DEVELOPMENT OF REDUCED-ORDER MODELS OF THE ION IMPACT DISTRIBUTION FUNCTION IN MAGNETIZED PLASMA SHEATHS

BY

MOHAMMAD MUSTAFA

THESIS

Submitted in partial fulfillment of the requirements
for the degree of Master of Science in Nuclear, Plasma & Radiological Engineering
in the The Grainger College of Engineering of the
University of Illinois Urbana-Champaign, 2023

Urbana, Illinois

Master's Committee:

Adviser: Professor Davide Curreli,
Second Reader: Professor Tomasz Kozlowski

Abstract

In magnetic-confinement fusion devices, high-fidelity models of the energy-angle distribution of the ions impacting on material walls are crucial for characterizing ion-surface interactions and impurity release. Typically, the Ion Energy-Angle Distributions (IEADs) are simulated using plasma kinetic models (e.g. Particle-In-Cell PIC codes, such as hPIC), which are usually computationally intensive. In this work, we constructed an effective surrogate model for the IEADs by means of a data-driven strategy in high-dimensional parameter space. The surrogate model considers up to four input parameters of relevance to the problem: (1) electron-to-ion temperature ratio, (2) magnetic field inclination, (3) magnetic field strength, and (4) plasma density. The hPIC2 code was utilized to generate necessary training and testing data sets. A sparse grid was employed to reduce the cost of the surrogate model construction without compromising accuracy. Least square approximation using data from a denser sparse grid was utilized to mitigate the effect of particle noise. Fitting of IEAD was performed in a transformed coordinate system of the distribution. Additionally, an artificial neural network strategy based on zero-inflated models was employed to construct a surrogate model of plasma sheath potentials. The surrogate models constructed as part of this work provide computationally efficient tools to emulate the output of hPIC with limited errors. A variance-based sensitivity analysis using samples drawn from the surrogate models was performed. The sensitivity analysis of plasma potentials showed that the floating wall potential is solely affected by the electron-to-ion temperature ratio, whereas the potential drop across the magnetic presheath and Debye sheath were significantly affected by the magnetic field inclination angle. The sensitivity analysis of the IEAD moments also showed a strong dependency of ions' impact energy on the electron-to-ion temperature ratio with insignificant dependence on the other physical parameters. On the other hand, the analysis revealed significant dependencies of ions' angle of impact on electron-to-ion temperature ratio and magnetic field inclination angle, with a much lower effect of the remaining parameters.

Acknowledgments

I would like to express my sincere gratitude to my supervisor, Dr. Davide Curreli, for his invaluable guidance, support, and encouragement throughout my research. Dr. Curreli's insights and feedback were instrumental in shaping my research and ensuring its success. His vast knowledge and expertise helped me navigate complex issues and overcome challenges. I deeply appreciate the time and energy that he devoted to helping me achieve my academic goals. Without his unwavering support and guidance, this research would not have been possible.

I would like to express my gratitude to my second reader, Dr. Tomasz Kozlowski, for providing insightful comments and suggestions that helped me to refine my ideas and improve the quality of my work. Dr. Kozlowski"s feedback and expertise were invaluable in ensuring the success of this research.

I am deeply grateful to Dr. Pablo Seleson and Dr. Cory Hauck, who provided valuable insights and expertise throughout the course of this research. They generously shared [his/her/their] time and expertise, providing critical feedback that helped me to improve the quality of my work. Their contributions were invaluable in advancing my research, and I could not have accomplished this without their support. I am fortunate to have had the opportunity to collaborate with someone as knowledgeable and supportive as Dr. Pablo Seleson and Dr. Cory Hauck, and I am truly grateful for their contributions.

I would like to express my gratitude to my colleagues for their support, constructive criticism, and stimulating discussions throughout my research. Their positive attitude and camaraderie made the long hours in the lab more enjoyable and productive. I feel fortunate to have worked with such a wonderful group of colleagues and will always cherish the memories of our time together.

I am also thankful to my family and friends for their unwavering support and encouragement throughout my academic journey. Their love and encouragement have been a source of strength and inspiration.

Thank you all for your support and guidance. This research would not have been possible without your help.

Table of contents

List of Abbreviations

IEAD	Ion Energy Angle Distribution
DS	Debye Sheath
CS	Chodura Sheath
MPS	Magnetic Pre-Sheath
PSI	Plasma Surface Interface
OAT	One At a time
SA	Sensitivity analysis
LSA	Local Sensitivity analysis
GSA	Global Sensitivity analysis
SG	Sparse Grid
TG	Tensor Grid
UQ	Uncertainty quantification
PIC	Particle In Cell
PPC	Particles Per Cell
RoI	Response of Interest
MC	Monte-Carlo
SM	Surrogate Model
WEST	tungsten "W" Environment in Steady-state Tokamak
ANN	Artificial Neural Network
SVC	Support Vector Classifier
LSF	Least Square Fitting
PFC	Plasma Facing Component

List of Symbols

K_B	Boltzmann's Constant
ε_0	Permittivity of Free Space
e	Electron's Charge
ϕ	Electric Potential
T_i	Ion's Temperature
T_e	Electron's Temperature
v_{th}	Thermal Speed
m_i	Ion Mass
m_e	Electron Mass
λ_D	Debye Length
C_s	ions' Acoustic Speed
n_i	ith Particles' Density
n_0	Plasma Density
B	Magnetic Field Strength
ψ	Magnetic Field Inclination with Respect to Surface Normal
E	Electrical field
$\dot{r}$	Larmor Radius
k	Wave Number
ω	Wave Frequency
ω_p	Plasma Frequency
ω_c	Cyclotron Frequency

θ	Ions' Impact Angle
ϕ_w	Floating Wall Potential
ϕ_{DS}	Potential Drop Across Debye sheath
ϕ_{MPS}	Potential Drop Across Magnetic Pre-Sheath

Chapter 1

Introduction

1.1 Motivation

The recent advancement in magnetically confined fusion devices enabled an increase in discharge time and plasma temperatures. The harsh environment generated by thermonuclear plasmas in contact with material surfaces underscores the importance of studying Plasma-Material Interactions **PMI**. Such interactions have a significant impact on the fusion device's performance and safety. The main impacts of the **PMI** on fusion devices operations include material erosion, tritium retention, and plasma heating. Material erosion can lead to a release of high charge impurities, which rapidly cools the plasma, negatively affecting plasma confinement and damaging plasma facing components. Retention of tritium, a radioactive isotope in the surface of plasma facing component can impose a hazard and reduce the amount of fuel available for fusion reaction. Plasma heating can impose thermal stresses on the **PFC** and affect its durability [1], [2]. Understanding and controlling plasma-material interactions is critical for the successful operation of tokamaks and the development of practical fusion energy. These interactions are dominated by ion-material interactions, including reflection, sputtering, implantation, etc. While an estimate of such interactions can be computed using the mean energies and angles of the impacting ions (e.g., energies provided by fluid models), such interactions are a strongly non-linear function of both ions' energies and angles [3], consequently, detailed knowledge of Ion Energy-Angle Distributions **IEAD**s at the **PSI** is often necessary.

1.2 Role of ion impact distributions on material walls

While plasma fluid models are able to capture the general features of the boundary plasma, they usually lack the ability to provide a detailed Ion Energy-Angle Distribution of the particles impact-

ing the material wall. Thus, kinetic simulation models are usually preferred. However, kinetic models are significantly more computationally expensive when compared to the fluid model. Accordingly, seeking a way to efficiently construct a reliable surrogate model that emulates the **IEAD** produced by a high-fidelity kinetic code (e.g., hPIC [4], [5]) is desirable. The primary goals of this work are to provide a methodology for efficient construction of a data-driven reduced order model in the high-dimensional space of the **IEAD**. To reveal the underlying physics impacting the **IEAD** by performing sensitivity analysis on the **IEAD**s' moments. To gain a better understanding of the effect of the sheath structure on the **IEAD** by constructing a surrogate model for the plasma sheath structure and utilizing it for sensitivity analysis. And to explore the possible extension of the model's dimensionality by performing local sensitivity analysis on multi-species fusion plasmas.

1.3 Construction of Reduced-Order Models

The construction of the **IEAD** reduced order model employed a Least Square Approximation **LSA** technique for the bin-by-bin fitting performed on the transformed coordinates. In addition, a sparse grid was utilized to overcome the high computational cost. On the other hand, in addition to the **LSA** technique, a machine learning based technique, optimized for exes-zero problems was used for the plasma sheath potentials surrogate model construction. Training and testing data sets were generated using hPIC simulations. Raw simulation's outputs of interest are lists of ions' velocity vectors at the boundary, plasma potential profile, zeroth and first fluid moments. The post-processes then convert the velocity vectors into **IEAD**s, identifies the location of sheath entrances using classical plasma sheath theories, and compute the potential drop across the various layers of plasma sheath. Finally, Sobol's indices were used to examine the sensitivity of several responses of interests to the input parameters.

1.4 Thesis structure

The structure of the thesis is as follows: Chapter 2 provides a general overview of sheath formation and its effect on the ions distribution function inside the plasma and subsequently the **IEAD**, for the cases of an unmagnetized sheath in 2.1, magnetized sheath 2.2, and demonstrates the need for a fully kinetic model for **IEAD** computations. Chapter 3 introduced the data-driven surrogate model construction of the **IEAD** generated by hPIC. Where the mathematical tools utilized in the construction as well as the construction procedures were summarized in section 3.1, section 3.2 summarized the training data ranges and simulation setup, and section 3.3 assess the accuracy of the trained model and its predictability. Chapter 4 constructed a surrogate model for plasma

potentials, where problem setup was reviewed in 4.1, least square approximation was utilized for the construction in 4.2, and an artificial neural network was used in 4.3. Chapter 5 performs global and local sensitivity analysis on both the **IEAD**, and plasma potentials. The sensitivity analysis methods used in this work are reviewed in 5.2, the global sensitivity analysis is reported in 5.3.1, and the local analysis can be found in 5.3.2. Finally, chapter 6.1 is the conclusion chapter.

Chapter 2

Plasma Sheath Formation and Structure

2.1 Unmagnetized Plasma Sheath

Sheath formation on plasma wall interface is one of the oldest problems in plasma physics yet still not fully understood. Since the introduction of the basic concept of sheaths 1929 [6], several attempts were made to develop a mathematical understanding of the phenomena, most notably the classical Bohm criterion [7]. The basic idea of plasma sheath can be summarized by noting that the average thermal speed of a charged particle v_{th} is proportional to the temperature of the particles' population and inversely proportional to its mass (i.e., $v_{th} = \left(\frac{K_B T}{m}\right)$). Accordingly, lighter species (i.e., electrons) are expected to leave the bulk plasma toward the wall at a higher rate than the positively charged heavier ions. The wall will adjust itself to be negative consequently accelerating the positive particle toward the wall and repelling the electrons leading to a floating net positive space charge known as plasma sheath, also known as Debye sheath (DS), the associated potential known as floating wall potential. Usually, the sheath will have a length in the order of Debye length λ_D.

The classical picture of plasma bulk and sheath is based on the classical Bohm criterion and can be found in many sources in literature, most notably [8], [9], and [10]. The classical picture of plasma boundaries consists of three main regions, the quasi-neutral bulk region that makes most of the plasma domain, followed by a pre-sheath region which is necessary to accelerate ions to sound speed C_s (which is necessary to fulfill Bomh's criterion that will be discussed shortly) and finally a sheath region. According to [11] the "pre-sheath" can have several meanings in literature. For example, it can refer to the thin layer adjacent to the sheath and required to accelerate ions, this definition of pre-sheath is widely used in literature as noted earlier, or it can refer to the whole domain of the collisionless plasma.

