ionised by an electric field E [GVm⁻¹] is:

$$p(\xi, E) = \frac{1.52 \cdot 10^{15} \times 4^n \xi}{n\Gamma(2n)} \left(\frac{20.5\xi^{3/2}}{E} \right)^{2n-1} \exp \left\{ \left(-\frac{6.83\xi^{3/2}}{E} \right) \right\} \quad (2.32)$$

with $n = 3.69\xi^{-1/2}$ for a single ionisation event [3].

The threshold electric field (E_{th}) for a gas species can be derived from (2.32) and values for common residual gas species are listed in Table. 2.3, assuming a 10% ionisation probability per bunch.

TABLE 2.3
ELECTRIC FIELD THRESHOLD FOR SELECTED GAS SPECIES [3].

Gas Species	Atomic mass number (A)	Electric Field Threshold (E_{th}) [GVm^{-1}]
H ₂	2	26
H ₂ O	18	18.5
CO	28	22
CO ₂	44	21.5

Determining the transverse position from the bunch centre at which the bunch-generated electric field exceeds the electric field threshold leads to the ionisation area defined by $\sigma_{x_{fion}}$ and $\sigma_{y_{fion}}$ in Fig. 2.5. The number of generated ions over distance can now be written as:

$$\frac{dN_{ion}}{ds} = \frac{\pi\sigma_{x_{fion}}\sigma_{y_{fion}}}{10 \times k_B T} \cdot \sum_n P_n \quad (2.33)$$

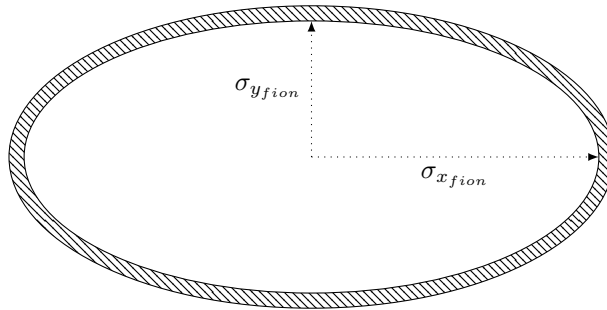


Figure 2.5: The ionisation area swept by the beam is the elliptical area determined by $\sigma_{x_{fion}}$ and $\sigma_{y_{fion}}$. The shaded area contains gas particles that are able to diffuse into the ionisation area during the bunch spacing.

The distribution of the ions generated by (2.33) can be seen in Fig. 2.6 to have the shape of a half circle.

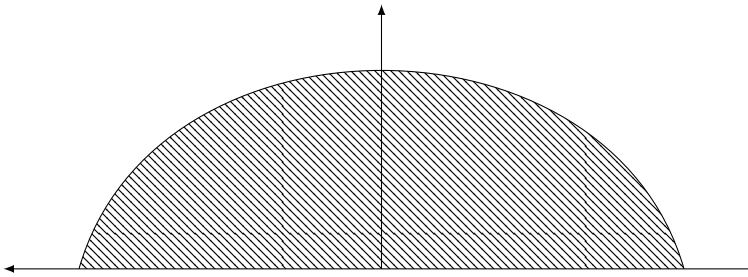


Figure 2.6: The distribution of ions generated by the electric field ionisation of residual gas.

Since only 10% of the residual gas is ionised in the area swept by the bunch, the area is fully ionised after 10 bunches have passed through. Thus the only gas particles to be ionised by subsequent bunches, are particles that have diffused into the area during the bunch spacing (ΔT) [3]. The distance travelled by a gas particle is determined by its average thermal velocity (v_{th}):

$$l_g = v_{th} \Delta T_b = \sqrt{\frac{8k_B T}{m}} \Delta T_b = \sqrt{\frac{8k_B T}{A \cdot p_m}} \Delta T_b \quad (2.34)$$

where A is the atomic mass number of the gas species, and p_m is the proton mass in kg. Hence the gas particles that can diffuse into the ionisation area exist within a small area around its perimeter, see Fig. 2.5. Assuming that the distance the gas particle can travel is significantly smaller than the transverse size of the ionisation area, $l_g \ll \sigma_{y_{fion}}$, this area is given by:

$$\begin{aligned} A_{ion} &= \pi \left((\sigma_{y_{fion}} + l_g)(\sigma_{x_{fion}} + l_g) - \sigma_{y_{fion}} \sigma_{x_{fion}} \right) \\ &= \pi \left(l_g (\sigma_{y_{fion}} + \sigma_{x_{fion}}) + l_g^2 \right) \\ &\approx \pi l_g (\sigma_{y_{fion}} + \sigma_{x_{fion}}) \end{aligned} \quad (2.35)$$

It is assumed that only a quarter of the ions are travelling in the correct direction to enter the ionisation area. Therefore the number of generated ions over unit length is given by [28]:

$$\frac{dN_{ion}}{ds} = \frac{1}{4k_B T} (\sigma_{x_{fion}} + \sigma_{y_{fion}}) \Delta T_b \cdot \sum_n P_n(v_{th})_n \quad (2.36)$$

The distribution of the ionised gas particles that diffused into the ionisation area can be seen in Fig. 2.7 to have a bowl shape.

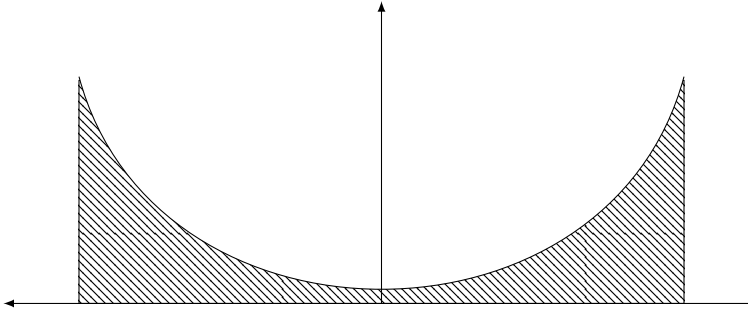


Figure 2.7: The shaded area contains ions which are able to reach the ionisation area

2.3 Fast Beam-Ion Instability

Unlike classical ion instabilities that manifest over multiple turns in a synchrotron, the fast beam-ion instability (FBII) manifests along a single passage of the bunch train in the

accelerator [8]. FBII can therefore affect linear machines, which are otherwise not affected by classical ion instability, as well as circular machines [25]. FBII was first observed in 1997 in the Advanced Light Source (ALS), and has since then been detected during commissioning of accelerators, in machines where impedance heating induces a pressure increase, and in dedicated experiments where a gas species has been injected into the beam duct [25], [37], [2].

When the first bunch interacts with the residual gas in the beam duct, the gas is ionised through one of the mechanisms described in section 2.2. The newly generated ions will receive an electromagnetic kick from the beam's electric field. The kick can be calculated through the linear approximation of the Bassetti-Erskine equation (2.30) [2]:

$$k_{x,y} = \frac{2N_b r_p c}{A \sigma_{x,y} (\sigma_x + \sigma_y)} \quad (2.37)$$

Applying the stability condition for a linear beam trajectory

$$k_{x,y} \Delta T_b < 4 \quad (2.38)$$

leads directly to the trapping condition and definition of critical mass number A_{trap} [25]:

$$A > A_{trap} = \frac{N_b r_p \Delta T_b c}{2 \sigma_{x,y} (\sigma_x + \sigma_y)} \quad (2.39)$$

The trapping condition states that for an ion to be trapped by the passing bunch, its atomic mass number must not be less than the critical mass number [38]. In other words, for lighter gas species the kick received from a bunch will over-focus the ion and it will be lost before the next bunch arrives [2]. The trapping condition has to be satisfied for both transverse planes, and for electron beams which typically have flat shaped bunches, $\sigma_y < \sigma_x$, the critical mass number is determined by the vertical condition [38] [24]. However, the trapping condition in (2.39) is an underestimate since it overestimates the magnitude of the electromagnetic kick on ions outside the bunch core [25]. Simulations of FBII in FCC-ee have shown that even ion species that do not satisfy (2.39) can be effectively trapped and induce instability, although the instability behaves distinctly different [38].

The ions that are trapped by the beam develop a transverse oscillatory motion around the bunch centroid. Due to the ions having a comparatively larger mass than the electrons, the ions oscillate with an oscillation period longer than the bunch spacing [2]. Thus, the ions transfer information from the preceding bunch to the subsequent bunch [25]. As the ions accumulate, their electric field can start to influence the electrons in the beam. If the ion buildup along the bunch train is significant, the ion oscillation and the beam betatron oscillation interact resonantly to cause mutually driven coherent transverse motion, resulting in the development of the coupled-bunch fast beam-ion instability.

Since the ion accumulation is linear along the bunch train, the instability sets in at the tail end [38]. The time-constant for the growth rate of FBII at the tail of a bunch train of length n is given by [38] [2]:

$$\tau_{x,y}^2 \propto \frac{\gamma^2 A \omega_\beta}{n_b^4 N_b^3 P^2 \Delta T_b c} (\sigma_x + \sigma_y)^3 \sigma_{x,y}^3 \quad (2.40)$$

where ω_β is the betatron oscillation frequency. Although (2.40) is derived from the linear approximation of (2.30), it indicates that the growth rate is heavily dependent on n , N_b , P , and $\sigma_{x,y}$. Furthermore, it shows that the instability manifests quicker for lighter gas species.

FBII can cause beam degradation such as emittance growth, fast growing coherent vertical centroid motion, and in the worst case beam loss [26]. An increase in emittance will reduce the growth rate of the instability as it increases the time-constant in (2.40). Nevertheless, an increase in emittance would lead to a reduction in achievable luminosity for the accelerator.

A similar instability can occur for positively charged particle beams called electron cloud instability, in which the charged beam generates electrons through residual gas ionisation and secondary emission from the beam duct wall [26].

2.3.1 Mitigation Strategies

Seeing that FBII and other collective effects can have a detrimental impact on the beam quality, it is of great importance to prevent and counteract the instability [8]. When discussing mitigation strategy there are two main approaches, beam parameters and environment, and active feedback control.

Beam parameters and environment refers to the parameters present in (2.40) and (2.39). An increase in the time-constant for the instability growth rate would slow down the instability manifestation, and increasing the critical mass number would reduce the trapping of lighter ion species. However, increasing the critical mass number does not remove possible instability due to weakly trapped ions [38]. Furthermore, reducing the beam trapping potential through increased vertical emittance is not a viable mitigation strategy if the pressure of the gas is high [30].

A key challenge when considering mitigation through beam parameters is the effect those parameters will have on the machine luminosity. For instance, altering the bunch intensity and transverse size to reduce the instability growth rate, would also reduce the luminosity according to (1.1). It has been found that dividing a longer bunch train into a set of smaller trains has a mitigating effect on FBII [30]. By providing a sufficiently long gap between the shorter trains for the ions to be lost to the chamber wall, the strategy takes advantage of the time-constant's second order dependence on the train length. However, doing so reduces the number of spaces in the accelerator occupied by a bunch, indirectly reducing the achievable luminosity of the machine. Both train length and the bunch spacing contribute to the collision frequency in (1.1), thus determining the luminosity.

The final parameter in (2.40) is the gas pressure, which is the only parameter that does not directly or indirectly determine the luminosity. A decrease in gas pressure will cause the instability to manifest slower. At a given pressure, the instability growth time will be longer than the beam's travel time. Thus, determining the pressure threshold for a machine's given beam parameters and luminosity target is an important part of investigating the performance of an accelerator.

Fig. 2.8 shows the basics of a bunch-by-bunch feedback system consisting of a beam position monitor (BPM), a processing unit, and an electromagnetic actuator called a "kicker". The position measurement of each bunch is processed in their own channel and the appropriate electromagnetic signal is applied to the corresponding bunch [7]. A bunch-by-bunch

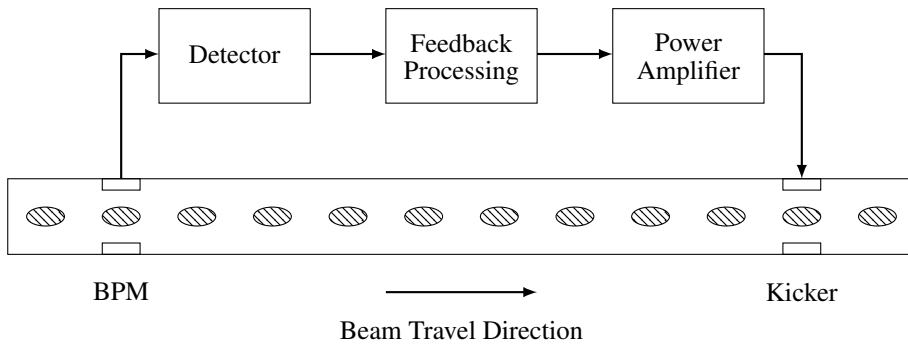


Figure 2.8: Diagram of a bunch-by-bunch feedback system. Adapted from [7]

feedback system is able to stabilise each bunch separately on the condition that the instability growth rate does not exceed the feedback damping time. This condition was observed in SOLEIL, where the feedback system was unable to stabilise the beam due to the rapid growth of FBII induced by localised outgassing of the vacuum chamber walls [2].

To ensure that a bunch receives the right correction in circular accelerators, the signal passes through a delay line. Thus the bunch is kicked one or more turns after the position measurement [7]. Since the beam only passes through a linear accelerator once, this approach is not an option. Therefore, a key characteristic of a feedback system for a linear accelerator is the feedback latency.

A digital bunch-by-bunch feedback system for a linear collider with a latency of 230ns has been demonstrated experimentally [39]. It is expected that future digital feedback systems will be able to take advantage of advances in computing hardware and architectures [40]. Nevertheless, the beam parameters proposed for future linear colliders push the bandwidth of currently available technology. As an example, a bunch-by-bunch feedback control of the electron beams in CLIC, which has a bunch spacing of 0.5ns, is not feasible. A proposed conceptual design for an intra-train feedback system has a latency of 40ns, which is still one order of magnitude smaller than what has been achieved experimentally [40].

Avenues for achieving latencies below 100ns are being explored using all-analogue components. The main disadvantage with these systems is that they are more sensitive to external noise than their digital counterparts [40].

2.4 Overview of Numerical Modelling Tools

Numerical modelling tools are widely used for intensity effects and accelerator limitations studies in accelerator physics and beam dynamics. Numerical modelling is often less complicated than experimental setups and usually use fewer approximations than analytical solutions. They are able to capture the majority of physical effects and allow for detailed study of the underlying mechanisms of intensity and accelerator limitations [32]. The main disadvantage is that they often take a long time to simulate on a computer.