In order to highlight the meaning of Bohm's criterion, the review provided by [11] will be summarized. In his approach, Bohm used a fluid model and assumed the following: i. Cold ions

($T_i = 0$), ii. Monoenergetic ions, iii. Boltzmann electron, given by Equation 2.1, iv. Unmagnetized plasma ($B = 0$), v. Single species, single charge ions, vi. Quasi-neutral bulk plasma. Among Bohm's assumptions, i,ii and vi are the roughest assumptions.

$$n_e = n_0 \exp\left(\frac{e\phi}{K_B T_e}\right) \tag{2.1}$$

Using plasma's fluid model and introducing dimensionless quantities shown in 2.2. Then the sheath can be described by; dimensionless ion continuity Equation 2.3, dimensionless conservation of energy 2.4, and dimensionless Boltzmann electrons 2.5.

$$\begin{aligned} \chi &= -\frac{e\phi}{K_B T_e} \\ \mathcal{M} &= \frac{1/2 m_i v_i^2}{K_B T_e} \\ \xi &= z/\lambda_D \\ N_{e,i} &= \frac{n_{e,i}}{n_0} \end{aligned} \tag{2.2}$$

$$N_i \mathcal{M}^{1/2} = \mathcal{M}_0^{1/2} \tag{2.3}$$

$$\mathcal{M} = \mathcal{M}_0 + \chi \tag{2.4}$$

$$N_e = \exp(-\chi) \tag{2.5}$$

Using Poisson's Equation 2.6 to relate the densities in the sheath to the electrical field. Which modifies using the dimensionless quantities and 1D into Equation 2.7.

$$\nabla^2 \phi = -\frac{\rho}{\varepsilon_0} = e\frac{n_e - n_i}{\varepsilon_0} \tag{2.6}$$

$$\frac{d^2\chi}{d\xi^2} = \left(1 + \frac{\chi}{\mathcal{M}_0}\right)^{-1/2} - \exp(-\chi) \tag{2.7}$$

The boundary conditions proposed by Bohm are shown in Equation 2.8, meaning the potential and field vanishes at a point remote from the sheath edge (i.e., quasi-neutral).

$$\chi|_{\xi\to 0}, \frac{d\chi}{d\xi}|_{\xi\to 0} \to 0 \tag{2.8}$$

Multiplying Equation 2.7 and integrating using the boundary conditions given by 2.8 leading to

$$\left(\frac{d\chi}{d\xi}\right)^2 = 4\mathscr{M}\left[\left(1+\frac{\chi}{\mathscr{M}_0}\right)^{-1/2} - 1\right] + 2(\exp(-\chi) - 1) \tag{2.9}$$

Using Taylor expansion leads to

$$\left(\frac{d\chi}{d\xi}\right)^2 = \left(1 - \frac{1}{2\mathscr{M}}\right)\chi^2 + \ldots \tag{2.10}$$

From 2.10 it can be noted that the first term on the RHS has to be greater or equal to zero. This is called Bohm's criterion and can be written in the form of Equation 2.11, where C_s is the ions' acoustic speed given by Equation 2.12 and Z is the charge number of the specie.

$$v_{i,0}^2 \geq C_s^2 \tag{2.11}$$

$$C_s = \left(\frac{ZK_BT_e}{m_i}\right)^{1/2} \tag{2.12}$$

The total potential drop across plasma boundary and the bulk is given by the floating wall potential shown in Equation 2.13 [10].

$$\frac{e\phi_w}{K_BT_e} = -\frac{1}{2}ln\left(\frac{m_i/m_e}{2\pi(1+\frac{T_i}{T_e})}\right) \tag{2.13}$$

Time and spatial evolution of particle's distribution in a collisionless plasma are commonly described using Vlasov's equation [12] shown in 2.14, where $f(\mathbf{r},\mathbf{v},t)$ is the particles' distribution function in the 7 dimensional space of position, velocities, and time; $\mathbf{E}$ and $\mathbf{B}$ are the electrical and magnetic fields; and $\mathbf{v}$ is the velocity vector. Electrical and magnetic fields are commonly evaluated using Maxwell's equations in the differential form, coupled with Vlasov equation it forms Vlasov-Maxwell system of equations. In the absence of a magnetic field, the Vlasov-Poisson system of ordinary differential equations formed by 2.14 & 2.15 can be used instead, such that ρ is the charge density.

$$\frac{\partial f}{\partial t} + \mathbf{v}\cdot\nabla f + \frac{q}{m}(\mathbf{E}+\mathbf{v}\times\mathbf{B})\cdot\frac{\partial f}{\partial \mathbf{v}} = 0 \tag{2.14}$$

$$\nabla^2\phi = -\frac{\rho}{\epsilon_0} \tag{2.15}$$

For time-independent ion-electron plasma, the Vlasov-Poisson system of equation in one dimension can be written in form of Equation 2.16, where ϵ_0 is the permittivity of free space. The resulting equation is an integro-differential equation that is challenging to solve. Further

simplification can be made by assuming Boltzmann electrons (i.e., $n_e = n_0 \exp\left(\frac{e\phi}{K_B T_e}\right)$) and plasma is quasi-neutral (i.e., $n_e \approx n_i$). Such assumptions lead to a non-linear ordinary differential equation given by 2.17. And obtaining an analytical solution for such a problem is still challenging despite simplifications.

$$-\frac{d^2\phi}{dx^2} = \frac{1}{\epsilon_0}\int q_i f_i(x,v)\,dv_i - \frac{q_e}{\epsilon_0} n_e \tag{2.16}$$

$$-\frac{d^2\phi}{dx^2} = \frac{q_i n_0}{\epsilon_0} - \frac{q_e n_0}{\epsilon_0}\exp\left(\frac{e\phi}{K_B T_e}\right) \tag{2.17}$$

Typically, plasma is in isothermal equilibrium and follows the Maxwell-Boltzmann distribution given by Equation 2.18. When an external force acts on the plasma and introduces a drift velocity $\mathbf{v}_0$ the distribution modifies to drifting Maxwellian distribution given by Equation 2.19, the plasma is still in isotropic equilibrium but the distribution is drifted as a whole, this can be accounted for by introducing Galilean transformation. Although drifting Maxwellian can be viewed as anisotropic, this is not the case. It is in fact a Maxwellian in which its mean is shifted by $\mathbf{v}_0$. If the drift is resulting from the plasma sheath, then the beam distribution occurs when the speed gained by the electrical field ($v_0 \propto \sqrt{2q\phi/m_i}$) is much larger than the ion's thermal speed. That requires the floating wall potential given by Equation 2.13 to be high, which can happen when $T_e/T_i >> 1$. schematics of the distributions are shown in Figure 2.1.

$$f(\mathbf{v})d^3\mathbf{v} = \left(\frac{m_i}{2\pi K_B T_i}\right)^{3/2} exp\left(-\frac{m\mathbf{v}^2}{2K_B T_i}\right) d^3\mathbf{v} \tag{2.18}$$

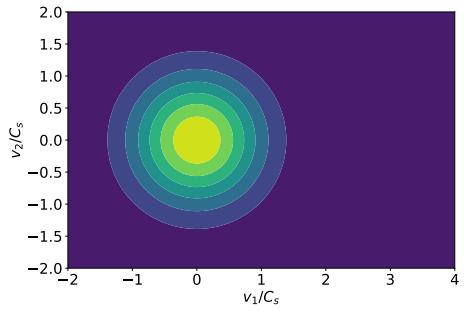
$$f(\mathbf{v})d^3\mathbf{v} = \left(\frac{m_i}{2\pi K_B T_i}\right)^{3/2} exp\left(-\frac{m(\mathbf{v}-\mathbf{v}_0)^2}{2K_B T_i}\right) d^3\mathbf{v} \tag{2.19}$$

Ions' angular distribution of the Maxwell-Boltzmann distribution for an unmagnetized plasma is given by Equation 2.20 [13], where v_t is the thermal mean velocity, v_0 is ions' drift velocity resulting from plasma sheath electrical field, such that, $v_0 = -2e\phi/m_i$.

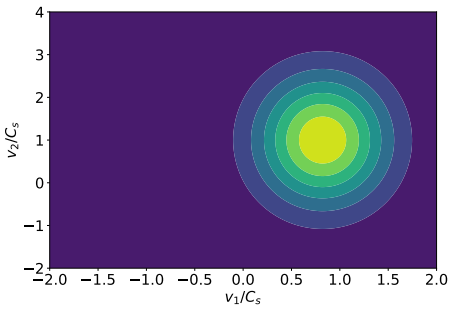
$$f(\theta) = \frac{v_0^2}{v_t^2}\frac{\tan\theta}{\cos^2\theta}\exp\left(\frac{v_0^2}{2v_t^2}\right) \tag{2.20}$$

2.2 Magnetized Plasma Sheath

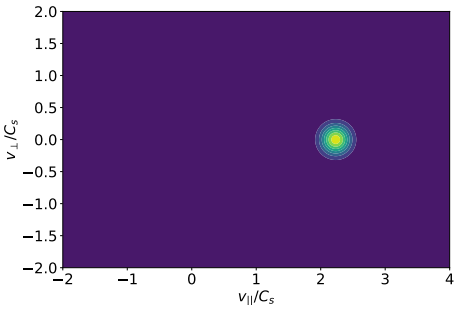
In magnetized plasma, with magnetic field B inclined with ψ degrees with respect to the surface normal, Equation 2.13 is still applicable and describes the total potential drop between the bulk plasma and sheath-wall interface. However, the sheath and pre-sheath structures are different.



a) Maxwellian Distribution



b) Drifting Maxwellian Distribution



c) Ions Beam Distribution

Figure 2.1: Variations of unmagnetized Maxwell-Boltzmann distribution, showing the drifting Maxwell-Boltzmann distribution in c).

The pre-sheath region still accelerates the ions to sound speed with a direction parallel to the magnetic field up to the entrance of another region known as the magnetic pre-sheath MPS or Chodura sheath [14]. In the MPS region, the ions accelerate from ion sound speed C_s parallel to the magnetic field to ion sound speed parallel to surface normal. The potential drop across Chodura's sheath is shown in Equation 2.21. The total potential drop given in Equation 2.13 does include the drop in MPS in addition to the classical sheath drop, also known as the "Debye sheath" DS as in Equation 2.22. As noted by [15], there is a critical magnetic field inclination angle in which ϕ_{MPS} exceeds the total potential drop as can be noted from Equation 2.22. While this is true, it reveals a limitation in Chodura's model rather than a unique physical phenomenon at such high angles.

$$\frac{e\phi_{MPS}}{K_B T_e} = ln(cos(\psi)) \tag{2.21}$$

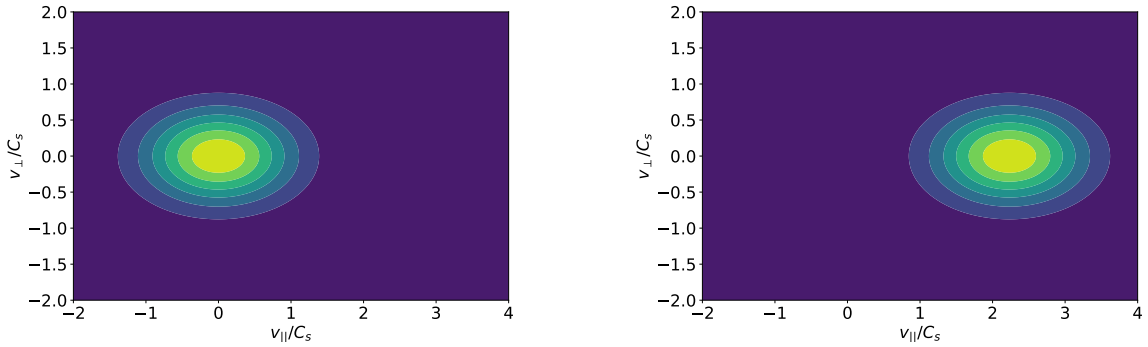
$$\phi_w = \phi_{DS} + \phi_{MPS} \tag{2.22}$$

While obtaining an analytical solution for Vlasov-Poisson system of equations describing unmagnetized plasma is challenging as discussed in section 2.1, more sophisticated analysis is required when dealing with magnetized plasma. Nevertheless, several analytical solutions are available in the literature, most notably the work done by [16], [17].

In the case of magnetized plasma, the distribution is axisymmetric and best described by bi-Maxwellian distribution as shown in Equation 2.23. The main idea is to have a temperature parallel to the magnetic field different than the temperature perpendicular to the field [18]. This is especially a common case for ICRF and NBI heating schemes.

$$f(\mathbf{v}_{||}, \mathbf{v}_{\perp}) d v_{||} d v_{\perp} = \left(\frac{m_i}{2\pi K_B T_{||}}\right)^{1/2} \left(\frac{m_i}{2\pi K_B T_{\perp}}\right) exp\left(-\frac{m(\mathbf{v}_{||} - \mathbf{v}_0)^2}{2K_B T_{||}} - \frac{\mathbf{v}_{\perp}}{2K_B T_{\perp}}\right) \tag{2.23}$$

Equation 2.23 accounts for the drift component by adding a drift component to the parallel direction. In a real case scenario, the drift component is resulting from the plasma sheath, which is directed toward the surface normal direction, thus the shift is expected to affect both the parallel and perpendicular velocity components. In addition, another drift component resulting from the $E \times B$ drift is expected to affect the perpendicular component. It should be noted that the drifting Maxwellian distribution is different from the ion beam distribution, the latter can be described as a Maxwellian distribution with a large shift and a small temperature (i.e., the drifting component is much larger than the thermal speed of the ions). A schematic of particles' distributions under magnetic constraints is shown in Figure 2.2. A review of Maxwell-Boltzmann distribution formulations commonly found in plasma is reviewed in [18].



a) Bi-Maxwellian Distribution

b) Drifting Bi-Maxwellian Distribution

Figure 2.2: Variations of magnetized Maxwell-Boltzmann distribution, showing the Bi-Maxwellian distribution in a), and drifting Bi-Maxwellian in b).

2.3 Fully Kinetic Particle-In-Cell (PIC) Model.

Analytical models of plasma sheath often fail to capture the exact particles' distributions at the Plasma Surface Interface **PSI**. Knowledge of such distributions is often required for ion-materials interactions at **PSI**. Accordingly, high-fidelity simulations using the Particle-In-Cell technique are commonly utilized. **PIC** simulations commonly track individual particles in continuous phase space, whereas distributions' moments are computed on stationary mesh points. **PIC** technique solves the Vlasov-Poisson of plasma's distribution function given by equations 2.14 & 2.15 using Boris' leapfrog method [19], [20], where particle push is split into two steps, particle acceleration due to electrical field and particle rotation due to the magnetic field. This step is denoted as particle push. Followed by resolving the field by solving Poisson's equation, where finite element, finite difference, and spectral methods can be used. Then particle and field weights of the shape functions are computed. This process is repeated over several time steps. Such cycle is denoted as "classical" **PIC** and is summarized in Figure 2.4 b).

Because of its credibility, **PIC** method was utilized for computing **IEAD** and plasma moments required for the analysis in chapters 3-5. **hPIC** code [4], [5] was used for such a purpose. A summary of **hPIC** code can be found in 2.3.1. **hPIC** computes the **IEAD** by storing the velocity coordinates $\mathbf{V} = (V_x, V_y, Vz)$ for ions crossing the boundaries of the domain. Then ion's energy and angle of impact for each ion are computed according to equations 2.24 and 2.25, respectively. An example of **IEAD** generated using **hPIC** simulation is shown in Figure 2.3.

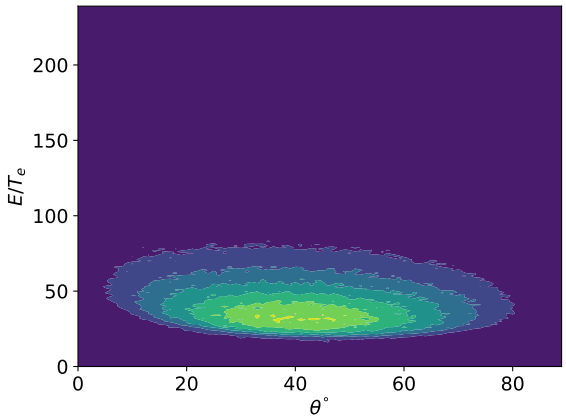


Figure 2.3: Example of Ion Energy-Angle Distribution **IEAD** generated using **hPIC** simulation with electron and ion temperatures $T_e = T_i = 10\,[eV]$, magnetic field strength $B = 8.0\,[T]$, and magnetic field inclination angle $\psi = 46°$.

$$E = \frac{1}{2} m_i \mathbf{v}^2 \tag{2.24}$$

$$\tan(\theta) = \frac{\left(V_y^2 + V_z^2\right)^{1/2}}{|V_x|} \tag{2.25}$$

2.3.1 Overview of hPIC

PIC is a fully kinetic, electrostatic Particle-In-Cell (PIC) developed for near-surface, plasma materials interactions for both magnetized and unmagnetized plasmas [4]. Similar to the classical **PIC**, hPIC solves the Vlasov–Poisson equation of plasma for distribution functions in phase space. The individual particles' velocity is updated using the equation of motion (Boris' leapfrog method). This step is combined with particle weighting, particle interpolation, and particle localization inside the mesh into one step, the particle advance. Particles post-process handles particles at the boundary and allows for particle communication. Finally, fields are updated using Poisson's equation. A schematic summarize hPIC cycle is shown in Figure 2.4 a).

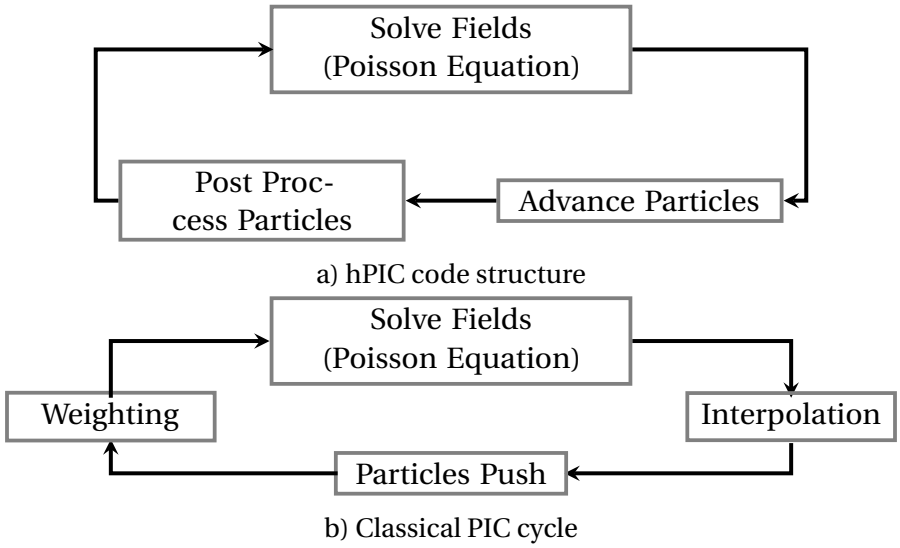


Figure 2.4: Schematics of the classical Particle-In-Cell cycle and hPIC cycle. Reproduced from [4].

Chapter 3

Data-Driven Surrogate Model of the Ion Energy-Angle Distributions (IEAD) produced by hPIC

In this chapter, a method for reconstruction of multidimensional particles distribution obtained from Particle in Cell **PIC** simulations over high-dimensional input parameters' space is presented and demonstrated. A data-driven surrogate model that maps inputs-outputs relationship at a significantly reduced computational cost was constructed. In order to maximize accuracy at a reduced number of simulations, **PIC** samples drawn on a Sparse Grid **SG** were collected [21]. Such samples form a training set that is used to approximate the distribution function in the energy-angle phase space locally. The approximation takes a form of a polynomial function of the input parameters using a Leas Square Fit **LSF**. Once constructed, the surrogate model is a handy tool that can be used when a large number of samples is needed or when computational resources are limited, for example, Uncertainty Quantification **UQ** and Sensitivity Analysis **SA**.

The method was applied to the problem of characterizing the Ion Energy Angle Distributions **IEAD**s [13] of ions impacting the Tokamak divertor. Such a problem is important to characterize Plasma-Material Interactions **PMI** in magnetically confined fusion devices. The training set (i.e., **IEAD**s) was produced from a large number of simulations produced using **hPIC** [4] within data ranges relevant to Tokamak (see Table 2.1.1 in [22]). More information about **hPIC** is reviewed in chapter 2.3.1. **PMI** processes at the Tokamak divertor are dominated by ion-material interactions and often lead to; impurity sputtering [23], ion implantation [24], particle reflection, and more [1], [25]. While some empirical and semi-empirical formulas provide an approximation for such process (e.g., [26], [27]). Such processes are strongly nonlinear as a function of ions' energy and angle, thus, accurate knowledge of **IEAD** is essential.

A surrogate model is a method used when the Response of Interest **RoI** cannot be easily mea-

sured or computed, so the model of the outcomes is used instead. A data-driven surrogate model [28], [29] utilizes data from an experiment or a high-fidelity simulation to construct a map of the response surface between input parameters and **RoI**s. The goal of a surrogate model is to mimic the behavior of the computational model as accurately as possible while being significantly cheaper to evaluate. Popular surrogate modeling approaches include; Galerkin methods, for example, polynomial chaos expansion **PCE**; radial basis functions **RBF**; Gaussian processes; Support Vector Machines (also, Support Vector Networks) **SVM**s; and Bayesian network; and Artificial Neural Network **ANN**. Despite being developed by different comminutes, these methods share many common features. Galerkin methods construct the approximation in the linear space with predefined basis functions, such as global polynomials [30], expansion coefficients then chosen to match data using projection. The limitation of Galerkin is that the **RoI**s should be evaluated at specific input parameters and cannot fit random points. Radial Basis Function **RBF** interpolation [31] also represents the interpolating functions as a linear combination of basis functions, but the basis functions only depend on the distance between the prediction and training points (i.e., decay radially away from the training point), making **RBF** a mesh-free method suitable for unstructured training data sets. However, **RBF** basis functions have a tunable hyper-parameter that requires calibration to match the expected behavior of the data (e.g., For Gaussian basis functions the hyper-parameter is the variance σ^2) [32]. Gaussian processes regression (also known as Kriging) [33] is an interpolation method based on Gaussian process governed by prior covariances, the latter is usually specified by two points kernel function, such kernel function commonly represented in the form of **RBF**. **SVM**s are considered a sub-class of machine learning methods, it is commonly used for both classification and regression [34]–[37]. Advantages of **SVM** include: effectiveness in high-dimensional space, even when the number of dimensions is larger than the number of samples; it is memory-efficient; and it is considered to be a versatile method. **SVM**s usually use linear approximation with coefficients determined by a constrained quadrature optimization problem. However, it can be used for non-linear approximations using a non-linear basis function as a lifting function using a two-points kernel function (e.g., **RBF**). **ANN** maps the inputs with outputs via a complex network of nodes forming a set of layers. The coefficients of **ANN** are calculated via non-linear regression to minimize what is known as the loss function. Such networks have the ability to capture non-linear responses, especially with high-dimensional spaces, but are often expensive to train and require some trial-and-error to find the best **ANN** architecture.

The organization of this chapter is as follows. The methodologies for surrogate model construction are presented in section 3.1, including the sparse grid in section 3.1.1, the least square approximation in section 3.1.2, and the workflow for **IEAD** surrogate model construction is summarized in section 3.1.3. Simulations setup, including ranges of input data and simulations'

tools, are summarized in section 3.2. And results for **IEAD** surrogate model training and testing procedures are reported in section 3.3.

3.1 Methodologies

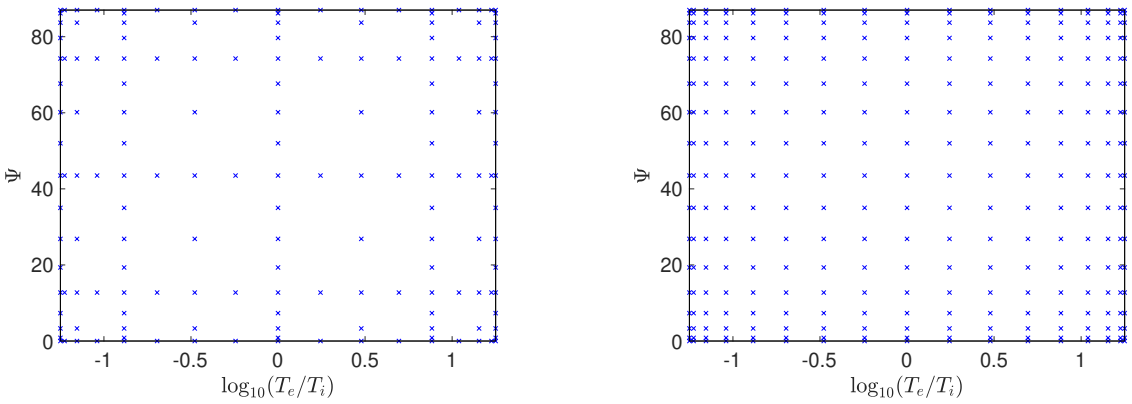
This section provides an overview of the tools used for **IEAD** surrogate model construction, including sparse grid used for efficient sampling of the training set, least square approximation utilized to reduce the effect of particles' noise, and coordinate transformation in ions' energy and angle space.

3.1.1 Sparse Grid

The standard way of representing multi-dimensional polynomial functions is tensor grid shown in Figure 3.1 b). In such a grid, the number of points (i.e., inputs) in which the **RoI**s have to be computed depends exponentially on the number of dimensions (i.e., number of input parameters). However, Particle In Cell **PIC** simulations are often computationally expensive, which motivates seeking a fitting method to adequately capture the behavior of **RoI**s using a reduced number of simulations without compromising accuracy. This can be achieved by utilizing Sparse Grid **SG** methods [21]. Such methods are built on classical interpolation theory and extended to handle multi-dimensions [38]. For a function $y(\mathbf{p})$ defined on a hypercube with m dimensions (e.g., $y(\mathbf{p}) : \Gamma \subset \mathbb{R}^m \rightarrow \mathbf{R}$ where $\Gamma = [-1,1]^m$), we seek an approximation function $f(\mathbf{p})$ as given by Equation 3.1, where, $\phi_l(\mathbf{p})$ and c_l are a set of properly chosen functions and its corresponding coefficients, respectively. The main goal of the sparse grid method is to obtain maximum accuracy of fitting with a reasonable computational cost.

$$y(\mathbf{p}) \approx f(\mathbf{p}) = \sum_{l=1}^{L} c_l \phi_l(\mathbf{p}) \tag{3.1}$$

The basis functions of choice for this work are Lagrange polynomials with inputs sampled from points on Clenshaw-Curtis **SG**. Such selection of basis functions and points allows for nesting, meaning that when increasing the total polynomial level, points from previous level(s) can be reused, allowing for a significantly reduce the number of simulations needed when higher accuracy is desired. The total polynomial levels used in the current work are 7, 10, and 13. Figure 3.1 shows a comparison of sparse grid and tensor grid in 2D inputs in data ranges adopted in this work. The reduction in the number of simulations required to construct the surrogate model when utilizing a sparse grid is summarized in Table 3.1 for the 2D & 4D hypercubes. The library (Toolkit for Adaptive Stochastic Modeling and Non-Intrusive ApproximatioN) **TASMANIAN** [38]



a) Sparse Grid b) Tensor Grid

Figure 3.1: Comparison of Sparse Grid Vs. Tensor Grid for the 2D case with a total polynomial level of 13.

Table 3.1: Comparison of the number of points in Clenshaw-Curtis sparse grid and tensor grid for the 2D and 4D cases at total polynomial levels of **P** = 7, 10 & 13.

Level	Number of points (SG)	Number of points (TG)	reduction in number of points
	2D Cases		
7	49	81	39.5%
10	129	289	55.4%
13	161	289	44.3%
	4D Cases		
7	721	6561	89.0%
10	2225	83521	97.3%
13	5563	83521	93.3%

was used for sparse grid implementation and interpolation.

3.1.2 Least Square Approximation

PIC simulations often suffer from particle noise due to the stochastic sampling of ions' speeds and positions during simulations. Consequently, the resulting **RoI**s are usually non-smooth. Global interpolation strategies can significantly amplify the noise. Accordingly, incorporating a strategy to mitigate the effect of noise when using the sparse grid method construction is essential. One straightforward method to reduce the noise in the outputs is to average out the values of **RoI**s using the data from a large number of simulations or using refined simulations by increasing the number of particles per cell and the number of cells. However, **PIC** simulations are computationally expensive, which will render either method inefficient. In addition, **hPIC**

used in the current work lacks parallelization capabilities, which imposes a limit on the level of refinement that can be used for numerical parameters. The first approach was used within reasonable computational cost, where each simulation was repeated a number of times such that $N_{PPC} \times N_{sims}$ is constant. While this strategy mitigated the effect of particle noise, it was not sufficient, and seeking additional method(s) was/were essential.