There are different types of numerical modelling tools such as time- or frequency-domain Vlasov solvers, beam envelope tracking algorithms or circulant matrix models [32]. The modelling tools used for studying FBII at CERN are numerical macroparticle models and will be covered in section 2.4.1 and section 2.4.2. The main differences between the tool sets are the programming language and available features [8].

A physical particle system can have of the order of 10^{14} particles, which is not convenient for computer simulations due to memory limitations. To overcome this the particles can be grouped into discrete sets of macroparticles, reducing the particle system to 10^7 macroparticles. All particles in one macroparticle are assumed to interact with the environment in exactly the same manner. Therefore it can be necessary to conduct convergence studies to ensure that any errors are the results of systematically wrong modelling and not from numerical noise [32].

Numerical macroparticle models can further be divided into weak-strong and strong-strong types, which describe how the macroparticles are set up. In weak-strong modelling of FBII only the ions are grouped into macroparticles, while in strong-strong models both ions and beam electrons are treated as macroparticle systems [25]. The macroparticle modelling tools HEADTAIL and ELOUD were among the first to simulate two stream effects in a strong-strong manner [32].

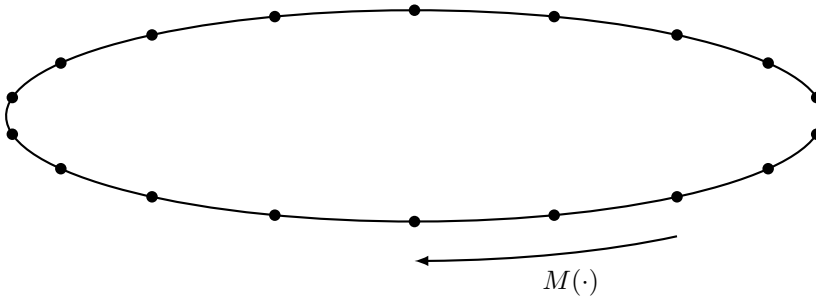


Figure 2.9: The accelerator is divided into interaction points along the accelerator where the beam and environment is modelled. The beam dynamics from one IP to the next is done with the transfer matrix ($M(\cdot)$) defined by the accelerator's optics parameters.

For the numerical treatment of FBII in a circular accelerator, the accelerator is described as a set of interaction points distributed along its structure as seen in Fig. 2.9. In strong-strong macroparticle simulations, the electrons and ions are grouped into separate sets of macroparticles.

At each IP the electromagnetic interaction between the bunch and ion macroparticles is modelled in 2D following the process steps seen in Fig. 2.10. First the bunch is propagated to the IP using the linear transverse transfer matrix in (2.24) with the associated optics parameters for the IP. Then the ions are generated using the mechanisms described in section 2.2 and grouped into macroparticles with assigned trace space coordinates.

Calculating the electromagnetic kick experienced by the electron and ion macroparticles is not a trivial task. However, there are a few assumptions that can be made to reduce the complexity of the problem [32].

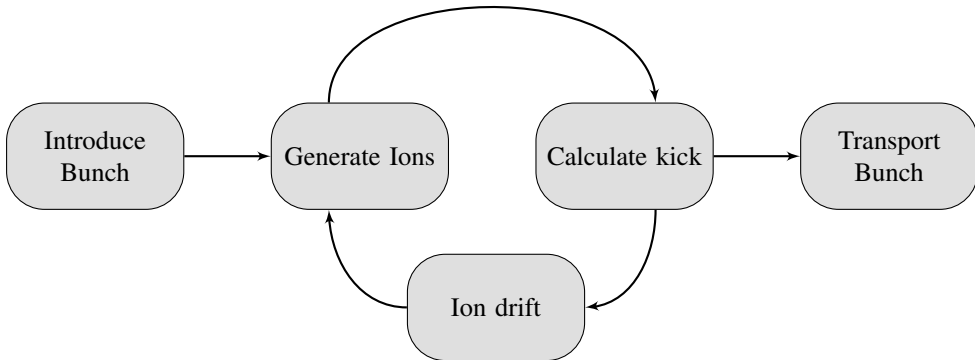


Figure 2.10: The flowchart shows the steps done at each IP. Adapted from []

From section 2.3 it is known that the timescale of the ion transverse motion is larger or comparable to the bunch spacing. Since the bunch length is significantly shorter than the bunch spacing, the bunches can be treated as transversely flat disks in relation to the ions [29]. Furthermore, it is reasonable to assume that the longitudinal dimension of the ion distribution is significantly larger than its transverse distribution. Hence, the electric field from the ions can be considered to be purely transverse. Since the ions are travelling at non-relativistic speed, the magnetic fields from the ions can be neglected [32].

Lastly, the bunches are travelling at relativistic speed which means that the electric fields perpendicular to the direction of travel are strongly enhanced. Thus the electric field from the electron beam can be treated as purely transverse as well [32]. With these assumptions the electromagnetic interaction between electron beam and ion distribution can be treated purely as a transverse electrostatic problem.

Numerical modelling tools often utilise a particle-in-cell (PIC) algorithm to solve the electrostatic problem. The basic PIC algorithm can be broken down into a few main steps [32]:

- Discretization of the region of interest into a mesh of regular or irregular shaped mesh cells.
- Computation of the charge density on the mesh through interpolation of the macroparticle position to the mesh nodes.
- Computation of the electric field by solving the Poisson equation on the mesh and taking the gradient of the resulting electric potential.
- Gather the force fields from the mesh node locations back to the macroparticle positions by interpolations. The same interpolation method as in step 2 has to be used to avoid violating the conservation of momentum.

Once the electromagnetic kick has been applied to all macroparticles, the bunch is propagated to the next IP is calculated with the associated transfer matrix, the ion macroparticle coordinates are updated to account for their drift until the next bunch is introduced to

the IP. At the next IP, the whole process is started again with ion generation, etc. until the beam propagation through the entire accelerator has been calculated [29].

2.4.1 FASTION

The FASTION simulation tool was developed specifically for studying FBII in linear machines and has previously been applied to study the effects of vacuum and gas composition in the CLIC's main LINAC and transfer lines [26], [3]. The majority of the code is written in C, with some calculations handled by code written in FORTRAN 77.

The modelling method of FBII in FASTION follows the same process outlined in section 2.4 and flowchart in Fig. 2.10. The electromagnetic interaction between the bunches and ions are treated in 2D at each IP in a strong-strong manner, and the acceleration of the beam is accounted for by assessing the relativistic gamma associated with each IP [29].

To determine which of the ionisation mechanisms in section 2.2 is dominant for the bunch, FASTION performs a scan of the bunch-generated electric field in the transverse plane. The number of macroparticle ions is determined by the number of ions generated over the distance between adjacent IPs.

To achieve good resolution of the electron beam and generated ions, FASTION employs a dual-grid setup with rectangularly (non-square) shaped grid cells. The dual-grid consists of a fine inner grid of constant size, and a dynamic, coarse outer grid that expands to accommodate the maximum ion trajectory [29].

The kick is calculated by assessing the electric fields from the bunch and the ions separately on the same grid. The velocity kicks are applied according to the electric field of the oppositely charged particles.

FASTION receives inputs from two different files. The first file is a configuration file with user defined inputs that determine the parameters of the residual gas and other simulation parameters. This includes the residual gas composition, the ionisation cross-section of the gas species, bunch spacing, and other beam parameters. The file has a specific structure in order for FASTION to interpret the inputs, the first few lines of a configuration file can be seen in Fig. 2.11.

```
Switch_for_fast_ion:          1
Number_of_ion_species:       3
Partial_pressures_[nTorr]:   20. 20. 40.
Atomic_masses:               2. 18. 44.
Ionization_cross_sections_[MBarn]: 2. 2. 2.
Number_of_electrons_per_bunch: 4.e+9
Bunch_spacing_[ns]:         0.5
Normalized_horizontal_emittance_(rms_value)_[nm]: 660.
Normalized_vertical_emittance_(rms_value)_[nm]: 10.
Bunch_length_(rms_value)_[ns]: 0.15e-3
...
```

Figure 2.11: First few lines from a FASTION configuration file for a multispecies simulation with H₂, CO, CO₂

The second file is an DAT file called TWISS which contains input parameters regarding the machine optics. Table 2.4 shows the structure and content of the TWISS file with example values. The first two columns provide information regarding the position along the accelerator and the beam energy, the remaining six columns provide optics parameters. The structure of the TWISS file is compatible with other tools used by CERN such as MAD-X [26].

TABLE 2.4
EXAMPLE OF TWISS FILE FOR THE CLIC MAIN LINAC

s	GeV	β_x	β_y	α_x	α_y	$accQ_x$	$accQ_y$
0.000	9.00	2.0211100	6.6820400	-0.0000000	0.0000000	0.0000000	0.0000000
2.185	9.13	6.3535288	2.1403304	1.8435426	-0.6941201	0.6051175	0.6634577
4.195	9.26	2.1404718	6.3523817	-0.6941759	1.8429584	1.2419306	1.1837194
6.205	9.39	6.3527274	2.1408572	1.8429043	-0.6944869	1.7621609	1.8205296
...

Although the FASTION code has been used for simulating a storage ring [30], each expansion of the code adds complexity and more conditions. Applying PyELOUD and PyHEADTAIL to the study of FBII and implementing FASTION features has allowed for a shared workload of maintenance and development, and enabled access to features present in PyCOMPLETE [29].

2.4.2 Python Tools

The complexity and variety in simulation scenarios has caused CERN to move away from using monolithic codes with rigid user interfaces, in favour of flexible toolboxes in the form of Python libraries [41], [8]. CERN's BE-ABP group has developed and uses the open source toolbox called "Python Collective Macro-Particle Simulation Libraries with Extendable Tracking Elements" (PyCOMPLETE) for studying coherent effects in particle accelerators []. The toolbox consists of different libraries mostly written in the Python language. For computationally heavy tasks, the libraries are able to be interface with dedicated C and FORTRAN programs through cython and f2py [41]. The advantage of adopting this approach is that arbitrarily complex simulations can be set up by the user in a Python script, and a host of Python packages are freely available for post-processing the simulation data [41]. The Python libraries of interest for the numerical modelling of FBII are called PyELOUD, PyPIC, and PyHEADTAIL.

The PyELOUD library was built on the legacy of ELOUD as a 2D macroparticle solver for the study of electron cloud formation in accelerator structures [41]. It has since be adapted to treat general particle clouds of different mass and charge. The cloud generation can be modelled as either residual gas ionisation though scattering as described in section 2.2.1, photoemission from the chamber wall due to synchrotron radiation, secondary electron emission from electron-wall collisions, or as a combination of the three [41].

The interaction between the bunch and cloud macroparticles is modelled in PyE-CLOUD with one of the many available PIC solvers from the PyPIC library. All the PIC solvers are structured to use the same interface such that they can be used as plug-in modules for different PyCOMPLETE applications. PyPIC also contains a multigrid feature which enables the creation of nested grids, which has been found to have a great impact on processing time [41]. Nested grids are particularly useful in cloud simulations, since the area of interest for the cloud is usually much larger than the area of interest around the bunch [].

The PyHEADTAIL library is a macroparticle tracking code developed for modelling collective beam dynamics in synchrotrons [42]. The machine can be modelled as a list of Python objects based on optics parameters provided by an external file, much like the TWISS file for FASTION [41]. The macroparticles are tracked in six-dimensional trace space and advance from one IP to the next by the transfer matrix M in (2.21) [42]. Dispersion effects are introduced ad-hoc by adding a contribution to the coordinates after the transfer matrix has been applied, and chromaticity is accounted for by an effective detuning model [42].

Returning to the flow chart in Fig. 2.10, it can be seen that the PyE-CLOUD and PyHEADTAIL libraries combined have the necessary features to model FBII. The beam-ion interactions are successfully modelled by PyE-CLOUD, while PyHEADTAIL models the beam dynamics through the accelerator machine. The combination of PyE-CLOUD and PyHEADTAIL was achieved due to the versatility and modularity of the two libraries by the creation of the PyEC4PyHT interface in PyE-CLOUD. PyEC4PyHT facilitates the creation of cloud objects that can be inserted between the machine objects generated by PyHEADTAIL as seen in Fig. 2.12 [41].

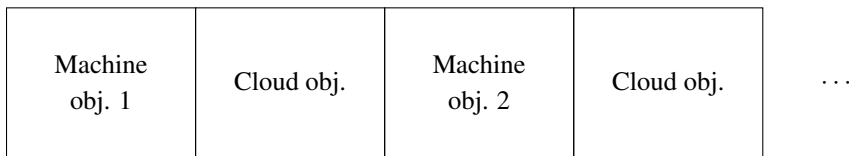


Figure 2.12: The machine is modelled as a list of alternating machine and cloud objects created by PyHEADTAIL and PyE-CLOUD respectively.

3

Application of Python Tools for FBII Studies

The first section will describe how the Python tools described in section 2.4.2 were applied and adapted to simulating FBII for a linear accelerator. The following section will go over how electric field ionisation was implemented as a feature in PyECLLOUD. The final section will present how different simulation studies for FBII in the CLIC's main LINAC and damping ring (DR) were set up, and the parameters used for these studies.

3.1 Simulation Setup for a Linear Accelerator

As mentioned in section 2.4.2, PyHEADTAIL was developed for instability studies in synchrotron machines. A consequence of this is that a python class has been developed for simulating synchrotron machines, however no python class for a LINAC has been implemented. Thus the synchrotron python class has been adapted to simulate FBII in LINACS. To this end, the example script for FBII simulation in the CLIC's DR found on the PyHEADTAIL GitHub was used as a basis [43].

Simulations of electron cloud instability in PyECLLOUD and PyHEADTAIL operate on a timescale within a single bunch and the bunch has been divided into slices. The effects of the electron cloud is calculated for each bunch slice. For simulation of FBII, the bunch slicing method is applied to slice the entire bunch train by setting the simulation time step Δt_{ref} equal to the bunch spacing. Consequently, each bunch is therefore treated as a flat disk just like in FASTION.

The synchrotron map created for PyHEADTAIL contains three parts, a set of transverse map elements corresponding to machine segments, an RF element for applying acceleration, and a longitudinal map element for longitudinal motion. The last two parts are not of interest and are therefore removed after the full map has been created during initialisation in PyHEADTAIL.