The next strategy is an extension to the sparse grid approach (section. 3.1.1). By computing samples on a denser sparse grid and computing polynomial basis functions coefficients using least square fitting **LSF**. The goal is to construct an approximation in the form given by Equation 3.1 for a given set of basis functions $\{\phi_l(\mathbf{p})\}_{l=1}^{L}$. In other words, for a given set of K data points $\{\mathbf{p}_k, y_k\}_{k=1}^{K}$, we are seeking coefficients that minimize the sum of square residuals R^2 given by Equation 3.2. Where $\mathbf{p}_k$ are the parameters (inputs) used in the k^{th} sample and y_k is the corresponding scalar **RoI**, and $\{c_l\}_{l=1}^{L}$ are the coefficients.

$$R^2 = \sum_{k=1}^{K} \left[f(\mathbf{p}_k) - y_k \right]^2 = \sum_{k=1}^{K} \left[\sum_{l=1}^{L} a_1 \phi_1(\mathbf{p}_k) - y_k \right]^2 \tag{3.2}$$

In matrix form Equation 3.2 can be written as,

$$\mathbf{c} = \underset{a \in \mathbb{R}^L}{argmin} ||\mathbf{A}\mathbf{a} - \mathbf{y}||_2^2 \tag{3.3}$$

where $\mathbf{c}$ and $\mathbf{a}$ are vectors of coefficients $(c_1, c_2, \ldots, c_L)$ and $(a_1, a_2, \ldots, a_L)$, respectively, $\mathbf{A}$ entries are $A_{k,l} = \phi_l(\mathbf{p}_k)$, and $||.||_2$ is the L_2 norm. For Lagrange polynomials, when $K = L$ the matrix A is the identity matrix and the solution is trivial (i.e., the solution of the global interpolation strategy in Equation 3.1). However, when $K > L$ a QR solver is employed and the solution is no longer trivial.

The main goal of using an over-sampled spare grid approach is to mitigate particle noise with a reasonable number of simulations. Using **LSF** with random samples often requires $K >> L$ and the computational cost can be prohibitively expensive. While the current approach requires $K > L$.

3.1.3 Construction of IEAD Surrogate Model

The general workflow of surrogate model construction and **SA** procedures is summarized in Figure 3.2. And shows 4 blocks of sub-workflow including: data generation, **IEAD**/**IEAD*** surrogate model, moments surrogate model, and the **SA** procedures model. The data generation sub-workflow creates the **SG** for a given total polynomial level P using TASMANIAN and scans the archive for simulations from previous simulations then initiate simulations for new input vectors and collects the ions' velocity data (V_x, V_y, V_z) for each simulation. Velocities data is then

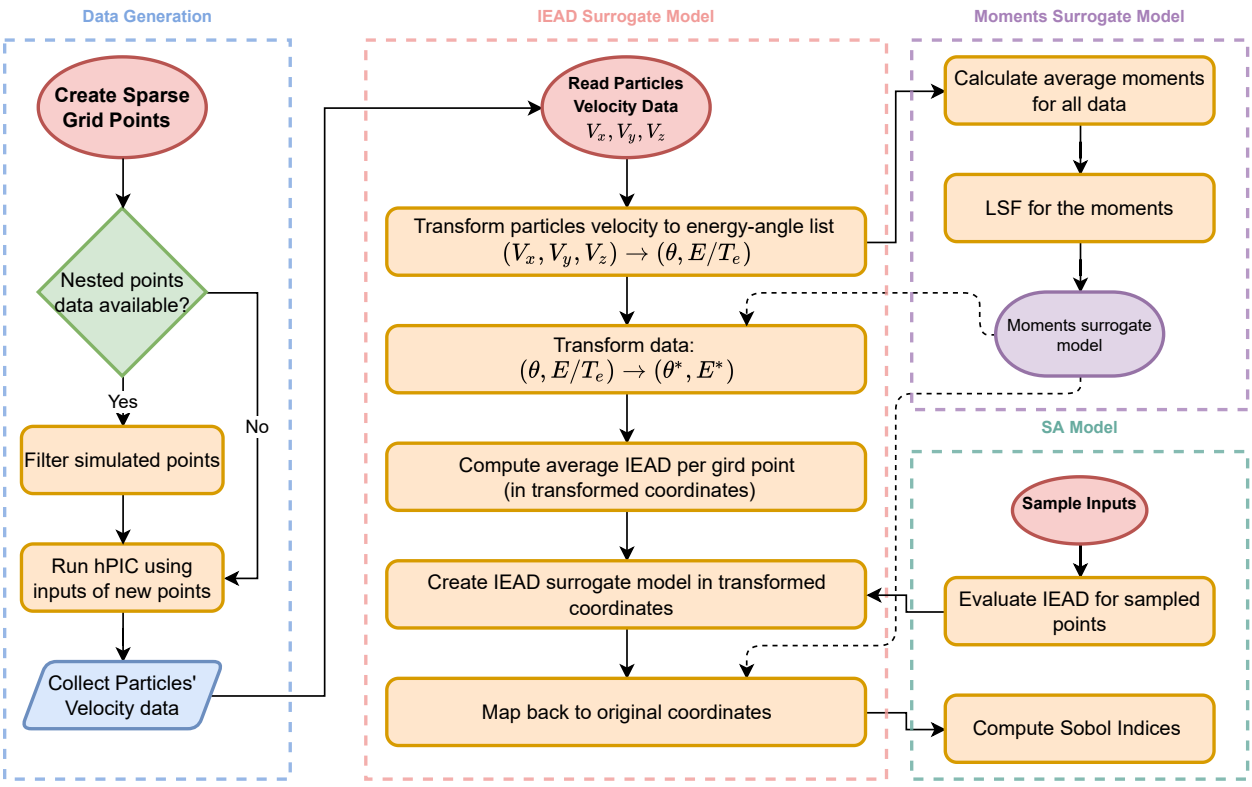


Figure 3.2: Workflow of surrogate model construction and sensitivity analysis. The dashed blue box represents the data generation model utilizing nested **SG**. The red dashed box is the actual surrogate model construction model which in turn uses the moments surrogate model inside the magenta dashed box. Finally, the green dashed box represents the sensitivity analysis model adopted in this work.

passed to the **IEAD** surrogate model construction block. The surrogate model construction sub-workflow reads the particles' velocities lists and computes the corresponding lists of scaled ions' energies and ions' angles $(E/T_e, \theta)$ that are used later to construct the moments surrogate model. Moments surrogate model itself is used to perform coordinate transformation for each particle in the $(E/T_e, \theta)$ list, followed by assigning each particle to a bin of the histogram in the transformed coordinate of **IEAD** (i.e., constructing **IEAD***) for each simulation. Then over-sampled **LSF** is performed on **IEAD*** bins from all simulations to construct the **IEAD*** surrogate model. Finally, inverse transformation to retrieve **IEAD** in the original coordinates using the moments surrogate model can be used when needed. The **SA** sub-workflow samples input list using any known sampling technique (e.g., Saltelli sampling), then send the input vectors to the surrogate model and receive **IEAD** in the original coordinates (or transformed coordinated if only **IEAD** moments are needed) followed by **SA** interpretation on **RoIs**.

The default **IEAD** output of **hPIC** has a form of 240×90 histograms, with the y-axis representing the normalized incident energies of ions E/T_e and the x-axis representing the incident

polar angles θ in degrees. This format makes it difficult to construct a surrogate model because the support of the distribution changes significantly for different input vectors as shown in figures 3.4 b) & 3.4 a). To resolve this issue, the surrogate model has been constructed using a transformed coordinates system for the **IEAD**s denoted as **IEAD***s. Coordinates transformation summarize as follows: Given a set of N ions arriving at the **PSI** with **RoI** data array given by $\mathbf{X} \in \mathbb{R}^{d\times N}$ defined by $\mathbf{X} = [\mathbf{x}_1 \ \mathbf{x}_2 \ \cdots \ \mathbf{x}_N]$, such that $\mathbf{x}_n \in \mathbb{R}^{d\times 1}$ is particle features of the n^{th} particle data. Then the first moment is given by Equation 3.4. In this analysis, the output features of interest are $\mathbf{x}_n = [E/T_e \ \theta]$. The second moment is defined as covariance tensor $\Theta \in \mathbb{R}^{d\times d}$ given by Equation 3.5.

$$\bar{\mathbf{x}} = \frac{1}{N}\sum_{n=1}^{N} \mathbf{x}_n \tag{3.4}$$

$$\Theta = \sum_{n=1}^{N} (\mathbf{x}_n - \bar{\mathbf{x}}) \bigotimes (\mathbf{x}_n - \bar{\mathbf{x}}) \tag{3.5}$$

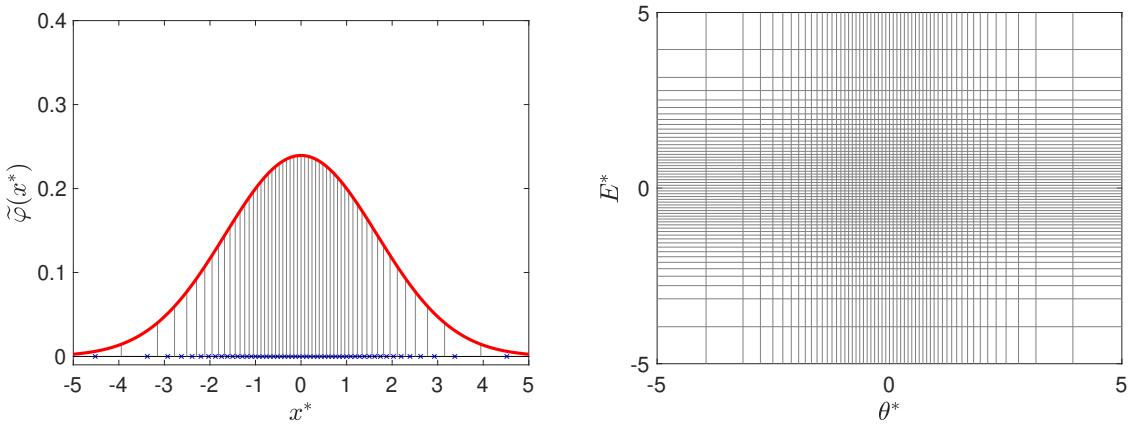
Data scaling to the transformed coordinate system is performed according to Equation 3.6. Figure 3.4 shows an example of **IEADs** in both original and transform coordinates. It can be noted that the support of the distribution in the transformed distribution for various inputs is comparable. In fact, it can be shown using direct calculations that the scaled data has a mean value $\bar{\mathbf{x}}^\star = 0$ and covariance tensor $\Theta^\star = I$. An additional benefit of using a transformed coordinate system is that a non-uniform binning of the **IEAD*** can be employed, a schematic of the non-uniform binding is shown in Figure 3.3. This can reduce the cost of surrogate model construction and enhance its accuracy. Since a finer mesh for the **IEAD*** is used toward the center of the distribution where more variation is expected and the mesh becomes coarser as moving away from the center. Due to the noise in PIC simulations, simulations associated with each input vector have been executed 10 times for the 2D cases and 40 times for the 4D cases[1], then, data were averaged out.

$$\mathbf{x}_n^\star = \Theta^{-1/2}(\mathbf{x}_n - \bar{\mathbf{x}}), \quad n = 1, \ldots, N \tag{3.6}$$

Performing coordinates transformation according to Equation 3.6 requires knowledge of moments $\bar{x}$ and Θ. More specifically, since the **IEAD** problem has two features of interest, five values of moments are needed. Namely, the mean of the features $\bar{E}/T_e$ and $\bar{\theta}$; and the three entries of the covariance matrix Θ_{11}, Θ_{22}, and Θ_{12}[2]. While the moments can be evaluated directly

[1] Due to the computational resources limit; 1000 **PPC** was used for 2D simulations and 400 **PPC** for 4D. Otherwise, this step could have been replaced with one finer simulation per point

[2] Note that $\Theta_{12} = \Theta_{21}$



a) 1D nonuniform binning

b) 2D nonuniform binning

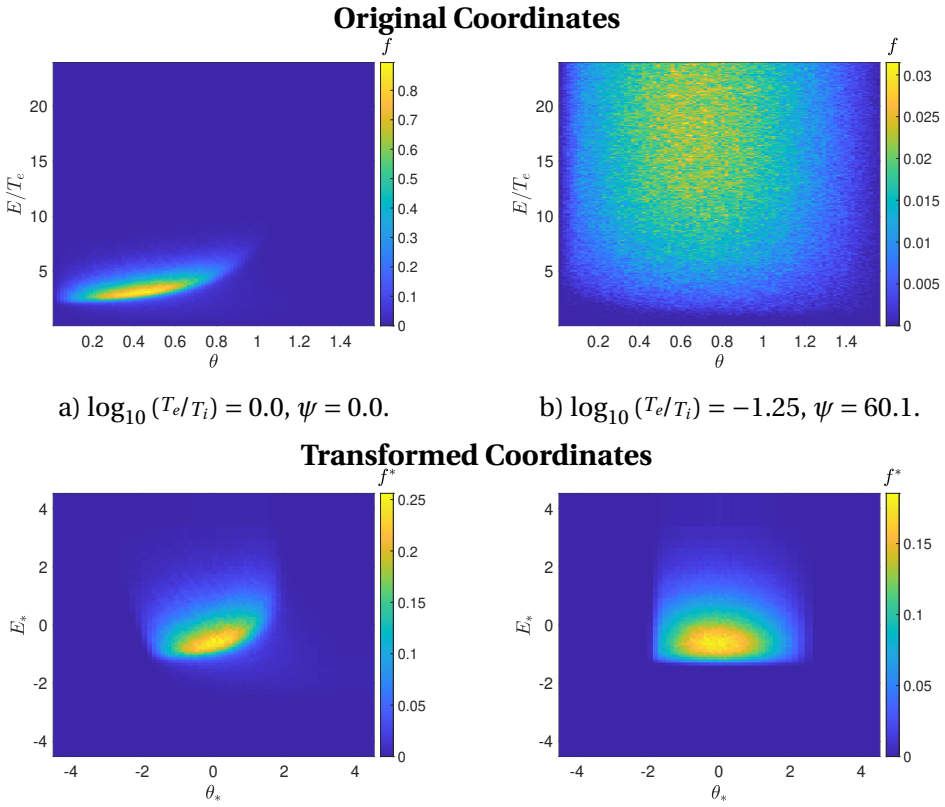
Figure 3.3: Illustration of non-uniform binning, in 1D (3.3 a)) generated by a scaled standard normal distribution. And in 2D (3.3 b)) generated by tensor product of the 1D non-uniform binning.

from the data, surrogate models were constructed for the moments. Such a step can be justified by two reasons. First, due to the particle noise in **PIC** simulations, the value of the moments cannot be uniquely determined for a given choice of physical parameters; this can lead to different transformations for different repetitions with the same physical parameters. Second, it provides a rule for inverse transformation to original coordinates as can be seen in Equation 3.9. A limitation of this approach is that it introduces additional sources of error, one associated with the construction of moments' surrogate models, and one associated with **IEAD*** surrogate model construction.

Moments of each point in the physical parameters space are computed as the mean of the moments values over the number of repetitions. The surrogate model for each moment ($\bar{x}_1 = \bar{\theta}, \bar{x}_2 = {}^{\bar{E}}/_{T_e}, \Theta_{11}, \Theta_{22}$, and $\Theta 12$) was constructed using the least square approximation described in section 3.1.2. Since Θ_{11} and Θ_{22} are non-negative by definition, the surrogate models for these moments were constructed using logarithmic values to ensure a non-negative response surface. Similarly, but instead due to the exponential dependence of the energy profile on electrons-to-ions temperatures ratios ${}^{T_e}/_{T_i}$, the logarithmic value of ${}^{\bar{E}}/_{T_e}$ was used for the construction.[3]

Proceeding the coordinates transformation, an average distribution $f^\star\left(\theta^\star, E^\star\right)$ is directly constructed from the transformed data. The transformed distribution now provides the data to construct the **IEAD*** surrogate model as follows. With a distribution function **IEAD*** represented as $f^\star\left(\theta^\star, E^\star; \mathbf{p}\right)$, where $\mathbf{p} \in \mathbb{R}^m$ is a vector of m physical parameters, the distribution function maps $f^\star : \mathbb{R}^2 \times \mathbb{R}^m \rightarrow \mathbb{R}$. We seek an approximation in the form of Equation 3.7. Where,

[3]To guarantee that the coordinate transformation is orientation-preserving, the moments surrogate model should satisfy the condition $\det(\Theta) > 0$ at any evaluation points.



a) $\log_{10}(T_e/T_i) = 0.0$, $\psi = 0.0$.

b) $\log_{10}(T_e/T_i) = -1.25$, $\psi = 60.1$.

c) $\log_{10}(T_e/T_i) = 0.0$, $\psi = 0.0$.

d) $\log_{10}(T_e/T_i) = -1.25$, $\psi = 60.1$.

Figure 3.4: Example of IEAD in the original and transformed coordinates, with varying T_e/T_i and ψ only; fixing other input parameters at $B = 8[T]$ and $n = 1 \times 10^{18}[m^{-3}]$. IEADs have been constructed from velocity space at PSI and averaged out using 10 simulations with 10^3 PPC.

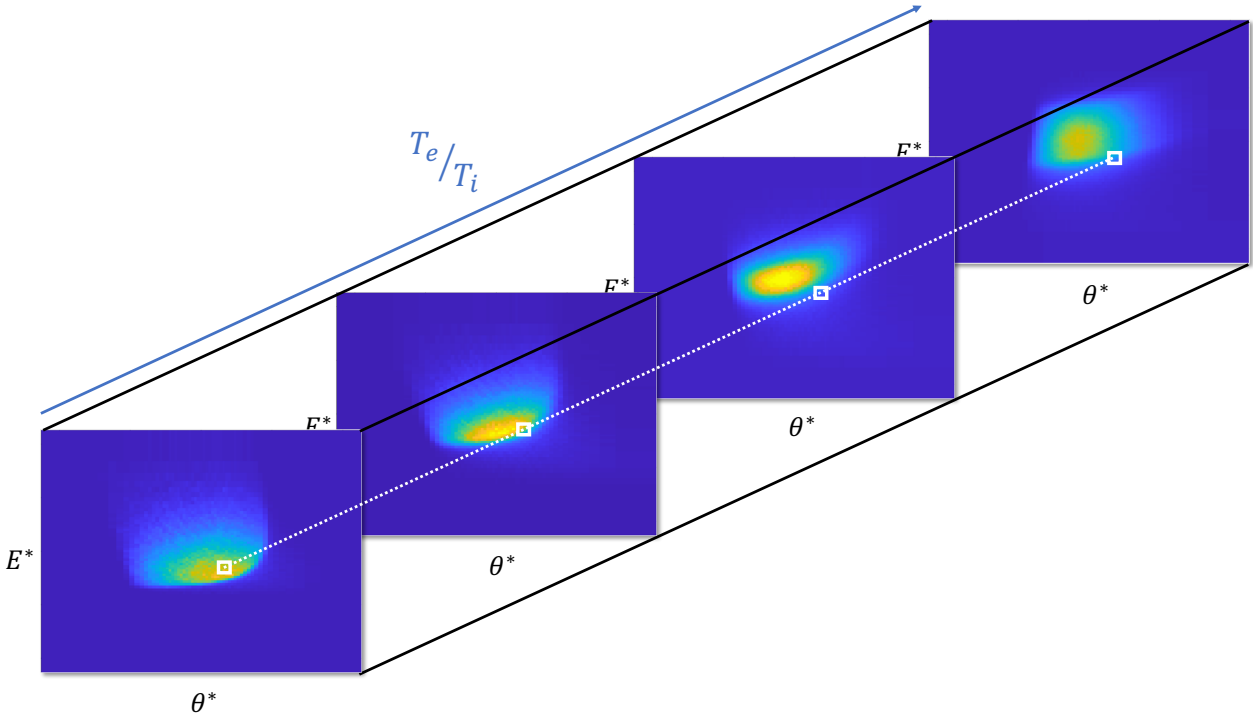


Figure 3.5: Illustration of bin-wise surrogate model construction for **IEAD*** in transformed coordinates. The figure illustrates the construction for a 1D hypothetical case with variable electron-to-ion temperature ratios T_e/T_i. The white square represents a bin and the dashed white line is the trajectory of **IEAD*** values in the path of varying the temperature ratios represented by the blue vector.

$\{B_{l^\star}(\theta^\star, E^\star)\}_{l^\star=1,\ldots,N_b^\star}$ are the basis functions, with the choice of basis functions given by Equation 3.8. Then, $N_b^\star$ is the number of bins and $b_{l^\star(\mathbf{p})}$ its corresponding coefficients at the $l^\star$th bin for a given value of physical parameter $\mathbf{p}$.

$$f^\star(\theta^\star, E^\star; \mathbf{p}) = \sum_{l^\star}^{N_b^\star} b_{l^\star}(\mathbf{p}) B_{l^\star}(\theta^\star, E^\star) \tag{3.7}$$

$$B_{l^\star}(\theta^\star, E^\star) = \begin{cases} 1 & (\theta^\star, E^\star) \in l^\star\text{th bin}, \\ 0 & \text{else}. \end{cases} \tag{3.8}$$

After determining the basis function, the goal now is to find an appropriate function to map the coefficients with input parameters. Which was achieved using the least square approximation introduced in section 3.1.2. The least square fitting was computed for each bin as illustrated in Figure 3.5. In other words, each bin has its own surrogate model in the transformed coordinates system.

For a discrete **IEAD*** prediction $\tilde{f}^{\star}\left(\theta^{\star}, E^{\star}\right)$ evaluated in the transformed coordinates, using the non-uniform binning illustrated in Figure 3.3. The inverse transformation of the prediction to the original coordinates $f(\theta, {}^{E}/_{T_e})$ is performed according to Equation 3.9. Where $l = 1, \ldots, 21600$ is the bin number of the fine 240×90 uniform binning of the original coordinates.

$$f\left(\theta_l, {}^{E_l}/_{T_e}\right) \approx \tilde{f}\left(\theta_l, {}^{E_l}/_{T_e}\right) = \frac{\tilde{f}^{\star}\left(\theta_{l^{\star}}, {}^{E_{l^{\star}}}/_{T_e}\right)}{\left|\det(\Theta)\right|^{1/2}} \tag{3.9}$$

3.2 Simulations Setup and Data Ranges

hPIC [4] was used to simulate all **RoI**s needed for surrogate model training and testing, including the **IEAD** surrogate model in section 3.3 and the potentials surrogate model in chapter 4. A set of input parameters that are expected to have a significant impact on IEAD within ranges relevant to Tokamak operating parameters (see, e.g., Table 2.1.1 in [22]) were chosen for demonstrating the surrogate models' construction procedures. Table 3.2 summarizes the input parameters selected for the current analysis. The reason behind choosing such a set of input parameters is evident from the analysis in chapter 2, which can be summarized as follows: the total potential drop given by Equation 2.13 is a function of ion-to-electron mass ratio ${}^{m_i}/_{m_e}$ and electron-to-ion temperature ratio ${}^{T_e}/_{T_i}$, the former is constant since the analysis was on *Hydrogen* plasma for simplicity. The potential drop in the **MPS** is solely a function of the magnetic field inclination angle ψ as Equation 2.21 reads. Thus, electron-to-ion temperature ratio ${}^{T_e}/_{T_i}$ and magnetic field inclination angle with respect to surface normal ψ are expected to have the most significant impact on the **IEAD** at **PSI**. Accordingly, the 2D case was limited to $({}^{T_e}/_{T_i}, \psi)$. Additionally, **DS** thickness scales to a few Debye lengths λ_D which is a function of the electron temperature T_e and plasma density n. On the other hand, the **MPS** thickness length scales with Larmor radius $\rho = \frac{m v_{\perp}}{qB}$ which is a function of magnetic field strength B. Thus, the 4D cases are extended to include plasma density n and the magnetic field strength B. The latter two parameters were fixed to $n = 1 \times 10^{18}$ $[m^{-3}]$ and $B = 8$ $[T]$ for the analysis of the 2D cases.

Table 3.2: Summary of input parameters and its corresponding range used in surrogate models construction.

Parameter	Range	Case
$log_{10}({}^{T_e}/_{T_i})$	$[-1.25, 1.25]$	2D, 4D
ψ $[deg]$	$[0.0, 87.0]$	2D, 4D
B $[T]$	$[0.0, 15.0]$	4D
$log_{10}(n_0)$ $[m^{-3}]$	$[16, 20]$	4D

Table 3.3: Summary of numerical parameters setup of **hPIC** simulations used to construct the surrogate model.

Numerical Parameter	2D cases	4D cases
Domain Size	$200\lambda_D$ or $3\,\text{ℓ}$	
Number of Cells	1 cell per λ_D	
N_{PPC}	1,000	400
N_{sims}	10	40

Numerical input parameters for all simulations are summarized in Table 3.3. Domain size was chosen to be at least 200 Debye lengths λ_D with one cell per λ_D unless ions' Larmor radius is comparable to the domain size (i.e., $3\,\text{ℓ} > 200\lambda_D$) then the domain size was set as $3\,\text{ℓ}$. The number of particles per cell was selected such that its multiplication with the number of repetitions is constant (i.e., $N_{PPC} \times N_{sims} = const.$), more specifically, for 2D simulations, $N_{PPC} = 1{,}000$ with 10 repetitions, and for the 4D cases, $N_{PPC} = 400$ with 40 repetitions. To gain flexibility in surrogate model construction, **IEAD**s were saved as a list of ions' velocities (V_x, V_y, V_z) at **PSI** as demonstrated in Figure 3.2, instead of 2D histograms of ions' energies and angles.

3.3 IEAD Surrogate Models Training and Testing

Using the tools developed in sections 3.1.1-3.1.3. **IEAD** surrogate model was constructed for data ranges given in section 3.2. Training data was produced using simulation data points with inputs associated with **SG** of levels $L_{data} = 7, 10\&13$. On the other hand, the polynomial levels used for the least square approximation were $L_{model} = 5, 8\&11$ following the rule $L_{model} = L_{data} - 2$, such an approach reduces the effect of particles' noise on the surrogate model as discussed in section 3.1.2. All moments' surrogate models were constructed using $L_{fit} = 4$. Comparison of the number of coefficients and basis functions needed to be evaluated during the surrogate models' construction of both the **IEAD** model and moments model at various polynomial levels of interest are tabulated in 3.4. The 5 in the moments surrogate model number of coefficients is associated with the number of moments. And, the 3136 in the **IEAD** surrogate models are the number of bins in the non-uniform **IEAD*** grid shown in Figure 3.3 b).