Without the RF element, the acceleration of the beam will not be modelled. This is

rectified by the creation of an energy update object to be inserted between the machine object and the cloud object as seen in Fig. 3.1. The energy update object collects the energy associated with the location along the machine for a list of energy values. The new energy is given by the extension of Einstein's mass-equivalence equation called the dispersion relation [1]:

$$E^2 = (pc)^2 + (m_0^2 c^2)^2 \quad (3.1)$$

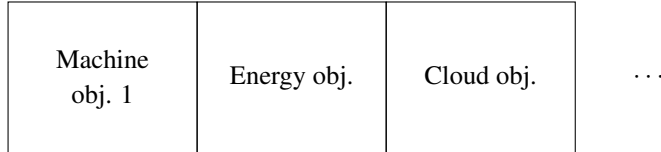


Figure 3.1: The machine modelled is changed to also contain a energy update object between the machine and cloud objects created by PyHEADTAIL and PyECLOUD.

The transverse size of the bunches are calculated in PyECLOUD-PyHEADTAIL according to the definition provided by (2.13c), namely as the standard deviation of the bunch mean. Since the coordinates are independent of the beam momentum, the transverse coordinates and their associated momentum have to be updated to match the momentum change:

$$x_{i+1} = x_i \times \sqrt{\frac{\beta_i \gamma_i}{\beta_{i+1} \gamma_{i+1}}} \quad (3.2a)$$

$$x'_{i+1} = x'_i \times \sqrt{\frac{\beta_i \gamma_i}{\beta_{i+1} \gamma_{i+1}}} \quad (3.2b)$$

3.1.1 Dual Grid Setup

Section 2.4.2 discussed the usefulness of nested grids when simulating cloud instabilities due to the size difference between the area of interest around the beam and the area where the ions move. The implementation of the multi-grid is described in a presentation by Eleonora Belli [] The multiple grid feature in PyPIC enables the implementation of an arbitrary number of nested grid in the simulation through its telescopic grid class.

The size of the refinement mesh of the main PIC grid is defined by:

$$S_0 = N_{min} \Delta h_0 \quad (3.3)$$

where Δh_0 is the mesh point spacing of the main grid as seen in Fig. 3.2 and N_{min} is the minimum number of points of the first internal grid. The size of the first internal grid is given by:

$$S_1 = f \cdot S_0 \quad (3.4)$$

where ($0 < f < 1$) is the magnification factor between the grids. This equation can be generalised to determine the size of the i -th internal grid:

$$S_i = f \cdot S_{i-1} \quad (3.5)$$

and the size of the final n -th grid is then given by:

$$S_n = f^n \cdot S_0 \quad (3.6)$$

$$S_n = f^n \cdot N_{min} \Delta h_0 = S_{target} \quad (3.7)$$

The number of internal grids are defined by:

$$n = \left\lceil \frac{\ln \left(\frac{S_{target}}{N_{min} \cdot \Delta h_0} \right)}{\ln(f)} \right\rceil \quad (3.8)$$

To implement the multiple grid in the simulation script requires the main PIC grid and the target grid has to be defined in the simulation script. The main PIC grid is defined by the Δh_0 and the aperture of the beam chamber through x_{aper} and y_{aper} . The parameters of the target grid S_{target} is x_{min} , x_{max} , y_{min} , y_{max} , and Δh_{target} . In the simulation script, Δh_0 and Δh_{target} has been given the parameter name `Dh_ext` and `Dh_target`.

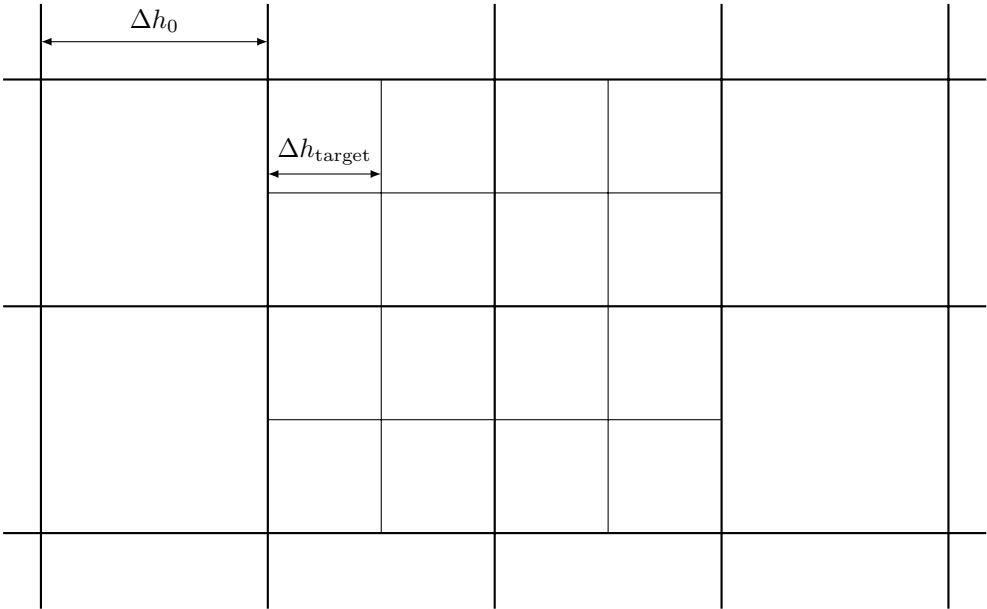


Figure 3.2: Nested grid setup.

3.1.2 Multiple Gas Species Setup

The inclusion of multiple gas species has been used in electron cloud buildup simulations in PyECLOUD. The setup of multiple gas species cloud objects followed instructions from [44] and used the buildup example found on GitHub as guidance.

A cloud file had to be created for each of the gas species present in the simulation. The cloud file had to contain information about the mass and charge of the ions, as well as values declaring the presence of photo-emission and gas ionisation. All the specifications for the cloud files can be found in [44]. The cloud files also contained information about the initial energy of the ions, the pressure of the residual gas, the ionisation cross section, etc. These parameters are strictly optional, but are required for accurate modelling of FBII. The names of the cloud files were listed in the `simulation_parameters.input` file under the `additional_clouds_file_list` parameter. Examples of a cloud file for CO₂ and an example `simulation_parameters.input` file can be found in Appendix C and B respectively.

In addition to the creation of the cloud files, an output location for saving cloud data for each slice interaction had to be provided in the simulation script. With the cloud output location no longer declared as `None`, the script would conduct an initialisation of a python class dedicated to saving the aforementioned cloud data. During this initialisation a Δt for the saver is defined by:

$$\Delta t_{\text{saver}} = (V - 0.0001) \times \Delta t_{\text{ref}} \quad (3.9)$$

where V is some variable in the python class. Because Δt_{ref} defined in the simulation script has not been collected at this stage of the initialisation, and the saver class uses the value from the `simulation_parameters.input` file. Thus, Δt_{ref} can not be set to `None` in `simulation_parameters.input`, like done for simulations with a single gas species. Besides these changes, a few lines of code in the PyEC4PyHT interface had to be commented out since multiple cloud simulations have not been tested with the interface. These lines would cancel the simulation if there were multiple cloud files present, see Fig. 3.3.

```

339
340         if self.cloudsim.flag_multiple_clouds:
341             raise ValueError("Multiple clouds not yet tested in PyEC4PyHT!")
342

```

Figure 3.3: Extract from `PyEC4PyHT.py` showing how the interface would cancel the simulation in the presence of multiple cloud files.

With all of this in place, the machine map would contain a cloud object for each gas species after initialisation, and the machine map seen in Fig. 3.1 would be changed to the structure seen in Fig. 3.4

3.2 Implementing E-Field Ionisation

Residual gas ionisation is handled in the PyECLOUD library by the `gas_ionisation` python class. This class includes an initialisation function and a generation function. The

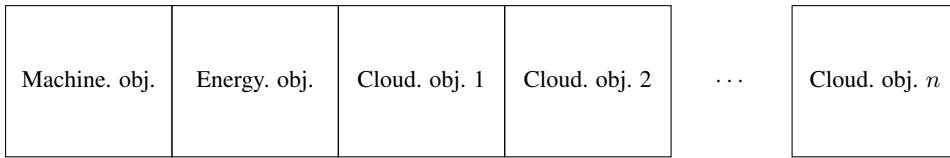


Figure 3.4: The machine object list contains a cloud object for each of the gas species included in the simulation study.

initialisation is important for declaring variables that need to be passed from bunch to bunch. The generation function computes the number of ions generated by the beam according to the collision mechanism described in section 2.2.1. The ions are assigned in macroparticles, which are then given coordinates to form a Gaussian distribution using:

$$x_{ion} = x + X\sigma_x \quad (3.10a)$$

$$px_{ion_k} = v_0 (X - 0.5) \quad (3.10b)$$

where X is a normally distributed random number and

$$v_0 = -\sqrt{2 \frac{E_{int_{ion}}}{3} \frac{e}{A}} \quad (3.11)$$

is the base velocity of the ions, which is based on the initial energy of the ions $E_{int_{ion}}$.

Figure 3.5 shows how `gas_ionisation` was modified to incorporate the electric field ionisation mechanism described in section 2.2.2. To ensure backwards compatibility with other instability studies and functions in the PyCOMPLETE toolbox, the initialisation function in `gas_ionisation` checks whether values have been assigned for the electric field threshold and electric field ionisation probability. If these parameters are not declared when initialising the gas clouds, the class will consider the feature disabled and calculate ion generation according to the collision ionisation mechanism. In other words, it will function like it did before electric field ionisation was implemented.

When these parameters have been declared, the ion generation function follows the same steps as found in FASTION. It is important to note that the two modelling tools treat the ion generation slightly differently. The FASTION modelling tool evaluates dN_{ion} during its ion generation function, while PyELOUD evaluates $\frac{dN_{ion}}{ds}$. If the bunch passed to `gas_ionisation` is the first in the bunch train, a search is done along the x and y axis to determine if the bunch-generated electric field exceeds the electric field threshold. A region of $\pm 10\sigma_x$ along the y -axis is scanned and at each iteration step of the search, the generated field in that location is calculated according to (2.30) and normalised with respect to the electric field threshold:

$$E_{norm} = \frac{|E|}{0.3\sqrt{2\pi}\sigma_z E_{th}} \quad (3.12)$$

The variable position along the axis where the normalised electric field exceeds 1 is assigned to $\sigma_{y_{fion}}$ from Fig. 4.1. Then a similar search is done along the x -axis at the y -position with the maximum electric field. When the normalised electric field again exceeds

1, the position along the x -axis is assigned to $\sigma_{x_{\text{fion}}}$. If $\sigma_{x_{\text{fion}}} > 10^{-7}$, then the electric field ionisation generation is activated. In the event that this condition does not hold, the ion generation is done according to the collision mechanism.

To achieve the ion distribution seen in Fig. 2.6, the coordinates of the ions are assigned according to:

$$x_{\text{ion}} = \langle x_{\text{bunch}} \rangle + X \sigma_{x_{\text{fion}}} \quad (3.13a)$$

$$y_{\text{ion}} = \langle y_{\text{bunch}} \rangle + X \sigma_{y_{\text{fion}}} \quad (3.13b)$$

where X is a uniformly distributed random number. For the ions generated after the area has been fully ionised, $n_b > E_{\text{prob}}$, the ion coordinates are assigned according to:

$$x_{\text{ion}} = \langle x_{\text{bunch}} \rangle + \sqrt{(\sigma_{x_{\text{fion}}} - 10^{-7})^2 + X (2\sigma_{x_{\text{fion}}} - 10^{-7}) \times 10^{-7}} \cos(X) \quad (3.14a)$$

$$y_{\text{ion}} = \langle y_{\text{bunch}} \rangle + \sqrt{(\sigma_{y_{\text{fion}}} - 10^{-7})^2 + X (2\sigma_{y_{\text{fion}}} - 10^{-7}) \times 10^{-7}} \sin(X) \quad (3.14b)$$

which results in the distribution seen in Fig. 2.7.

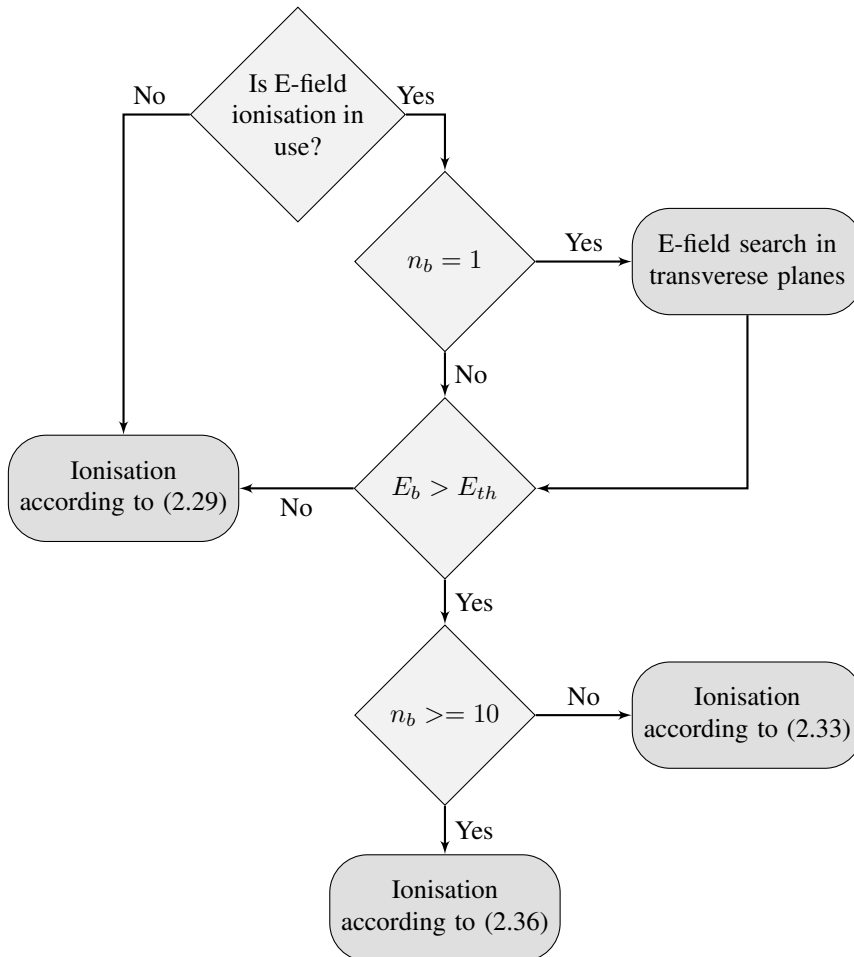


Figure 3.5: The flowchart shows how the ionisation mechanism is determined for the ion generation in PyECLOUD.

3.3 Pressure Studies in the CLIC Main LINAC

The machine parameters used for all simulations are listed in Table 3.1, unless something else has been specified. The simulations were run with the use of the CERN Batch Service, which is a computer cluster. Some of the simulations were also run locally on an HP Elite Mini with an intel i7 vPRO processor.