Two sources of error can be identified in the **IEAD** approximation. The first is moments' surrogate model related error resulting from the approximation of the moments rather than using the exact values of moments. The second is related to **IEAD** surrogate model error, resulting from the bin-by-bin fitting procedure demonstrated in Figure 3.5. Errors of the **IEAD** surrogate model have been computed with respect to the training data (denoted as "training error") and

Table 3.4: Number of basis functions and coefficients needed to construct moments surrogate model **IEAD** surrogate model for different model's polynomial levels.

Surrogate Model	L_{model}	Number of Basis Functions		Number of Coefficients	
		2D	4D	2D	4D
Moments	4	21	137	5×21	5×137
IEAD	5	29	249	3136×29	3136×249
IEAD	8	65	1041	3136×65	3136×1041
IEAD	11	129	3185	3136×129	3136×3185

Table 3.5: Moment surrogate model training error for 2D and 4D data associated with various **SG** levels.

L_{data}	$\overline{\theta}$	$\overline{E}/T_e$	Θ_{11}	Θ_{22}	Θ_{12}
	2D				
7	0.0088	0.0012	0.017	0.012	0.025
10	0.0097	0.0012	0.017	0.017	0.036
13	0.0097	0.0012	0.017	0.017	0.035
	4D				
7	0.017	0.0060	0.041	0.018	0.040
10	0.020	0.0066	0.047	0.022	0.044
13	0.020	0.0065	0.049	0.021	0.064

with respect to data points different from the training point (denoted as "testing error").

The moments' surrogate model errors were assessed by evaluating the relative error, as follows. For a vector $\mathbf{z}$ containing moments values at a given point on the **SG** for a certain polynomial level L_{data}. The associated approximation provided by the moments surrogate model is denoted as $\tilde{\mathbf{z}}$. Then the relative error is evaluated as $^{||\tilde{\mathbf{z}}-\mathbf{z}||_2}/_{||\mathbf{z}||_2}$ such that $||.||_2$ is the L^2-norm. The values for surrogate model training error and testing error are tabulated in tables 3.5 and 3.6, respectively.

Errors for **IEAD*** have been estimated in a similar fashion. Such that, the error between the actual value of **IEAD*** data point $f^*_{m^*}(\mathbf{p})$ for the $m^* th$ bin for a given input vector $\mathbf{p}$ and the **IEAD*** approximation evaluated using the surrogate model was computed according to Equation 3.10, where $A^*_{m^*}$ is the area of the $m^* th$ pin. The overall error for all points was calculated using Riemann's sum over the **IEAD*** domain. A histogram of the errors' distributions calculated with respect to the training data points (training error) are shown in Figure 3.7. Values of means and standard deviations of **IEAD** surrogate model "training errors" at various **SG** levels are tabulated in 3.7 and show that as the level increases, errors the mean decreases, and its standard deviation

Table 3.6: Moment surrogate model training error for 2D and 4D data associated with various **SG** levels.

L_{data}	$\overline{\theta}$	$\overline{E}/T_e$	Θ_{11}	Θ_{22}	Θ_{12}
	2D				
7	0.0141	0.0020	0.0214	0.0183	0.0778
10	0.0128	0.0017	0.0192	0.0223	0.0672
13	0.0119	0.0018	0.0197	0.0202	0.0660
	4D				
7	0.0237	0.0070	0.0583	0.0217	0.0676
10	0.0233	0.0070	0.0528	0.0236	0.0663
13	0.0231	0.0070	0.0536	0.0226	0.0641

Table 3.7: Statistical values of **IEAD** surrogate model "training errors" for 2D and 4D data associated with various **SG** levels.

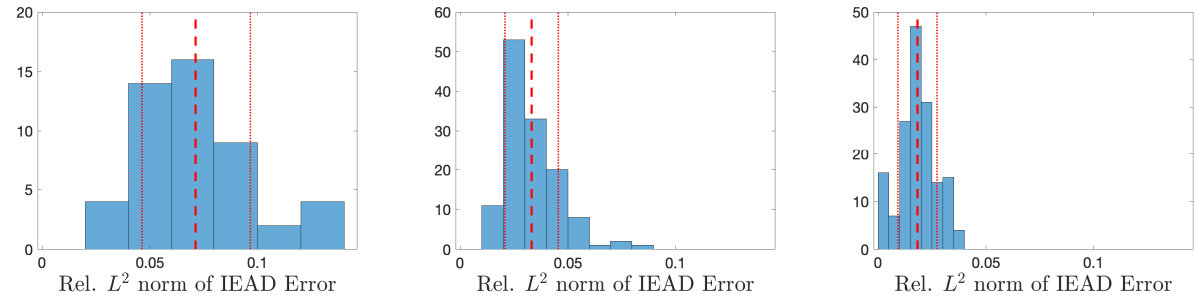
L_{model}	Mean	Standard Deviation	Max
	2D		
5	0.0715	0.0252	0.1331
8	0.0331	0.0124	0.0853
11	0.0183	0.0091	0.0392
	4D		
5	0.0838	0.0485	0.4506
8	0.0365	0.0221	0.2414
11	0.0234	0.0108	0.1473

reduces.

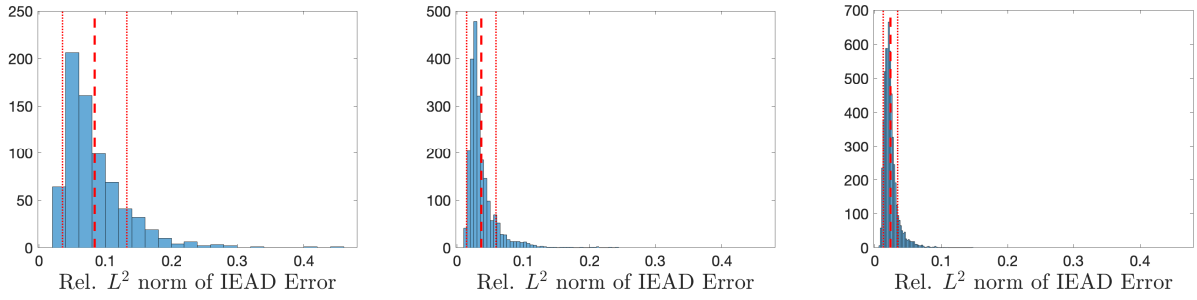
$$e(\mathbf{p}) = \frac{\left|\left|\tilde{f}^*(\mathbf{p}) - f^*(\mathbf{p})\right|\right|_{L^2}}{\left|\left|f^*(\mathbf{p})\right|\right|_{L^2}} \approx \frac{\left[\sum_{m^*=1}^{M^*} \left(\tilde{f}^*(\mathbf{p}) - f^*(\mathbf{p})\right)^2 A^*_{m^*}\right]^{1/2}}{\left[\sum_{m^*=1}^{M^*} \left(f^*(\mathbf{p})\right)^2 A^*_{m^*}\right]^{1/2}} \tag{3.10}$$

Distributions of **IEAD** surrogate model "training error" in the physical parameters space for the 2D cases are shown in Figure 3.6, where it is clearly evident that the error decreases as the total polynomial level increases.

In order to test the **IEAD** surrogate model's predictive capabilities, a uniformly spaced grid with input parameters within ranges listed in Table 3.2 was created. A 400 points grid (20×20) was used for the 2D cases testing, as shown in Figure A.1 in Appendix A. And a 625 points grid ($5 \times 5 \times 5 \times 5$) was chosen for the 4D cases. Simulations for the testing grid points were executed

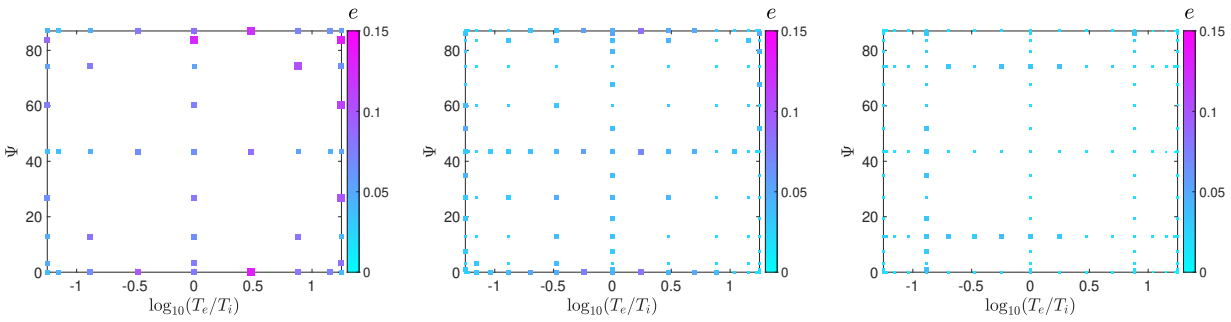


a) 2D: $L_{model} = 5$ b) 2D: $L_{model} = 8$ c) 2D: $L_{model} = 11$



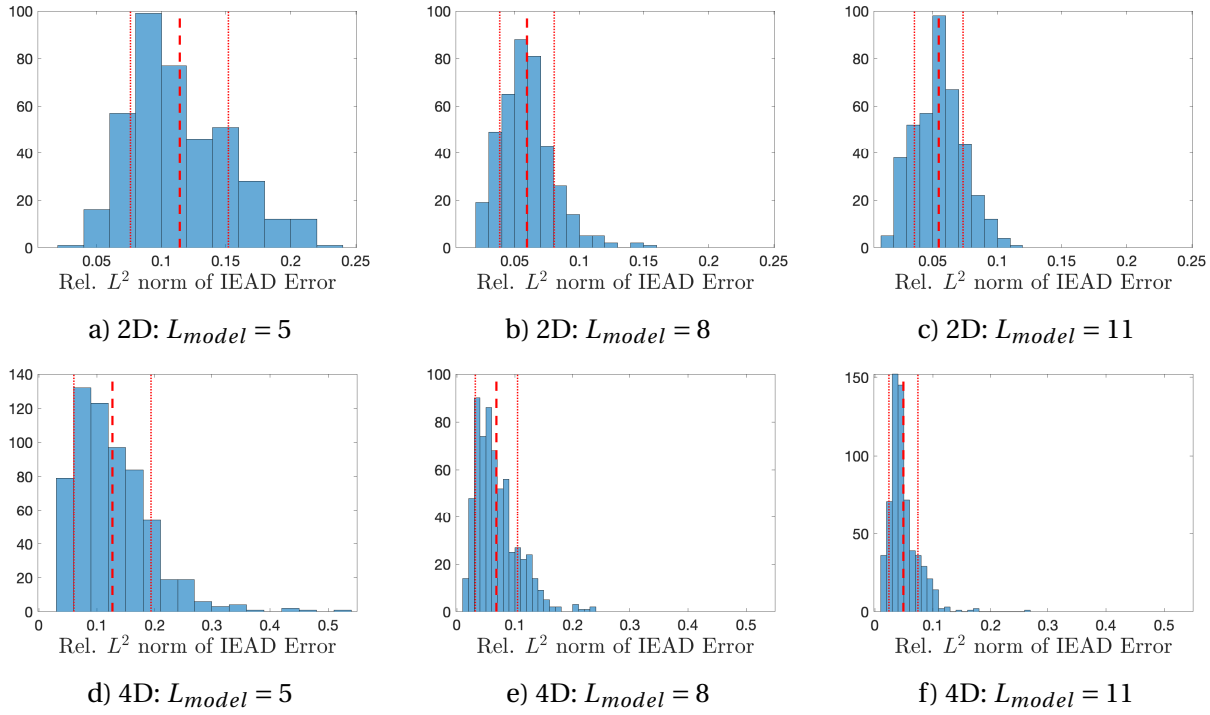
d) 4D: $L_{model} = 5$ e) 4D: $L_{model} = 8$ f) 4D: $L_{model} = 11$

Figure 3.6: **IEAD** surrogate model "training errors" in 2D (top row) and 4D (bottom row) for $L_{model} = 5, 8 \& 11$. The red dashed line represent the mean and the red dotted lines represents the standard deviation boundaries.



a) $L_{model} = 5$ b) $L_{model} = 8$ c) $L_{model} = 11$

Figure 3.7: **IEAD** surrogate model "training errors" for each simulation in 2D **SG**. The color intensity and marker size represents the error for each simulation's point.



a) 2D: $L_{model} = 5$ b) 2D: $L_{model} = 8$ c) 2D: $L_{model} = 11$

d) 4D: $L_{model} = 5$ e) 4D: $L_{model} = 8$ f) 4D: $L_{model} = 11$

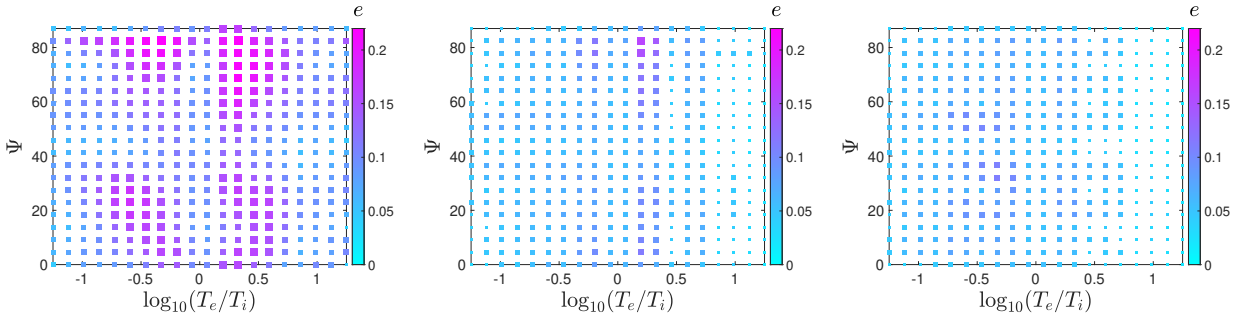
Figure 3.8: **IEAD** surrogate model "testing errors" in 2D (top row) based on the grid in Figure A.1 And 4D (bottom) row for $L_{model} = 5, 8 \& 11$. The red dashed line represents the mean and the red dotted lines represent the standard deviation boundaries.

using hPIC with similar numerical parameters used in the original data generation. Histograms of the **IEAD** testing errors are shown in Figure 3.8 as calculated by Equation 3.10, in addition, statistics about errors are shown in Table 3.8. The top row summarizes testing errors for 2D cases, and the bottom row shows errors for 4D cases. The advantage of using a higher polynomial level is evident from Table 3.8, it can be noted that increasing the total polynomial level decreases the mean and maximum of the error as well as the standard deviation. Although, comparing values of "training errors" in Table 3.7 and "testing errors" in Table 3.8 it can be noted that the overall error is larger for the "testing error", which is expected since the training procedures were completely agnostic of the testing data points. **IEAD** testing errors at each point on the testing grid physical parameters space for the 2D cases is shown in Figure 3.9.