TABLE 3.1
CLIC MAIN LINAC PARAMETERS AND VALUES

Parameter	Symbol	Value
Energy	p_0 [GeV]	9 to 1500
Bunch Population	N	4×10^9
Norm. Horizontal emittance	[nm]	660
Norm. Vertical emittance	[nm]	10
Bunch length	σ_z [ns]	0.15×10^{-3}
Bunch spacing	ΔT_b [ns]	0.5
Bunches per train	n_b	312
Main LINAC Length	L [km]	20.5
Scattering ionisation cross-section	σ_{ions} [MBarn]	2.0
Electric field threshold	E_{th} [GV/m]	18

A set of simulations using a single grid setup and the dual grid setup were conducted for Xe over a pressure range of 500 nTorr to 5000 nTorr with a pressure step of 500 nTorr. The parameters used for the single and dual grid can be seen in Fig. 3.6 and Fig. 3.7 respectively. For these simulations the electric field ionisation feature was disabled.

```
PyPICmode = 'FFT_OpenBoundary'
x_aper = 15 * sigma_x
y_aper = 15 * sigma_y
Dh_x = x_aper / 125.
Dh_y = y_aper / 125.
Dh_sc = [Dh_x, Dh_y]
```

Figure 3.6: Parameters for creating the single grid setup used for FBII studies of CLIC main LINAC.

Two simulations with the single and the dual grid setup were run with electric field ionisation activated for Xe at a pressure of 1000.0 nTorr. A third set of simulations were run with the electric field ionisation enabled for all the gas species listed in Table 3.2 for

the pressures 2.0 nTorr, 5.0 nTorr, 10.0 nTorr, 20.0 nTorr, 40.0 nTorr, 70.0 nTorr, and 100.0 nTorr. These pressure studies were done with the single grid setup.

TABLE 3.2
GAS SPECIES AND ATOMIC MASS NUMBER

Gas Species	Atomic mass
H ₂ O	18
CO	28
CO ₂	44
Kr	84
Xe	132

```

PyPICmode = 'ShortleyWeller_WithTelescopicGrids'
x_aper = 3e-3
y_aper = 3e-3
Dh_ext = 3e-3 *2/300
Dh_sc = 1e-07
f_telescope = 0.02
target_grid = {'x_min_target': -4e-05, 'x_max_target': 4e-05,
               'y_min_target': -4e-05, 'y_max_target': 4e-05,
               'Dh_target': Dh_sc}
N_nodes_discard = 1.
N_min_Dh_main = 4

```

Figure 3.7: Parameters for creating the dual grid setup used for FBII studies of CLIC main LINAC.

3.4 Multiple Gas Species in the CLIC Damping Ring

The beam and machine parameters used for the multiple gas species simulations in the CLIC DR are listed in Table 3.3, and the gas species and pressure combination sets are found in Table 3.4. The overall pressure was kept at 60 nTorr for the different combinations of H₂, CO, and CO₂. The pressure values for CO and CO₂ were kept below or equal to H₂'s pressure value.

The time step Δt_{ref} was altered from 1.000 ns for the single species simulations setup, to 1.007 ns for the multiple gas species setup. The reason for this change is discussed in chapter 5. All the simulations were run using the single grid setup.

TABLE 3.3
CLIC DR PARAMETERS AND VALUES (NOT UPDATED)

Parameter	Symbol	Value
Energy	p_0 [GeV]	9 to 1500
Bunch Population	N	4×10^9
Norm. Horizontal emittance	[nm]	660
Norm. Vertical emittance	[nm]	10
Bunch length	σ_z [ns]	0.15×10^{-3}
Bunch spacing	ΔT_b [ns]	1.0
Bunches per train	n_b	156
Main LINAC Length	L [km]	20.5
Scattering ionisation cross-section	σ_{ions} [MBarn]	2.0

TABLE 3.4
PRESSURE COMBINATIONS FOR MULTIPLE GAS SPECIES STUDIES

Gas Species	H ₂	CO	CO ₂
Pressure set.1 [nTorr]	50.0	5.0	5.0
Pressure set.2 [nTorr]	50.0	0.0	10.0
Pressure set.3 [nTorr]	50.0	10.0	0.0
Pressure set.4 [nTorr]	40.0	10.0	10.0
Pressure set.5 [nTorr]	40.0	0.0	20.0
Pressure set.6 [nTorr]	40.0	20.0	0.0
Pressure set.7 [nTorr]	30.0	15.0	15.0
Pressure set.8 [nTorr]	30.0	0.0	30.0
Pressure set.9 [nTorr]	30.0	30.0	0.0
Pressure set.10 [nTorr]	20.0	20.0	20.0

4

Results

4.1 Electric field ionisation in PyECLoud and FASTION

The ionisation area limits calculated by FASTION and the electric field ionisation search implemented in PyECLoud can be seen in Fig. 4.1.

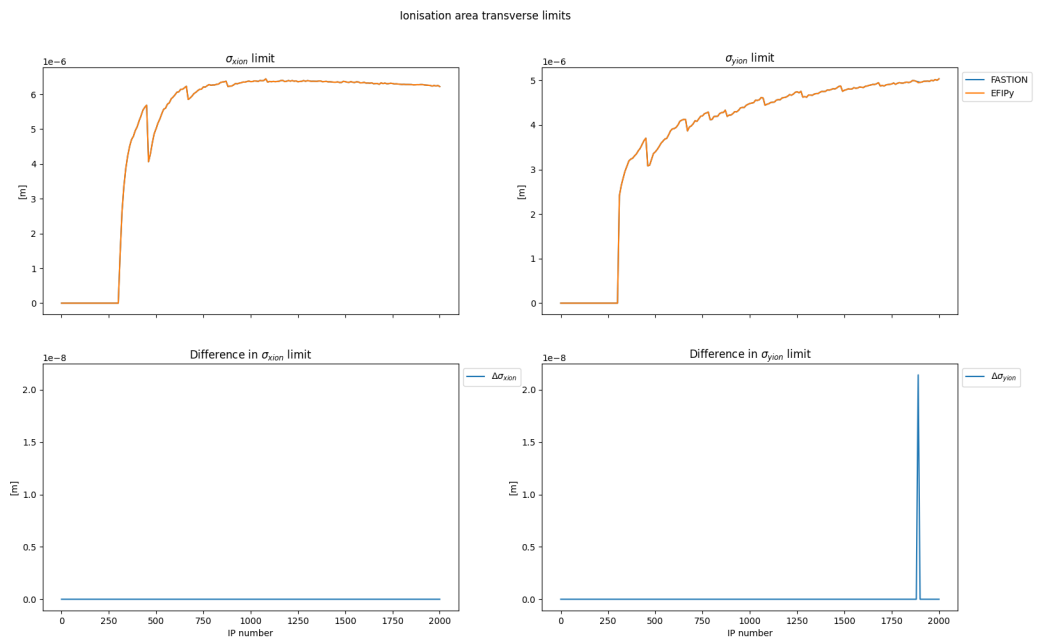


Figure 4.1: Ion limits

The figure also shows the difference in the $\sigma_{x_{\text{fion}}}$ and $\sigma_{y_{\text{fion}}}$ values calculated by the two.

The plots in Fig. 4.2 shows the number of generated ions by FASTION and PyE-CLOUD, the difference, and percentage difference between the two tools. Besides the peak towards the end of the IPs, the recorded difference had a mean and standard deviation of $(0.223 \pm 0.001)\%$ for the scattering process and electric field ionisation for the first 10 bunches, and $(0.146 \pm 0.001)\%$ for electric field ionisation after the first 10 bunches had passed.

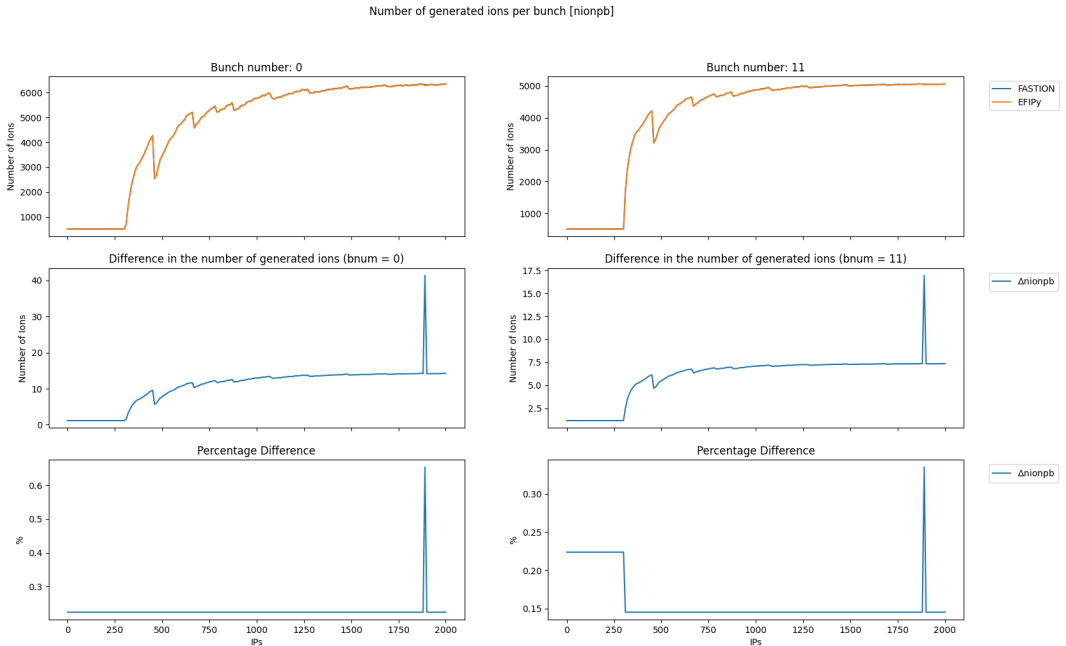


Figure 4.2: Number of ions generated in FASTION and PyE-CLOUD over the length of the CLIC main LINAC.

4.2 CLIC main LINAC simulation results

The results presented in this section are from the simulation studies described in section 3.3. The next section will present the result from the simulation studies described in section 3.4.

4.2.1 Without Electric Field Ionisation

The bunch train for a beam interacting with a residual gas of Xe with a pressure of 1000.0 nTorr at IP: 2005 is seen in Fig. 4.7. One of the plots shows the bunch train

obtained by one simulation using a single grid setup, and one simulation using a dual grid setup. The two simulations have been placed next to the bunch train obtained from a FASTION simulation where the gas pressure was 5.0 nTorr.

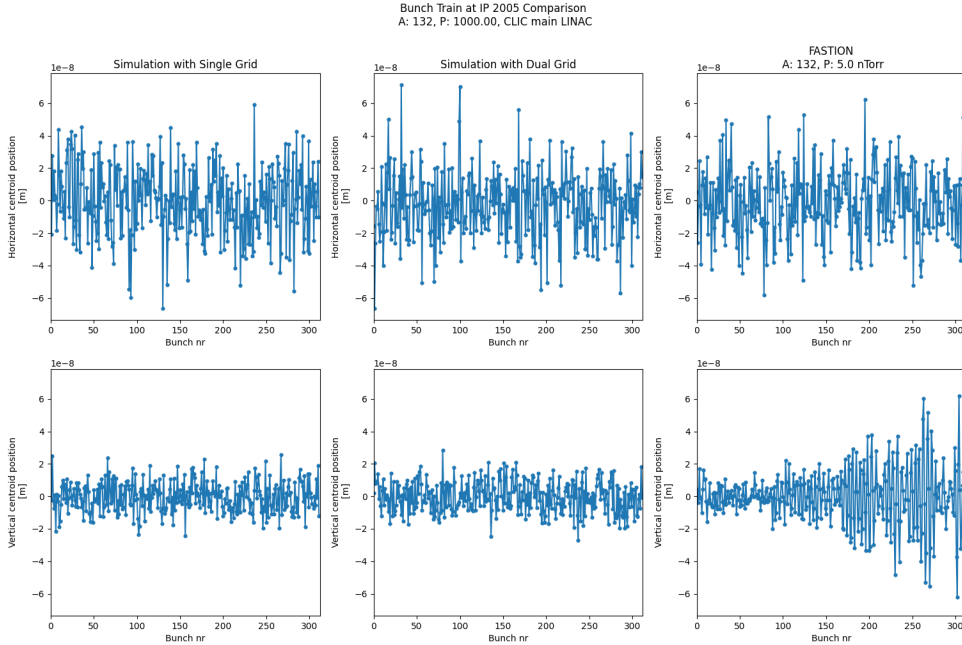


Figure 4.3: The left column shows the bunch train at IP:2005 for a simulation using a single grid. The middle column shows the same simulation using a dual grid. The two simulations were run without electric field ionisation activated in PyECLoud. The column on the right shows the bunch train for a simulation with A: 132 and P: 5.0 nTorr in FASTION.

The evolution of the centroid position of bunch 300 for the same three simulations is shown in Fig.4.4. The sigma values can be seen in Fig. 4.5. The maximum bunch centroid offset for the single and dual grid simulation for the pressure range of the simulations is seen in Fig. 4.6.

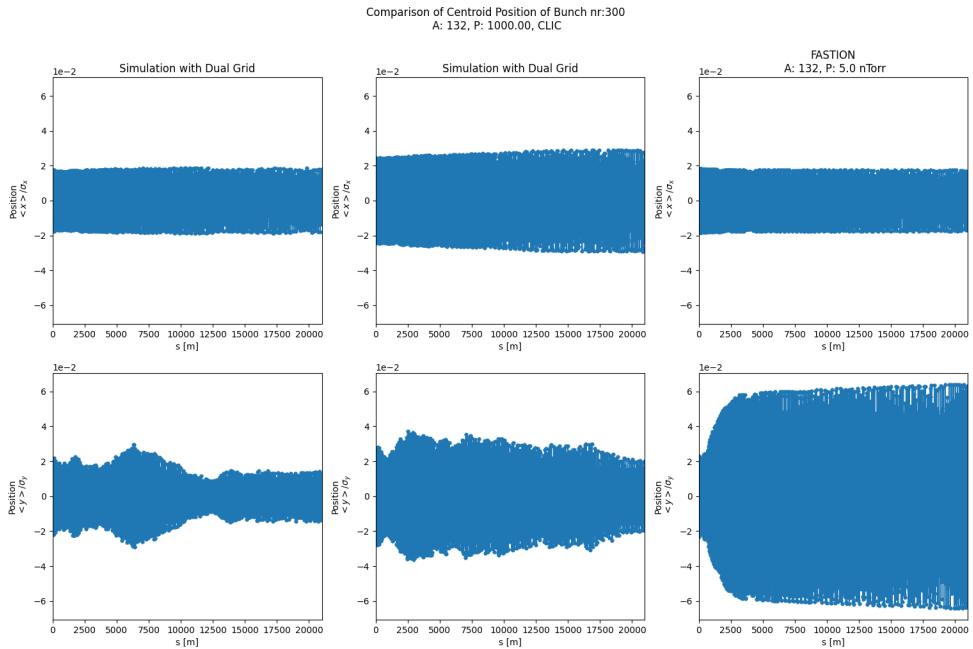


Figure 4.4: The left column shows the evolution of bunch 300 along the length of the LINAC for a simulation using a single grid. The middle column shows the same simulation using dual grid. The two simulations were run without electric field ionisation activated in PyECLOUD. The column on the right shows the bunch evolution for a simulation with A: 132 and P: 5.0 nTorr in FASTION.

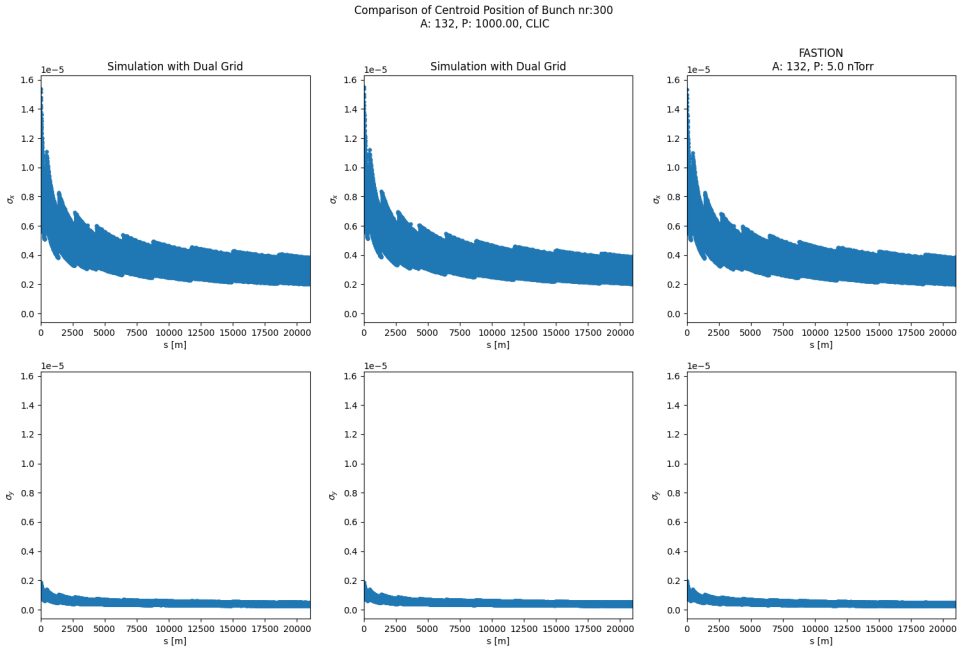


Figure 4.5: The left column shows evolution of the bunch transverse size along the length of the LINAC for a simulation using a single grid. The middle column shows the same simulation using dual grid. The two simulations were run without electric field ionisation activated in PyECLOUD. The column on the right shows the bunch transverse size evolution for a simulation with A: 132 and P: 5.0 nTorr.

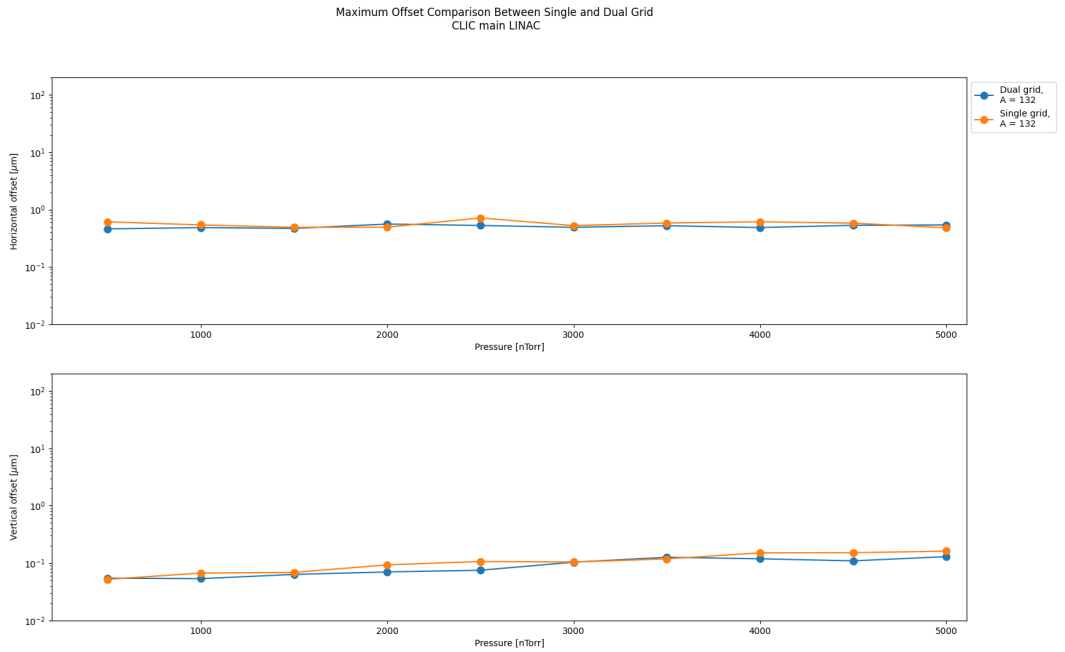


Figure 4.6: The two plots show the maximum offset of the bunch centroid over pressure for one simulation using a single grid and one using a dual grid. The simulations were run without electric field ionisation activated in PyECLLOUD.

4.2.2 With Electric Field Ionisation

The results for single and dual grid simulations of the main LINAC with a gas species with atomic mass 132 and a pressure of 1000.0 nTorr are shown in Fig. 4.7, Fig. 4.8, and Fig. 4.9. The simulations were done with electric field ionisation activated in PyECLLOUD. The results are presented next to the results from a FASTION simulation of a gas with an atomic mass number of 132 and a pressure of 5.0 nTorr, similar to what was done in section 4.2.1.

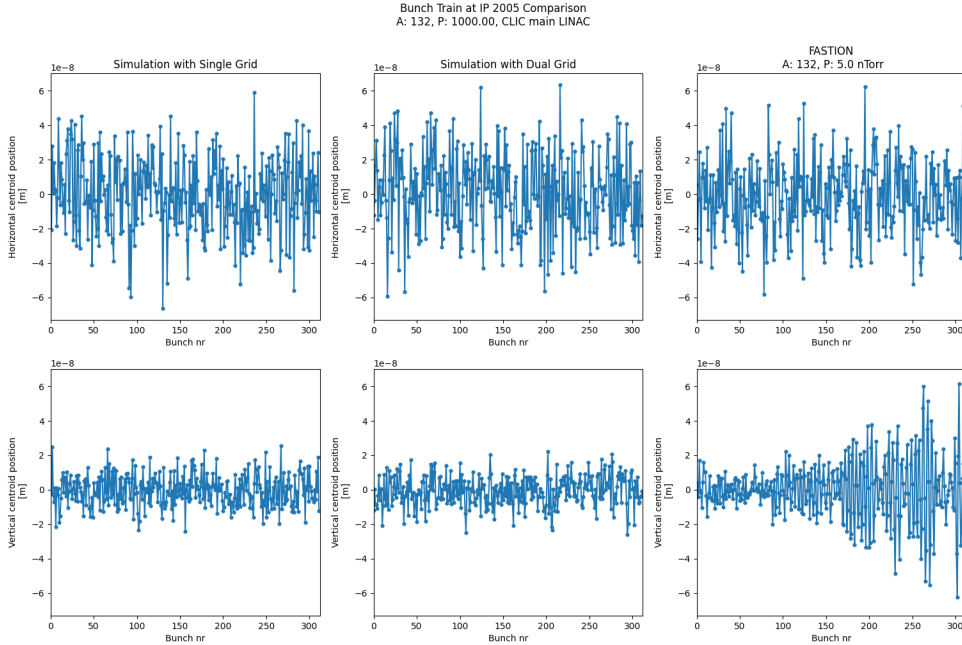


Figure 4.7: The left column shows the bunch train at IP:2005 for a simulation using a single grid. The middle column shows the same simulation using a dual grid. The two simulations were run with electric field ionisation activated in PyECLLOUD. The column on the right shows the bunch train for a simulation with A: 132 and P: 5.0 nTorr in FASTION.

The results for a pressure study of the gases listed in Table 3.2 are presented in Fig. 4.10, Fig. 4.11, and Fig. 4.12. The maximum offset of the bunch centroid in x and y for the gas species over pressure from the simulation is compared in Fig. 4.10 to the maximum offset from the same study done in FASTION.

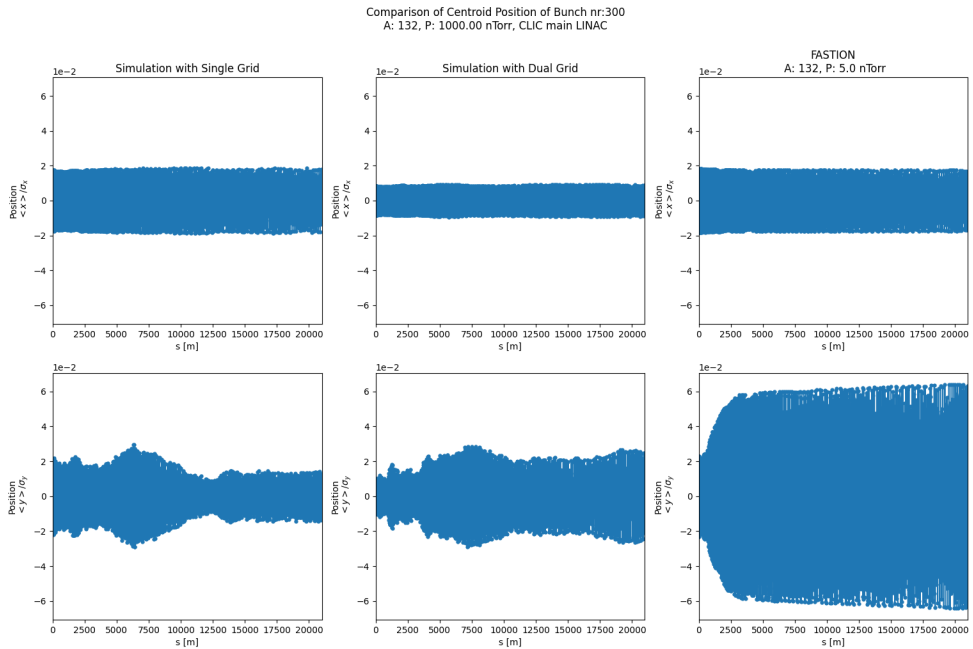


Figure 4.8: The left column shows the evolution of bunch 300 along the length of the LINAC for a simulation using a single grid. The middle column shows the same simulation using dual grid. The two simulations were run with electric field ionisation activated in PyECLOUD. The column on the right shows the bunch evolution for a simulation with A: 132 and P: 5.0 nTorr in FASTION.

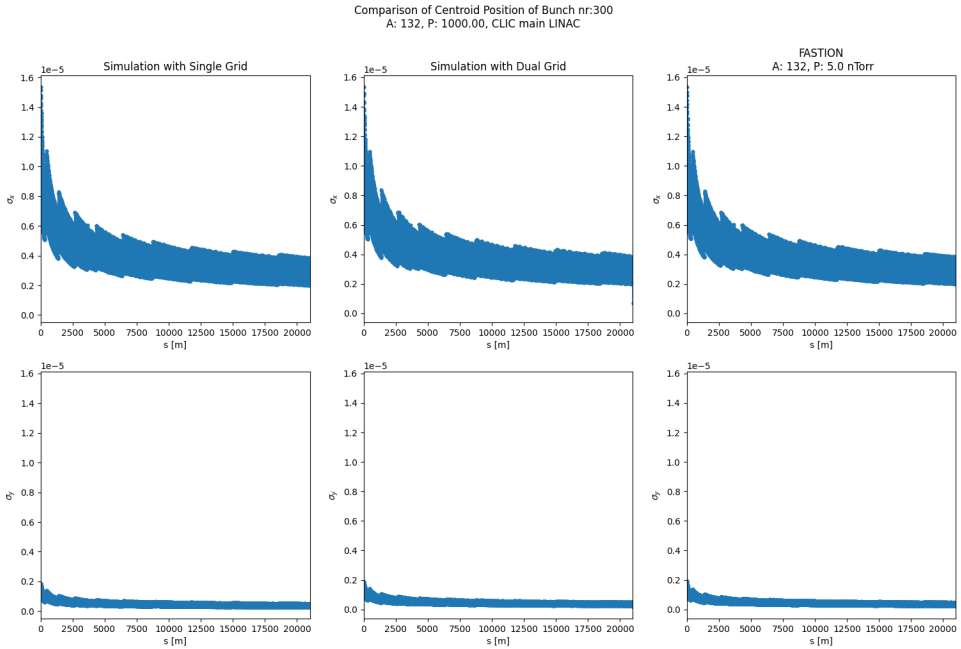


Figure 4.9: The left column shows evolution of the bunch transverse size along the length of the LINAC for a simulation using a single grid. The middle column shows the same simulation using dual grid. The two simulations were run with electric field ionisation activated in PyECLOUD. The column on the right shows the bunch transverse size evolution for a simulation with A: 132 and P: 5.0 nTorr

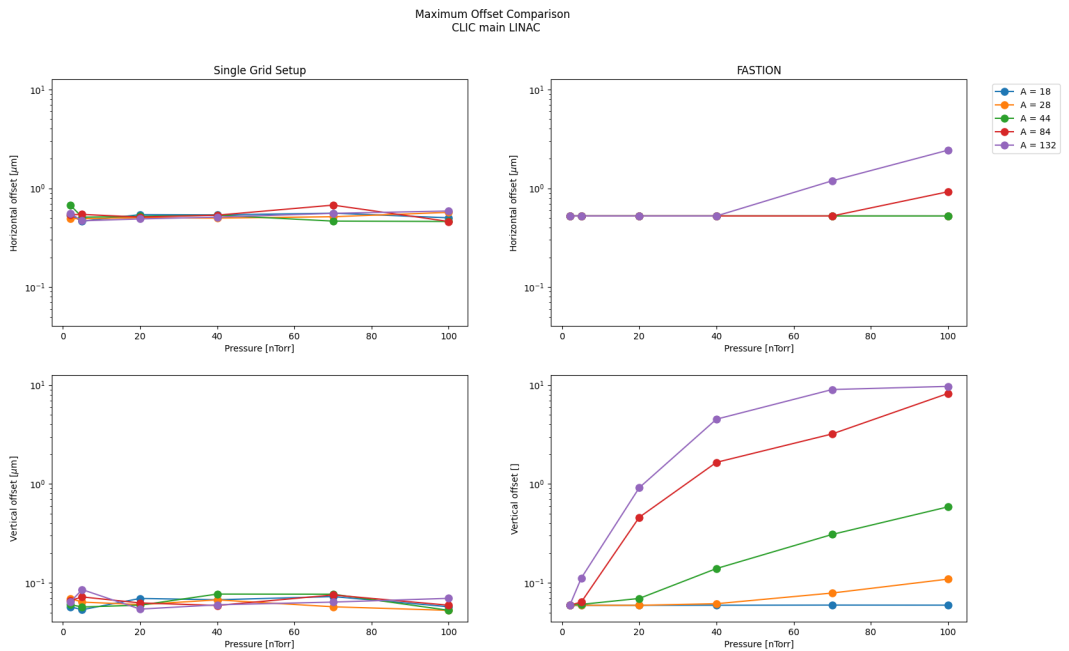


Figure 4.10: The column on the left show the maximum offset of the bunch centroid in x and y for all the gas species listed in Table 3.2 over pressure for one simulation using a single grid with electric field ionisation activated in PyECLOUD. The column on the right shows the results for the same simulation done in FASTION [8].