Besides eliminating the particles' noise of the simulation, the surrogate model significantly reduces the computational cost, which is especially important when a large number of simulations are needed (e.g., when performing **SA**). Table 3.9 shows a comparison of the estimated computational time required to generate the data needed to construct the surrogate model using hPIC Vs. using the surrogate model. hPIC time was estimated as $T = N_t \times t_{step}$, where N_t is the total number of time-steps and t_{step} is the time estimated per time-step. The latter was estimated as $t_{step} = N_{part} \times t_{field} + N_{nodes} \times t_{pp}$, where N_{part} is the total number of computational

Table 3.8: Statistical values of **IEAD** surrogate model testing error for 2D and 4D data associated with various **SG** levels.

L_{model}	Mean	Standard Deviation	Max
	2D		
5	0.1141	0.0381	0.2218
8	0.0594	0.0210	0.1594
11	0.0546	0.0188	0.1105
	4D		
5	0.1270	0.0666	0.5357
8	0.0682	0.0365	0.2388
11	0.0492	0.0250	0.2630

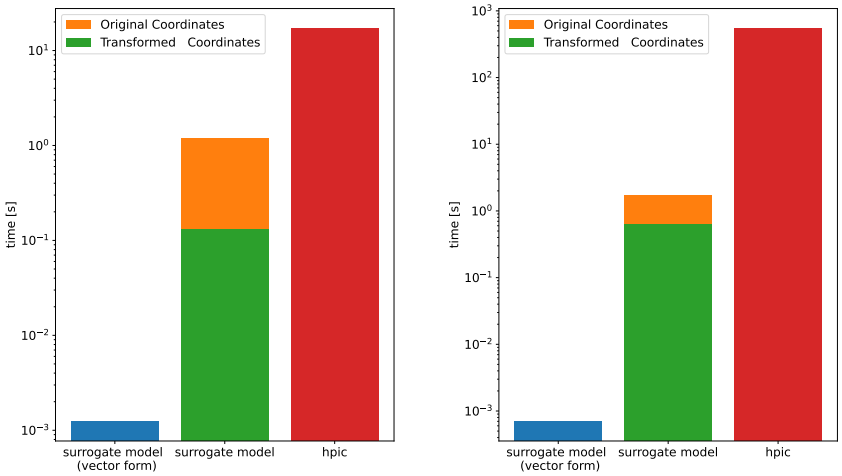


a) $L_{model} = 5$ b) $L_{model} = 8$ c) $L_{model} = 11$

Figure 3.9: **IEAD** surrogate model "testing errors" for simulations on the uniform testing grid shown in Figure A.1. Color intensity and marker size represents the error for each simulation's point.

Table 3.9: Comparison of the computational time estimated for **IEAD**s evaluation using hPIC and the **IEAD** surrogate model for training data with polynomial level $L_{data} = 13$.

Data Set	Number of Points	hPIC Time	**SM** Time	Reduction in Time
2D ($L_{data} = 13$)	161	7.45 $[hr]$	40 $[s]$	90.05%
4D ($L_{data} = 13$)	5569	186.46 $[day]$	21 $[min]$	99.53%



a) 2D b) 4D

Figure 3.10: Comparison of the total computational time needed to generate **IEAD**s for input data of $L_{data} = 13$, using the surrogate model in vectorized form, surrogate model including the inverse coordinates transformation, and hPIC simulations. Showing the comparison for 2D in 3.10 a) and 4D in 3.10 b).

particles in the domain, $t_{field} = 70[ns]$ is the field solver time, $N_{nodes} = N_{cells} + 1$ is the number of nodes in the domain, and $t_{pp} = 1500[ns]$ is the particle push time. The number of steps was selected such that the total time is at least one ion transit time. The number of cells equal to the number of Debye lengths λ_D in the domain. In general, domain size is $x = 100\lambda_D$ unless the Larmor's radius $2\,\hat{r}$ is larger. The time of the surrogate model includes two steps, evaluating the surrogate model in transformed coordinates (**IEAD***), and the inverse transformation to original coordinates (**IEAD**). If the inverse mapping of **IEAD*** to **IEAD** is not needed, then the surrogate model evaluation can be performed in a vectorized form, leading to a significant reduction in computation time. A summary of the estimated time required to evaluate **IEAD** for input parameters used to construct the surrogate model with $L_{data} = 13$ is shown in Figure 3.10.

3.4 Main Takeaways

- A strategy for constructing a surrogate model for the **IEAD** in high-dimensional parameters space is introduced. The surrogate model's goal is to efficiently emulate **IEAD** without compromising accuracy.

- Training and testing **IEAD**s data was produced by hPIC simulations with input parameters ranges relevant to magnetically confined fusion devices. For simplicity, Hydrogen plasmas were used in the construction. Up to four parameters are considered in the construction. Namely, electron-to-ion temperature ratios T_e/T_i, the magnetic field inclination angle with respect to surface normal ψ, magnetic field strength T, and plasma density n. Such input parameters are expected to have the most significant impact on **IEAD** according to the analysis in chapter 2.

- The strategy for surrogate model construction is summarized as follows:

    1. Inputs parameters list was sampled using a sparse grid based on Clenshaw-Curtis polynomials, this step aims to reduce the number of simulations required for surrogate model construction.
    2. hPIC was utilized to simulate **IEAD** using the input parameters list in (1), for the current analysis, the output of hPIC has the form of a velocities list (V_x, V_y, V_z) which is then converted to energies and angles list (E, θ).
    3. The **IEAD**s were normalized via coordinates transformation to overcome the challenge of variable support of the **IEAD**. This also allowed for using a nonuniform binning of the **IEAD**.
    4. Surrogate models for the **IEAD** moments were also constructed to overcome the noise of the simulations and to provide a rule for inverse transformation.
    5. A bin-by-bin least square data fitting was presented. The **LSA** used polynomial order of $L-2$, where L is the polynomial level of the data **SG**. Such a step is important to mitigate the effect of particles' noise inherent in the training data and to avoid over-fitting.

- Followed by the IEAD surrogate model construction, it has been tested using data points different than the training data points. The L_2 norm of the error is used as a metric to assess the performance of the surrogate model. The mean and maximum of the training errors for the 4D cases using the $L_{data} = 13$ are 0.0234 and 0.1473 respectively; and for testing data the values are 0.0250 and 0.2630.

Chapter 4

Data-Driven Modeling of Plasma Potentials

Detailed knowledge of plasma potentials is important to understand the behavior of plasma at **PMI**. As reviewed in 2, the total potential drop within the plasma domain in presence of an inclined magnetic field is separated into three layers, known as plasma pre-sheath, plasma magnetic pre-sheath **MPS** (also known as Chodura sheath), and Debye sheath **DS**. While total potential drop controls the total energy of ions at the **PMI**, it does not provide insight about the angle of impact. As demonstrated in chapter 2, ions' impact angle is affected by the initial thermal distribution of ions; the inclination of the magnetic field; the value of potential drop across the **MPS** (which is also a function of ψ), where plasma potential accelerates ions parallel to the magnetic field; and the potential drop across **DS**, where plasma potential accelerates ions in a direction perpendicular to the surface. All such factors affect the resulting **IEAD**.

In this chapter, surrogate models for the response surfaces of floating wall potential, magnetic pre-sheath potential, and Debye sheath potential with variable electron-to-ion temperature ratios T_e/T_i and magnetic field inclination angles ψ were constructed. The construction used the least square approximation introduced in section 3.1.2. In addition, two artificial neural network models were introduced and applied for the "excess zero" cases (i.e., in the cases where a large number of zero outputs is expected).

The importance of plasma potentials surrogate models is summarized as follows: (1) To provide guidance to interpret the results of global sensitivity analysis for **IEAD** in section 5.3.1. (2) To identify the points in the physical parameter space where **DS** dissipates. (3) To provide a tool to estimate the values of potential at a reduced computational cost (e.g., sensitivity analysis in sections 5.3.1).

The structure of this chapter is as follows. Section 4.1 summarized the workflow of constructing the response surfaces used for surrogate models training and testing. The results of surrogate models constructed using the least square approximation are shown in section 4.2. Excess zero artificial neural network models are introduced and applied to the plasma potentials problem in

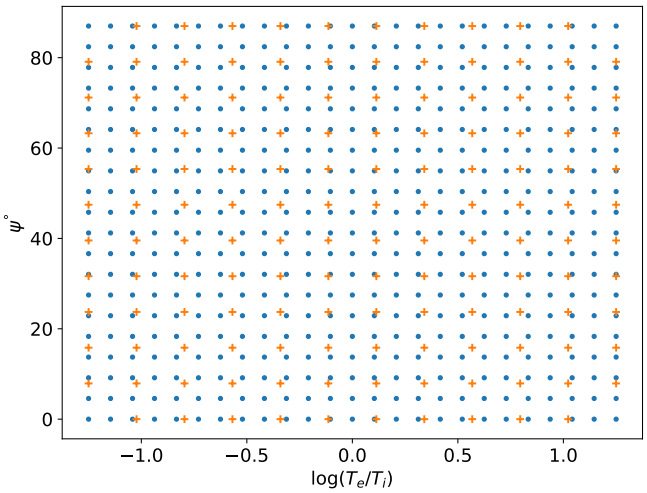


Figure 4.1: Uniform grids used for the **ANN** models training and testing. Blue data points are the training data for the **ANN** models and training data for the **LSF** model. Orange data points are the testing data for the **ANN** model.

section 4.3.

4.1 Construction of plasma potentials training and testing response surfaces

This section reviews the process of constructing the response surfaces for the total potential drop, the potential drop across the magnetic pre-sheath, and the potential drop in the Debye sheath. Data ranges for the current analysis are similar to the 2D cases shown in Table 3.2. The training data for the least square approximation surrogate model was created using input parameters on **SG** with level $L_{data} = 13$ shown in Figure 3.1 a). Training data for the **ANN** models were produced using data points on the uniform grid shown in Figure 4.1. The same data points were used for the **LSF** surrogate model testing. A separate uniform grid with different data points was used for **ANN** models testing as shown in Figure 4.1. Simulations for data points on the **SG** were produced using **hPIC** [4]. And **hPIC2** [5] was used for data points on the uniform grids.

As reviewed in chapter 2, **MPS** entrance is identified as the point where ions' velocities parallel ($V_{||_B}$) to the magnetic field lines is equal to ions' Bohm acoustic speed C_s. Similarly, **DS** entrance is identified as the point where ions' speed parallel surface normal is equal to C_s. Such a definition was used for plasma potential response surfaces construction. While the general workflow used to generate the response surfaces for the **SG** and the uniform grid data

points is similar, there are a few differences that will be pointed out later. The workflow for the **SG** data points is summarized as follows. Zeroth order moment profile (plasma density n), lab frame first order moments profile (momentum density), and plasma potential profile were collected from the simulations. Then, to reduce the particles' noise effect, the average profile for each output was computed using several repetitions at the same data point. Bohm acoustic speed was calculated using input parameters according to Equation 2.12. Using the averaged moments, ions flow velocities (V_x, V_y, V_z) were computed by dividing momentum density in the ith direction by plasma density (i.e., $V_i = P_i / n$). Followed by evaluating ions' velocity parallel to the magnetic field $V_{||}$[1]. The normalized ions' velocity profile with respect to ions' acoustic speed in a direction parallel to magnetic field lines $V_{||}/C_s$ and parallel to surface normal V_x/C_s were used to identify the locations of **MPS** and **DS** entrances, respectively. The corresponding points on the plasma potential profiles are the values of the potentials (ϕ_{MPS}, ϕ_{DS}). The difference in the approach used for the uniform grid data points is that one finer simulation per data point is used instead. In addition, a digital signal filter was applied to the profiles. An example of the procedure used to identify the potentials of **MPS** and **DS** is shown in Figure 4.8. A detailed workflow breaking down the procedures of identifying the potentials drops across the **MPS** and **DS** is shown in Figure 4.2.