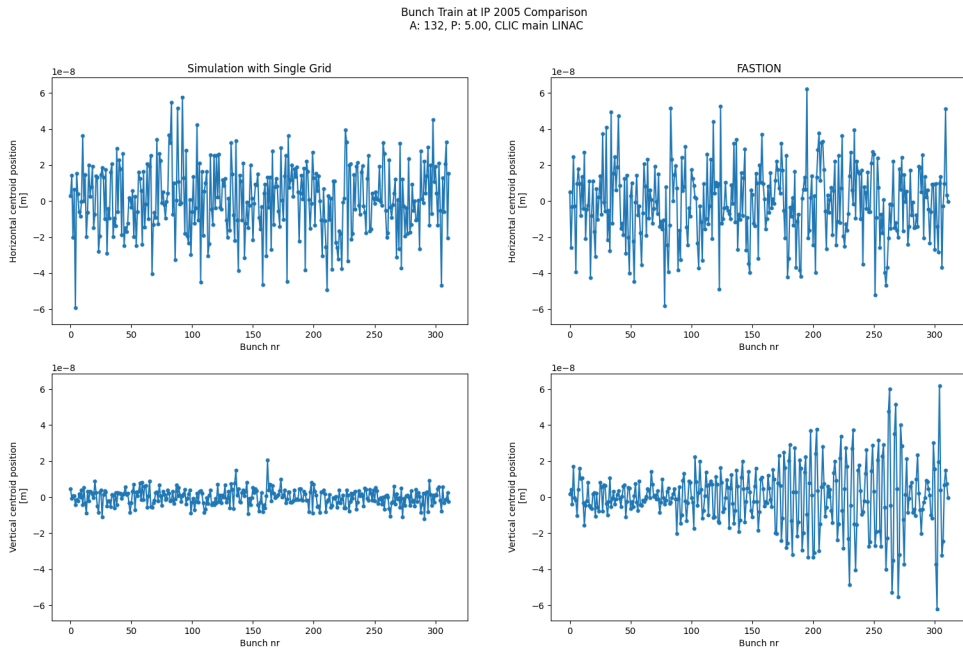


Figure 4.11: The left column shows the bunch train at IP:2005 for a simulation using a single grid, where the gas has atomic mass number of 132 and a pressure 5.0 nTorr. The column on the right shows the results for the same simulation done in FASTION [8].

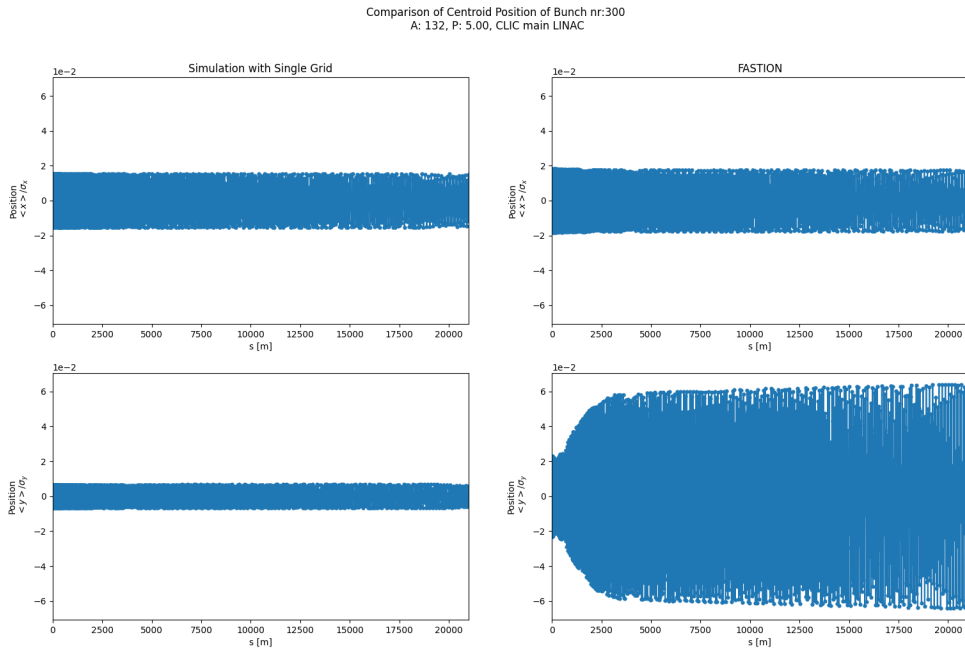


Figure 4.12: The left column shows evolution of the bunch transverse size along the length of the LINAC for a simulation using a single grid, where the gas has atomic mass number of 132 and a pressure 5.0 nTorr. The column on the right shows the results for the same simulation done in FASTION [8].

4.3 Multiple gas species in CLIC DR

The bunch train at IP: 258 in the CLIC DR can be seen in Figure 4.13 for the pressure sets 1 through 9 from Table 3.4. The centroid offsets have been normalised with respect to the corresponding beam transverse size.

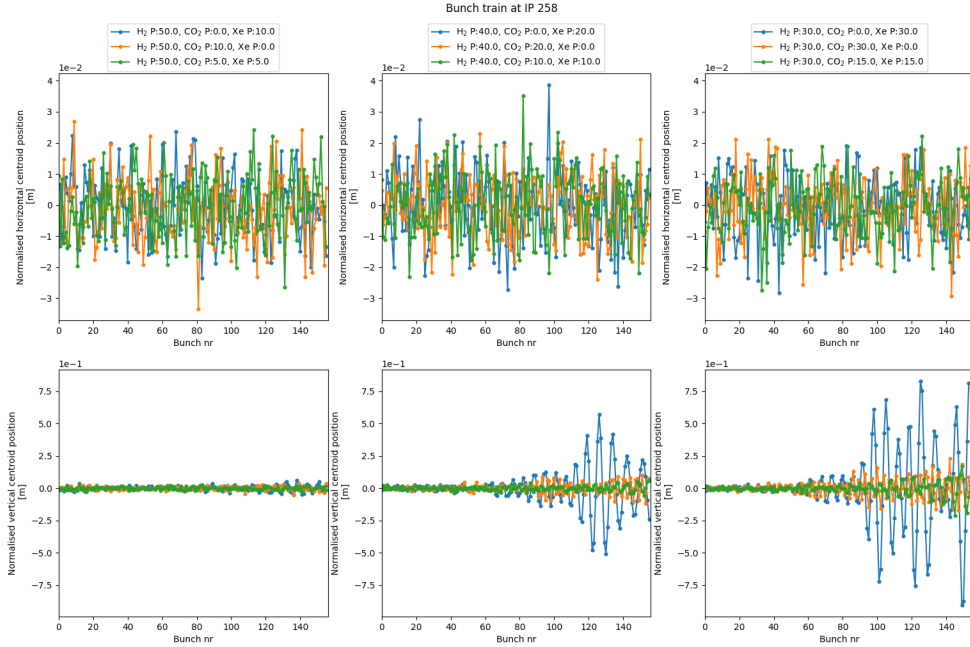


Figure 4.13: The left column shows the bunch train in x and y at IP: 258 in the CLIC DR for the pressure sets 1, 2, and 3, found in Table 3.4. The middle plots pressure sets 4, 5, and 6, while column on the right shows the bunch train for pressure sets 7, 8, and 9.

5

Discussion

First and foremost, FASTION is not a perfect modelling tool and has been built with its own assumptions and approximations. Therefore the comparisons presented are only considered as guidance and not as definite. Furthermore, it is seen in the coordinate assignment for the ion macroparticle in section 3.2 that there is an amount of randomness implemented in the tools. Thus, it is unreasonable to expect the results from FASTION and the Python tools to be identical.

5.1 Ion Generation in FASTION and PyECLLOUD

The standard difference of 0.22% for the scattering case and a 0.14% standard difference for the field ionisation case when comparing the number of ions generated per bunch between FASTION and the implemented electric field ionisation feature in PyECLLOUD PyHEADTAIL. This difference arises from the fact that FASTION uses a constant value for $k_B T$ in (2.29) evaluated at roughly $T = 300\text{K}$, while PyECLLOUD-PyHEADTAIL calculates it based on the user input of T .

The peak seen in Fig. 4.1 and Fig. 4.2 was caused by a minor standard difference in the normalised electric field calculated from the ionisation areas search conducted by FASTION and the electric field ionisation feature. This standard difference has a value of 6.47×10^{-6} , which in most cases would be too small to have a significant impact. However, for very specific values of σ_x , the difference would result in the scan of σ_{xion} to stop at a different iteration. This in turn would propagate and result in a different number of generated ions. The standard difference was found to arise from the fact that FASTION had defined the electron charge to be $1.6021892 \times 10^{-19}$ C instead of the elemental charge of $1.6021766 \times 10^{-19}$ C.

During testing and implementation of the electric field ionisation feature into PyECLLOUD, it was observed that the Bassetti-Erskine function used in PyECLLOUD was unable to calculate for the case where the beam was perfectly circular ($\sigma_x = \sigma_y$). This shortcoming was not present in the equivalent calculation function in FASTION.

5.2 Simulation Results

5.2.1 Results from the CLIC Main LINAC

The results presented in section 4.2.1 do not exhibit the characteristic blow up of the tail centroid position that is associated with FBII. It was initially believed that this was due to ions being lost to the boundary of the PIC grid, which was defined as a perfect absorber, or that the bunch was insufficiently resolved on the grid. In the event that the bunch was insufficiently resolved on the grid, then the centroid motion of the bunches will not be propagated from one IP to the next in a manner for the instability to manifest. Thus the simulation was repeated using the dual grid setup described in 3.1.1. The comparison of the results in Fig. 4.3 and Fig. 4.4, show that the issue lies elsewhere. This assessment is further cemented by Fig. 4.6, which clearly shows that the dual grid has no discernible effect on the bunch centroid position.

Inspection of the FASTION output files from [8] showed that the electric field ionisation feature was activated during the simulation of the CLIC main LINAC. It can be seen in Fig. 4.2 that electric field ionisation generates significantly more ions than collision ionisation. Since, the maximum centroid offset of the bunch can be seen in Fig. 4.6 to be weakly dependent on the gas pressure. It is assumed that there are not enough generated ions when simulating the CLIC main LINAC with only collision ionisation.

Hence, the simulation was conducted again for a pressure of 1000.0 nTorr with the electric field ionisation feature activated in PyECLOUD. Due to time constraints, a pressure study was conducted in parallel for a range of different gas species. The results from these simulations presented in section 4.2.2 clearly show that the activation of the electric field ionisation feature had no impact on the issue.

A possible reason is that the number of ions generated by the electric field ionisation feature is not comparable to the number of ions generated in FASTION. The results in Fig. 4.1 and Fig. 4.2 were obtained from tests conducted before the feature was implemented into PyECLOUD. However, it has not been tested properly after implementation into PyECLOUD.

Another possible reason is that the ions are somehow lost between the bunches. The ion loss could be caused by the bunch spacing in the simulation being different from what it is believed to be. The ions could also be lost if the ion macroparticles are not properly transferred between the bunches in the code. Further investigation is required to resolve this issue.

The decrease in transverse bunch size seen in Fig. 4.5 and Fig. 4.9 indicates that the bunches are being accelerated by the energy update object added to the machine map as described in section 3.1. In combination with the relation between gas pressure and centroid position, it is reasonable to assume that the poor modelling of FBII in the machine is not due to the adaption of the synchrotron class. Thus, the LINAC machine has successfully been modelled in PyECLOUD and PyHEADTAIL.

5.2.2 Results from the CLIC DR

It can be seen in Fig. 4.13 that the tail of the bunch train becomes more unstable the large ratio between the light gas H_2 and the heavier CO_2 and Xe. Moving from left to

right, the tail of the bunch train in the vertical plane increases more as the amount of H_2 decreases and the amount of heavier gas increases. The relation between atomic mass number and beam instability is clearly visible in the middle and right column. The pressure combinations with H_2 and CO_2 result in a more stable beam than the pressure combinations with H_2 and Xe . This is because of Xe comparatively higher atomic mass with respect to CO_2 . Interestingly, the pressure combination with H_2 and CO_2 manifests earlier in the bunch train than the pressure combination with all three gases, even though the combined atomic mass is greater in the latter case. Furthermore the two combinations result in comparable centroid offset towards the end of the bunch train. Further investigation is required to determine if this is due to instability induced by CO_2 having a smaller time constant, and thus developing quicker, than the instability induced by the combined gases. Overall, the results in section 4.3 agree with the known theory presented in section 2.3.

5.3 Some Assumptions and Approximations

The instability modelling tools are created such that the any potential vacuum limits found through simulation are determined by overestimation. This is also motivated by the fact that there are other instabilities and effects present that can either enhance or damp the instability being modelled.

An example of such an over estimate is the electric field threshold for electric field ionisation. The simulations conducted in the project defined the ionisation threshold at 18 GVm^{-1} . However table 2.3 shows that non of the species used are ionised at this electric field strength. Effectively, the simulation assumes that a species will be ionised by the beam's electric field when in field ionisation mode, even if the species, such as CO_2 , in reality has an electric field threshold of 21.5 GVm^{-1} .

The FASTION code applies a correction factor for the Bassetti-Erskine equation when it calculates the bunch-generated electric field during the electric field ionisation search. This correction factor was not included in PyECLLOUD when conducting the same search. Since the search only covers $\pm 10\sigma_x$ from the bunch centroid, and the need to apply the correction factor occurs beyond $\pm 40\sigma_x$.

5.4 Challenges

It was mentioned in section 5.2.1 that the pressure study was run in parallel due to time constraints. Part of this was due to the fact that simulations using the single grid setup took roughly 81301 s when run on the local computer and 129636 s on CERN's computer cluster. This computation time significantly slowed down speed at which bugs and other errors could be identified, since they would only show up after the simulation had been running for several hours. Some of these bugs only occurred when running the simulations on the computer cluster, which reduced the advantage of running bug test simulations on the local computer. The computation time became a significant challenge when conducting dual grid simulations, where the computation time increased to 295975.0 s when run on the local computer. The time constraint imposed by the computation time contributed to the multiple gas species feature only being tested for the CLIC DR.

Another reason why the feature was not tested on the CLIC main LINAC had to do with how the simulation script had to be changed for the feature to function. It was mentioned in section 3.4 that Δt_{ref} had to be changed to 1.007 ns. This change had to be made because of the inclusion of an output location for saving cloud data. When setting up the simulation script to include the multiple species feature, the script required an output file for the different clouds. This is because the multiple gas species feature is often used for electron cloud buildup simulations. Since the cloud output location has been defined and is no longer None, the script conducts an initialisation of a python class dedicated to saving the data from the gas clouds. In this initialisation, a Δt for the saver is defined by:

$$\Delta t_{\text{saver}} = (V - 0.0001) \times \Delta t_{\text{ref}} \quad (5.1)$$

where V is some variable declared in the python class. At this stage Δt_{ref} defined in the simulation script has not been collected, and the saver uses the value from the `simulation_parameters.input`. As such, it can not be set to None in the file, like done for a single gas species. Defining Δt_{ref} in `simulation_parameters.input` causes a different error to arise after the initialisation has completed and the script starts to loop through the machine map objects. As the script begins the particle tracking, it has to calculate the length of the slice in units of time:

$$\delta t_{\text{slice}} = \frac{dz}{\beta_r \times c} \quad (5.2)$$

where dz is the length of the slice. It then checks that length of the slice is not longer than Δt_{ref} as seen in alg. 1.

Algorithm 1 Extract from Track Single Slice

```

...
 $\delta t_{\text{slice}} = dz \times (\beta_r \times c)^{-1}$ 
if  $\Delta t_{\text{ref}} \neq \text{None}$  then
  if  $\delta t_{\text{slice}} > \Delta t_{\text{ref}}$  then
    rise Error ("Slice cannot be longer than buildup time step")
  else
    ...
  end if
...
else
  ...
end if
...

```

For the CLIC DR the slice length was slightly longer than Δt_{ref} by 7×10^{-3} ns, which only constituted a small fraction of Δt_{ref} . Thus it was not seen as an issue to increase Δt_{ref} from 1.000 ns to 1.007 ns when simulating the damping ring. However, for the main LINAC the slice length was calculated to be significantly larger with $\delta t_{\text{slice}} = 49$ ns, while $\Delta t_{\text{ref}} = 0.5$ ns. Thus the same approach could not be applied

to the main LINAC. Furthermore, the saver creates a file in the output location for each cloud file at every IP, resulting in a significant number of files created for the relatively short CLIC DR. Had the multiple gas species feature functioned for the main LINAC, it would have resulted in 312 files for each of the 2010 machine segments per gas species.

6

Future Work

Based on some of the points brought up in chapter 5, there are a few areas that future work on FBII simulation in the CLIC main LINAC should be directed. One of these is the establishment of a LINAC class in PyHEADTAIL like the synchrotron class that was adapted in chapter 3. The inclusion of a LINAC python class would make the setup for simulation studies of other LINACs simpler for users that are not familiar with the modelling tools. This should however have a low priority as it is not as critical as other areas.

The most critical aspect that future work should be directed towards is the lack of instability seen for the LINAC at lower pressures. A systematic approach should be conducted to address the possible reasons outlined in chapter 5. A focus should first and foremost ensure that the number of ions generated by the electric field ionisation feature matches its performance outside of PyECLOUD.

It is reasonable to assume that if the previously mentioned issue had been resolved, then the performance of the single and dual grid would not necessarily had been as comparable as they seemed in Fig. 4.6. It is clear that future studies will require the dual grid feature to ensure proper modelling of beam instabilities for flat shaped beams. If the dual grid is to be utilised, then the computation time of the dual grid has to be reduced. The square shaped grid cells are computationally inefficient due to the flat shape of the CLIC main LINAC bunches. Adopting non-square grid cells of rectangular shape would require less computation time by reducing the resolution along the x -axis without sacrificing resolution along the y -axis. Making these changes to PyPIC should be a high priority for future work, as it will be very computationally beneficial for future studies of the CLIC main LINAC.

The primary focus of future work in regards to the multiple gas species setup revolves around the removal of the cloud saver in the initialisation of PyEC4PyHT. This would consequently remove the need to declare Δt_{ref} in `simulation_parameters.input`, which in turn would eliminate issue with size check of Δt_{ref} discussed in section ???. This would ultimately also eliminated the issue with the number of cloud data files created during a simulation run. The potential number of cloud data files that would be created for

a simulation of the CLIC main LINAC is unnecessary and unsustainable.

The multiple species setup should be improved further with a form of cloud manager to handle the creation and alteration of cloud files directly in the simulation script. This could also facilitate for inclusion of species-defined electric field threshold, ionisation cross section, and electric field ionisation probability. The cloud manager would primarily make pressure studies of multiple gas species in machines easier to setup.

In section 1.2 it is mentioned that the CLIC main LINAC can experience vacuum breakdown, which can cause increased local pressure. Therefore, future work should look into simulations with non-uniform pressure profiles along the machine length to investigate the effect it would have on FBII. Such an investigation could be extended further by consulting with the Vacuum, Surfaces and Coatings group (VSC) at CERN's Technology Department (TE) to determine pressure profiles and gas compositions that would be realistic to observe in the CLIC main LINAC.

7

Conclusion

In conclusion, although new vacuum specifications were not established for the CLIC main LINAC, the project has laid a solid foundation for the completion of specific pressure studies in the future. This was achieved through the adaptation of the synchrotron Python class for the simulation of FBII in a LINAC, and establishing the feasibility of multiple gas species simulations with the PyEC4PyHT interface, by conduction studies of different combinations of H₂, CO₂, and Xe in the CLIC's DR. The results from the multiple gas species simulations corresponded well with theory. These studies also identified steps that need to be taken in the future to ensure that the feature can be used for similar studies in the CLIC's main LINAC.

Several areas have been identified for future work throughout the project. This includes, but is not limited to:

- The expansion of the multiple internal grid feature in PyPIC to accommodate for non-square shaped grid cells.
- Further investigation into the electric field ionisation mechanism to ensure correct physical modelling.
- Testing and studies of non-uniform pressure profiles along the accelerator machine.

References

- [1] K. Wille, *The physics of particle accelerators: an introduction*. Oxford ; New York: Oxford University Press, 2000.
- [2] R. Nagaoka, in Proceedings of the CAS–CERN Accelerator School: Intensity Limitations in Particle Beams, Geneva, Switzerland, 2–18 November 2015, edited by W. Herr, CERN Yellow Reports: School Proceedings, Vol. 3/2017, CERN-2017-006-SP (CERN, Geneva, 2017), pp. 519 – 556, DOI: <https://doi.org/10.23730/CYRSP-2017-003.519>.
- [3] J. B. Jeanneret, G. Rumolo, and D. Schulte, “Vacuum specifications for the CLIC Main LINAC,” in *Proc. of 1st International Particle Accelerator Conf. (IPAC’10)*, Kyoto, Japan, 2010, pp. 3401–3403. [Online]. Available: <https://cds.cern.ch/record/1271352>
- [4] A. Abada, M. Abbrescia, S. S. AbdusSalam *et al.*, “FCC-ee: The Lepton Collider,” *Eur. Phys. J. Spec. Top.*, vol. 228, pp. 261–623, 2019, <https://doi.org/10.1140/epjst/e2019-900045-4>.
- [5] CERN, “Accelerator Overview.” [Online]. Available: <https://clic-study.web.cern.ch/accelerator-overview>
- [6] W. Hillert, proceedings of the CERN–Accelerator–School course: Introduction to Particle Accelerators pp. 230 – 294, arXiv:2107.02614.
- [7] M. Lonza and H. Schmickler, in Proceedings of the CAS–CERN Accelerator School: Advanced Accelerator Physics, Trondheim, Norway, 18–29 August 2013, edited by W. Herr, CERN–2014–009 (CERN, Geneva, 2014), pp. 503 – 546, DOI: <http://dx.doi.org/10.5170/CERN-2014-009.503>.
- [8] F. Grønvold, “Modelling of fast beam-ion instability in particle accelerators,” TTK4551 Specialization project report, Dept. of Engineering Cybernetics, Norwegian University of Science and Technology, 2022.
- [9] CERN, “Launch of the LHC Run 3,” Accessed: Feb. 27, 2023. [Online]. Available: <https://home.cern/events/launch-lhc-run-3>

-
- [10] —, “LHC Run 3,” Accessed: Feb. 27, 2023. [Online]. Available: <https://home.cern/press/2022/run-3>
- [11] —, “LHC Run 3: physics at record energy starts tomorrow,” Accessed: Mar. 3, 2023. [Online]. Available: <https://home.cern/news/news/physics/lhc-run-3-physics-record-energy-starts-tomorrow>
- [12] —, “LS3 schedule change,” Accessed: Feb. 27, 2023. [Online]. Available: <https://hilumilhc.web.cern.ch/article/l3-schedule-change>
- [13] European Strategy Group, “2020 Update of the European Strategy for Particle Physics (Brochure),” CERN, Geneva, Tech. Rep., 2020. [Online]. Available: <https://cds.cern.ch/record/2721370>
- [14] A. Abada, M. Abbrescia, S. S. AbdusSalam *et al.*, “FCC-hh: The Hadron Collider,” *Eur. Phys. J. Spec. Top.*, vol. 228, pp. 755–1107, 2019, <https://doi.org/10.1140/epjst/e2019-900087-0>.
- [15] F. Borzumati and E. Kato, “The higgs boson and the international linear collider,” *Frontiers in Physics*, vol. 2, jun 2014. [Online]. Available: <https://arxiv.org/abs/1407.2133>
- [16] M. Ruan, “Higgs measurement at e^+e^- circular colliders,” *Nucl. Part. Phys. Proc.*, vol. 273-275, pp. 857–862, 2016, comments: 6 pages. Report at ICHEP 2014. [Online]. Available: <https://cds.cern.ch/record/1971660>
- [17] CEPC Accelerator Study Group, “Snowmass 2021 White Paper AF3-CEPC,” 2022. [Online]. Available: <https://arxiv.org/abs/2203.09451>
- [18] X. C. Lou, “The Circular Electron Positron Collider,” *Nature Reviews Physics*, vol. 1, pp. 232–234, 2019. [Online]. Available: <https://doi.org/10.1038/s42254-019-0047-1>
- [19] CERN, “Overview,” Accessed: Dec. 5, 2022. [Online]. Available: <https://fcc.web.cern.ch/overview>
- [20] “The Compact Linear Collider (CLIC) - 2018 Summary Report,” edited by P.N. Burrows, N. Catalan Lasheras, L. Linssen, M. Petrič, A. Robson, D. Schulte, E. Sicking, S. Stapnes, CERN Yellow Reports: Monographs, Vol. 2/2018, CERN-2018-005-M (CERN, Geneva, 2018). DOI: <https://doi.org/10.23731/CYRM-2018-002>.
- [21] CERN, “FAQ,” Accessed: Dec. 5, 2022. [Online]. Available: <https://clic-study.web.cern.ch/faq>
- [22] —, “The compact linear collider,” Accessed: Dec. 5, 2022. [Online]. Available: <https://home.web.cern.ch/science/accelerators/compact-linear-collider>
- [23] —, “How did we discover the Higgs boson?” Accessed: Feb. 27, 2023. [Online]. Available: <https://home.cern/science/physics/higgs-boson/how>

-
- [24] A. Oeftiger and G. Rumolo, “Fast beam-ion instabilities in clic main linac vacuum specifications,” CERN, Geneva, Switzerland, Tech. Rep. CERN-OPEN-2011-050, CLIC-Note-930, 2011. [Online]. Available: <https://cds.cern.ch/record/14060500>
- [25] L. Mether, in Proceedings of the ICFA mini-Workshop on Mitigation of Coherent Beam Instabilities in Particle Accelerators, Zermatt, Switzerland, 23–27 September 2019, E. Métral, G. Rumolo, T. Pieloni (eds.), CERN-2020-009 (CERN, Geneva, 2020), pp. 15 – 22, DOI: <http://doi.org/10.23732/CYRCP-2020-009.15>.
- [26] G. Rumolo and D. Schulte, “Fast Ion Instability in the CLIC Transfer Line and Main Linac,” in *Proc. of 11th European Particle Accelerator Conf. (EPAC’08)*, Genoa, Italy, 2008, pp. 655–657.
- [27] S. Stapnes, “Clic looks towards 2025,” *Accelerating News*, no. 42, 2022. [Online]. Available: <https://acceleratingnews.web.cern.ch/news/issue-42/compact-linear-collider-clic/clic-looks-towards-2025>
- [28] G. Rumolo and D. Schulte, “Update on Fast Ion Instability Simulations for the CLIC Main LINAC,” in *Proc. of 23rd Particle Accelerator Conf. (PAC’09)*, Vancouver, BC, Canada, 2009, pp. 4658–4660.
- [29] L. Mether, G. Iadarola, and G. Rumolo, “Numerical Modeling of Fast Beam-Ion Instabilities,” in *Proc. of 57th ICAF Advanced Beam Dynamics Workshop on High-Intensity and High-Brightness Hadron Beams (HB2016)*, Malmö, Sweden, 2016, pp. 368–372.
- [30] A. Chatterjee, K. Blaser, W. Hartung, D. Rubin, and S. T. Wang, “Fast ion instability at the cornell electron storage ring test accelerator,” *Phys. Rev. ST Accel. Beams*, vol. 18, p. 064402, Jun 2015. [Online]. Available: <https://link.aps.org/doi/10.1103/PhysRevSTAB.18.064402>
- [31] T. O. Raubenheimer and F. Zimmermann, “Fast beam-ion instability. I. Linear theory and simulations,” *Phys. Rev. E*, vol. 52, pp. 5487–5498, Nov 1995. [Online]. Available: <https://link.aps.org/doi/10.1103/PhysRevE.52.5487>
- [32] K. Li, in Proceedings of the CAS–CERN Accelerator School: Intensity Limitations in Particle Beams, Geneva, Switzerland, 2–18 November 2015, edited by W. Herr, CERN Yellow Reports: School Proceedings, Vol. 3/2017, CERN-2017-006-SP (CERN, Geneva, 2017), pp. 247 – 290, DOI: <https://doi.org/10.23730/CYRSP-2017-003.247>.
- [33] CERN, “Accelerating: Radiofrequency cavities,” Accessed: Dec. 5, 2022. [Online]. Available: <https://home.web.cern.ch/science/engineering/accelerating-radiofrequency-cavities>
- [34] K. Li, in Proceedings of the CERN–Accelerator–School course: Introduction to Particle Accelerators, pp. 449–501. [Online]. Available: <http://arxiv.org/abs/2107.06109>
-

-
- [35] H. Bartosik, Y. Papaphilippou, and A. Wolski, proceedings of the CERN–Accelerator–School course: Introduction to Particle Accelerators pp. 295 – 330, arXiv:2201.01532.
- [36] M. Bassetti and G. A. Erskine, “Closed expression for the electrical field of a two-dimensional Gaussian charge,” CERN, Geneva, Tech. Rep., 1980. [Online]. Available: <https://cds.cern.ch/record/122227>
- [37] J. Byrd, A. Chao, S. Heifets *et al.*, “First Observations of a “Fast Beam-Ion Instability”,” *Phys. Rev. Lett.*, vol. 79, pp. 79–82, Jul 1997. [Online]. Available: <https://link.aps.org/doi/10.1103/PhysRevLett.79.79>
- [38] L. Mether, A. Oeftiger, and G. Rumolo, in Proceedings of the ICFA Mini-Workshop on Impedances and Beam Instabilities, Benevento, Italy, 18–22 September 2017, edited by V. Brancolini, G. Rumolo, M. R. Masullo and S. Petracca, CERN Yellow Reports: Conference Proceedings, Vol. 1/2018, CERN-2018-003-CP (CERN, Geneva, 2018), pp. 63 – 68, DOI: <https://doi.org/10.23732/CYRCP-2018-001.63>.
- [39] D. R. Bett, N. Blaskovic Kraljevic, T. Bromwich *et al.*, “High-resolution, low-latency, bunch-by-bunch feedback system for nanobeam stabilization,” *Phys. Rev. Accel. Beams*, vol. 25, p. 022801, Feb 2022. [Online]. Available: <https://link.aps.org/doi/10.1103/PhysRevAccelBeams.25.022801>
- [40] D. Filippetto, C. Serrano, Q. Du *et al.*, “Feedback and control systems for future linear colliders: White Paper for Snowmass 2021 Topical Group AF07-RF,” 2022. [Online]. Available: <https://arxiv.org/abs/2204.00701>
- [41] G. Iadarola, E. Belli, K. Li *et al.*, “Evolution of Python Tools for the Simulation of Electron Cloud Effects,” in *Proc. of 8th International Particle Accelerator Conf. (IPAC’17)*, Copenhagen, Danmark, 2017, pp. 3803–3806.
- [42] A. Oeftiger, “An Overview of PyHEADTAIL,” *CERN-ACC-NOTE-2019*, 2019. [Online]. Available: <https://cds.cern.ch/record/2672381>
- [43] CERN, “PyHEADTAIL,” <https://github.com/PyCOMPLETE/PyHEADTAIL>, 2023.
- [44] G. Iadarola, E. Belli, P. Dijkstal *et al.*, “PyELOUD Reference Manual,” Accessed: Jan. 16, 2023. [Online]. Available: <https://github.com/PyCOMPLETE/PyELOUD/wiki/Reference-Manual>

Appendix

A Linear Beam Dynamics

The following derivation of the transverse linear equations of motion are adapted from [1] and [6].

Consider the charged particle q travelling parallel to the ideal path of a reference particle around a circular accelerator. The location of q in relation to the reference particle can be described by the Cartesian coordinate system $(\hat{e}_x, \hat{e}_y, \hat{e}_z)$, whose origin follows the orbit of the reference particle along the ideal path.

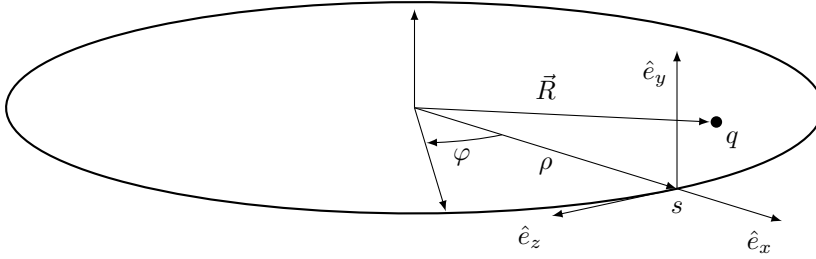


Figure 7.1: Motion from position s_0 to s_1 on the orbit which is described by the transfer matrix

The arc length parameter s is measured along the ideal orbit starting from a defined position. This position is often the injection point of the accelerator or a symmetry point of the synchrotron. Since the origin of the coordinate system follows the reference particle, the coordinate system is rotated by the angle φ about the \hat{e}_y -axis as the particle travels from s_0 to s_1 :

$$\begin{bmatrix} \hat{e}_x \\ \hat{e}_y \\ \hat{e}_z \end{bmatrix} \Big|_{s_1} = \begin{bmatrix} \cos \varphi & 0 & \sin \varphi \\ 0 & 0 & 0 \\ -\sin \varphi & 0 & \cos \varphi \end{bmatrix} \begin{bmatrix} \hat{e}_x \\ \hat{e}_y \\ \hat{e}_z \end{bmatrix} \Big|_{s_0} \quad (7.1)$$

Deriving the unit vectors with respect to the rotation angle gives:

$$\frac{d\hat{e}_x}{d\varphi} = \hat{e}_z \quad (7.2a)$$

$$\frac{d\hat{e}_z}{d\varphi} = -\hat{e}_x \quad (7.2b)$$

$$\frac{d\hat{e}_y}{d\varphi} = 0 \quad (7.2c)$$

Since the path element $ds = \rho d\varphi$, where ρ is the radius of the ideal path, then the

derivative of the unit vectors with respect to s are:

$$\frac{d\hat{e}_x}{ds} = \hat{e}'_x = \frac{1}{\rho}\hat{e}_z \quad (7.3a)$$

$$\frac{d\hat{e}_z}{ds} = \hat{e}'_z = -\frac{1}{\rho}\hat{e}_x \quad (7.3b)$$

$$\frac{d\hat{e}_y}{ds} = \hat{e}'_y = 0 \quad (7.3c)$$

The location of q with respect to the centre of the orbit is described by the position vector:

$$\vec{R} = (\rho + x(s))\hat{e}_x + y(s)\hat{e}_y = r(s)\hat{e}_x + y(s)\hat{e}_y \quad (7.4)$$

Defining $r(s) = \rho + x(s)$ and assuming that $\rho' = \rho'' = 0$, the first and second derivatives of \vec{R} with respect to s are:

$$\vec{R}' = x'(s)\hat{e}_x + \frac{r(s)}{\rho}\hat{e}_z + y'(s)\hat{e}_y \quad (7.5)$$

$$\vec{R}'' = \left(x''(s) - \frac{r(s)}{\rho^2}\right)\hat{e}_x + y''(s)\hat{e}_y + \frac{2x'(s)}{\rho}\hat{e}_z \quad (7.6)$$

By the assumption that there is no coupling between transverse and longitudinal motion and that $R_z = R'_z = R''_z = 0$, the second derivative of \vec{R} is:

$$\vec{R}'' = \left(x''(s) - \frac{r(s)}{\rho^2}\right)\hat{e}_x + y''(s)\hat{e}_y \quad (7.7)$$

The transverse momentum of particle q can be described by the time derivative of the position vector \vec{R} :

$$\vec{p} = \gamma_r m_0 \frac{d\vec{R}}{dt} \quad (7.8)$$

where $\gamma_r m_0$ gives the mass of q . Thus the change in momentum from the Lorentz force exerted on the particle is:

$$e(\vec{v} \times \vec{B}) = \dot{\vec{p}} = \gamma_r m_0 \frac{d^2 \vec{R}}{dt^2} \quad (7.9)$$

Assuming that q is moving with a constant longitudinal velocity v_z and $v_z \gg |v_x|, |v_y|$. The relation between ds and dt for the particle q is:

$$ds = v_z \frac{r(s)}{\rho} dt \quad \Rightarrow \quad \frac{d}{dt} = v_z \frac{\rho}{r(s)} \frac{d}{ds} \quad (7.10)$$

With this relation, the right hand side of (7.9) can be written with s -dependence as

$$\gamma_r m_0 \ddot{\vec{R}} = \gamma_r m_0 \left(v_z \frac{\rho}{r(s)}\right)^2 \vec{R}'' = \gamma_r m_0 v_z^2 \frac{\rho^2}{r(s)^2} \left[\left(x''(s) - \frac{r(s)}{\rho^2}\right)\hat{e}_x + y''(s)\hat{e}_y \right] \quad (7.11)$$

Since $B_z = 0$:

$$e \left(\vec{v} \times \vec{B} \right) \approx ev_z (-B_y \hat{e}_x + B_x \hat{e}_y) \quad (7.12)$$

The magnetic field contributions from dipoles and quadrupoles is:

$$B_x = -\frac{p_0}{e} ky \quad (7.13a)$$

$$B_y = \frac{p_0}{e} \left(\frac{1}{\rho} - kx \right) \quad (7.13b)$$

Then left the hand side of (7.9) becomes:

$$ev_z (-B_y \hat{e}_x + B_x \hat{e}_y) = -v_z p_0 \left[\left(\frac{1}{\rho} - kx \right) \hat{e}_x + ky \hat{e}_y \right] \quad (7.14)$$

The longitudinal particle momentum $p = \gamma_r m_0 v_z$ is assumed to be a small deviation from the momentum of the reference particle $p = p_0 + \Delta p$. This condition is often well satisfied in accelerators, which allows for the approximation of $\frac{1}{p}$ by its first order Taylor expansion:

$$\frac{1}{p} = \frac{1}{p_0} \left(1 - \frac{\Delta p}{p_0} \right) \quad (7.15)$$

Thus (7.9) can be written as:

$$\left(x''(s) - \frac{r(s)^2}{\rho^2} \right) \hat{e}_x + y''(s) \hat{e}_y = - \left(1 - \frac{\delta p}{p_0} \right) \frac{r(s)^2}{\rho^2} \left[\left(\frac{1}{\rho} - kx \right) \hat{e}_x + ky \hat{e}_y \right] \quad (7.16)$$

The equation of transverse motion can then be separated for the two transverse planes:

$$x'' = \frac{x}{\rho} + \frac{1}{\rho} - \left(1 - \frac{\Delta p}{p_0} \right) \left(1 + \frac{x}{\rho} \right)^2 \left(\frac{1}{\rho} - kx \right) \quad (7.17a)$$

$$y'' = - \left(1 - \frac{\Delta p}{p_0} \right) \left(1 + \frac{x}{\rho} \right)^2 ky \quad (7.17b)$$

Neglecting all non-linear components leads to the linear transverse equations of motion:

$$x''(s) + \left(\frac{1}{\rho^2(s)} - k(s) \right) \cdot x(s) = \frac{1}{\rho(s)} \frac{\Delta p}{p} \quad (7.18a)$$

$$y''(s) + k(s) \cdot y(s) = 0 \quad (7.18b)$$

B Example of Simulation_parameter.input file

C Example of CO₂ Cloud file

```

# SIMULATION PARAMETERS
machine_param_file='machine_parameters.input'
secondary_emission_parameters_file =
    'secondary_emission_parameters.input'
beam_parameters_file='beam.beam'
logfile_path = 'logfile.txt'
progress_path = 'progress'
stopfile = 'stop'
# Additional clouds
additional_clouds_file_list = ['CO2.cloud', 'Xe.cloud']
cloud_name = 'H2'
Dt = 1.007e-9          #None
t_end = None
#import numpy as np
#dec_fact_out = int(np.round(5 * 25e-12/Dt))
lam_th = None         #e-/m
Dx_hist = 1e-3        #m
r_center = 1e-3       #m
Dt_En_hist = 25e-9   #s
Nbin_En_hist= 2000
En_hist_max= 2e3     #eV
t_ion=100.;          #s
N_mp_max=125000;     #size of allocated vectors
#Regen parameters
N_mp_regen=200000;
N_mp_regen_low=5000;
N_mp_after_regen=10000;
t_ON_regen_low=10.
fact_split=1.5;
fact_clean=1e-6;
regen_hist_cut = 1.e-4
N_mp_soft_regen = 30000
N_mp_after_soft_regen = 10000
nel_mp_ref_0 = None #redefine in the script
# Number of bins
Nx_regen=51;#it must be odd!
Ny_regen=51;#it must be odd!
Nvx_regen=51;#it must be odd!
Nvy_regen=101;#it must be odd!
Nvz_regen=51;#it must be odd!
#Sp_ch params
Dt_sc = 1.
Dh_sc = None
t_sc_ON=0e-9;          #s
#PyPICmode = 'FFT_OpenBoundary'
sparse_solver = 'klu'
flag_movie = 0          #1/0
flag_sc_movie = 0       #1/0
save_mp_state_time_file = -1

```

```
#CO2.cloud
cloud_charge = 1.602176565e-19
cloud_mass = 7.35953646e-26
# Residual gas ionization flag
gas_ion_flag = 1
# Photoemission flag
photoem_flag = 0
# Uniform initial distribution flag
init_unif_flag = 0
# Uniform initial density flag
init_unif_edens_flag = -1
# Secondary emission model
switch_model = 0
#Size of MP allocation
N_mp_max=125000;
#Regeneration parameters
N_mp_regen=250000;
N_mp_regen_low=5000;
N_mp_after_regen=10000;
t_ON_regen_low=10.
fact_split=1.5;
fact_clean=1e-6;
regen_hist_cut = 1.e-4
#Soft Regeneration parameters
N_mp_soft_regen = 75000
N_mp_after_soft_regen = 25000
# Residual gas ionization parameters
nel_mp_ref_0= 20. * 1.5 / 37.89 #e-/m
P_nTorr = 20.;
sigma_ion_MBarn = 1.5;
Temp_K = 300.;
E_init_ion = 6.90800000e-27
unif_frac = 0.
```