4.2 Least Square Approximation Surrogate Model

The results of constructing a surrogate model using the least square approximation of an over-sampled **SG** for plasma potentials are discussed in this section. The data set used for global interpolation surrogate model construction is the sparse grid data at a polynomial level equal to 13 (i.e., $L_{data} = 13$). Polynomial levels used for surrogate model training are variable for each response surface as summarized in Table 4.1, training polynomials' levels were chosen such that the mean and max errors are minimal for both training and testing. The priorities for selecting the fitting polynomial order are as follows: 1. testing error means is minimum, 2. maximum value of testing error is minimum, 3. training error mean is minimum, 4. maximum value of training error is minimum. Surrogate models were constructed for both the absolute value of potential ϕ as well as the dimensionless value $\frac{e\phi}{K_B T_e}$ within data ranges listed in Table 3.2.

In general, it was found that training the model using dimensionless potential response surfaces $\frac{e\phi}{K_B T_e}$ performs significantly better than the response surfaces for the absolute potential drops. Recall that such a problem was partially tackled by constructing the surrogate model on the log of the **RoI**s values in section 3.3. Table 4.1 shows the mean square errors for all **RoI**s as

[1] Note that the magnetic field inclination angle ψ is defined with respect to surface normal in the xy-plane, thus, only V_x and V_y profiles are needed.

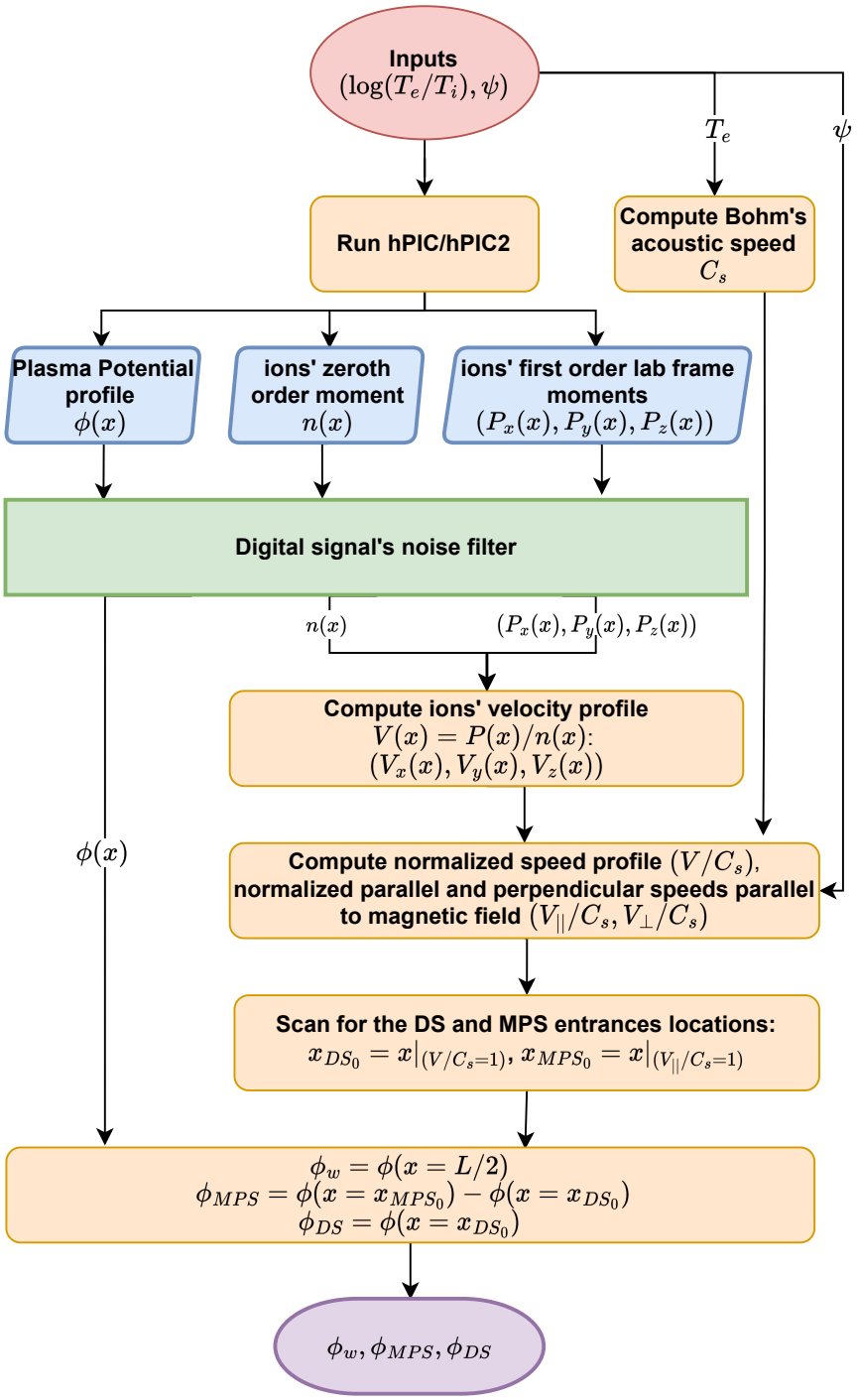


Figure 4.2: Workflow of evaluating the floating wall potential ϕ_w, potential drop across the Debye sheath ϕ_{DS}, and the potential drop across the magnetic pre-sheath ϕ_{MPS}, for a single input point. Showing the post-processing done on hPIC's outputs of interest and the computation of the potential drops according to the discussion in chapter 2.

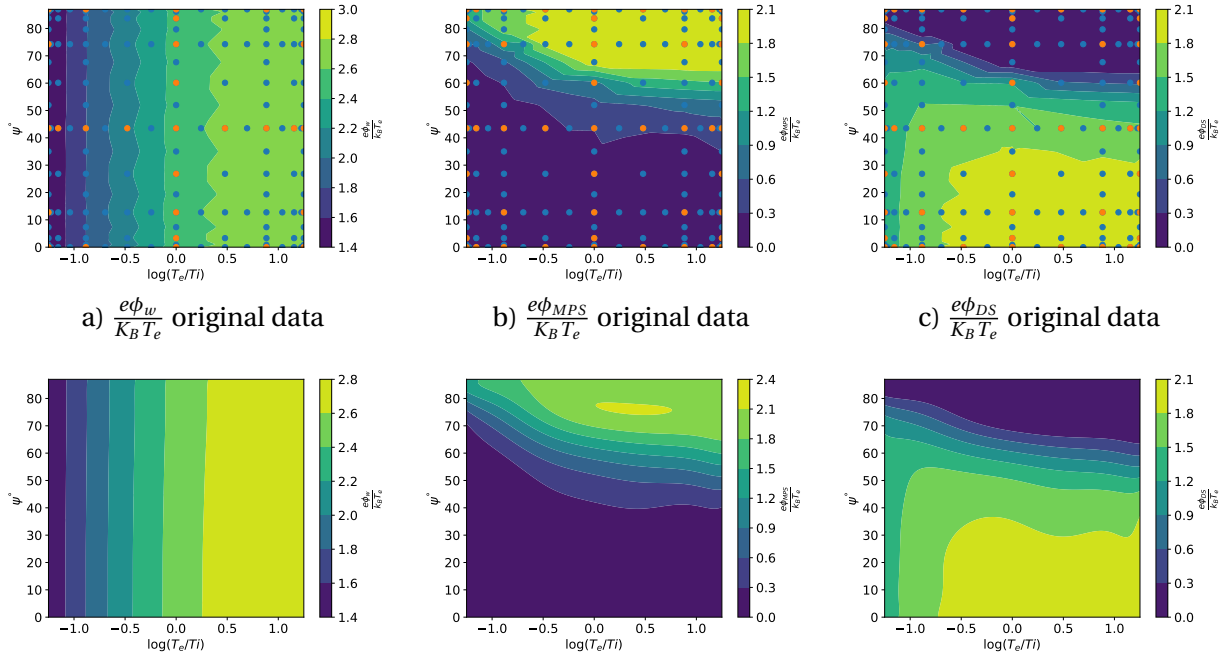
Table 4.1: Summary of surrogate models construction errors for the floating wall potential ϕ_w, the potential drop across the magnetic pre-sheath ϕ_{MPS}, and potential drop across the Debye sheath ϕ_{DS}; as well as the associated dimensionless potential drops. Showing the **MSE** for all cases; polynomial levels for data and training (L_{data}&L_{fit}); errors statistics for each case, percent errors were calculated for ϕ_w cases and absolute errors for all other cases.

RoI	MSE		Error	Training Error			Testing Error			L_{fit}	L_{data}
	Testing	Training	Type	mean	max	stdev	mean	max	stdev		
ϕ_w	1.8×10^{5}	0.0853	$\%Error$	0.7775	11.075	1.581	0.5942	10.969	1.1571	8	13
$\frac{e\phi_w}{K_B T_e}$	0.3864	8.09×10^{-6}	$\%Error$	0.0267	0.1786	0.029	0.1927	1.0436	0.194	5	13
ϕ_{MPS}	1.21×10^{5}	18.3507	$\|Error\|$	1.8105	29.2892	3.8823	2.3311	53.5039	5.0811	6	13
$\frac{e\phi_{MPS}}{K_B T_e}$	2.7021	0.0042	$\|Error\|$	0.0366	0.3387	0.0539	0.0566	0.3888	0.0705	6	13
ϕ_{DS}	6888.6386	11.6318	$\|Error\|$	1.3673	24.2357	3.1244	1.922	54.2819	4.4111	8	13
$\frac{e\phi_{DS}}{K_B T_e}$	2.0822	0.0035	$\|Error\|$	0.0309	0.3277	0.0508	0.0475	0.3923	0.0658	6	13

well as statistics about errors for each **RoI**. Absolute percent errors were calculated for ϕ_w and $\frac{e\phi_w}{K_B T_e}$, while the absolute error was used for all other response surfaces since such surfaces have several zeros. For the reason mentioned above, in later analysis, only dimensionless response surfaces surrogate models of the plasma potential drops are considered.

The response surfaces of plasma potentials generated using the original data and the **LSF** surrogate models are shown in Figure 4.3. It can be noted that the surrogate model of the floating wall potential ϕ_w in Figure 4.3 d) was successfully captured and smoothed as shown in 4.3 a) without significant over-fitting. However, for the other two response surfaces (ϕ_{MPS} and ϕ_{DS} response surfaces) several challenges arise: first, the response behavior is more complicated (i.e., data is more non-linear); second, the presence of exactly zero values (i.e., when the **MPS** or **DS** does not exist) alter the behavior of the fitting; third, the preparation of the raw data involved using the moments, plasma density and potential profile, all such quantities are affected by particles' noise and post-processing propagates and subsequently amplifies the noise. These challenges combined lead to a trade-off between accuracy and over-fitting. The fitted models in figures 4.3 e) and 4.3 f) were found to be the best fits to smooth-out raw data and minimize the errors. Some of these challenges were addressed in the **ANN** method.

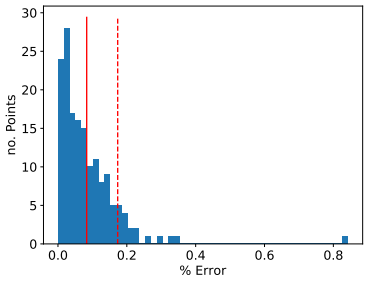
As a means to gain a better understanding of the accuracy of the surrogate models' training procedures, the errors of the surrogate models' predictions at the training data points with respect to the exact values of the training data points were computed, such errors will be referred to as "training error". Histograms of training errors are shown in Figure 4.4. Training errors computed on sparse grid data are shown in Figure 4.5. It was found that the mean value of the absolute percent "training" error of the total potential drop $\frac{e\phi_w}{K_B T_e}$ equal to 0.0267% with a standard deviation of 0.029, Figure 4.4 a) shows that most of the error values are around the mean value with few outliers errors exceed 0.8% such outliers mainly occur at corners of the sparse grid



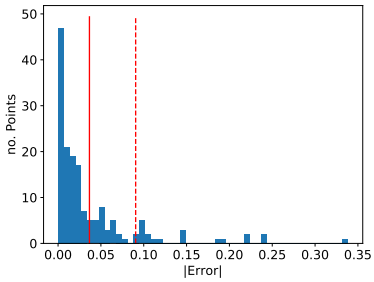
a) $\frac{e\phi_w}{K_B T_e}$ original data b) $\frac{e\phi_{MPS}}{K_B T_e}$ original data c) $\frac{e\phi_{DS}}{K_B T_e}$ original data

d) $\frac{e\phi_w}{K_B T_e}$ surrogate model e) $\frac{e\phi_{MPS}}{K_B T_e}$ surrogate model f) $\frac{e\phi_{DS}}{K_B T_e}$ surrogate model

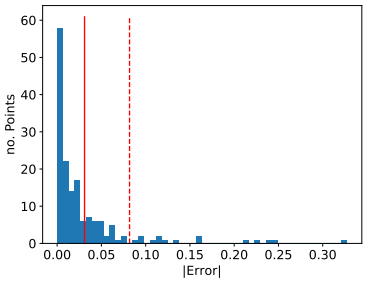
Figure 4.3: Dimensionless potential drops response surfaces of original data (figures. a), b) & c)) Vs. surrogate model (figures d), e) & f)). For ranges shown in Table 3.2. Showing the data points in blue and training points in orange



a) $\frac{e\phi_w}{K_B T_e}$ training absolute percent error



b) $\frac{e\phi_{MPS}}{K_B T_e}$ training absolute error



c) $\frac{e\phi_{DS}}{K_B T_e}$ training absolute error

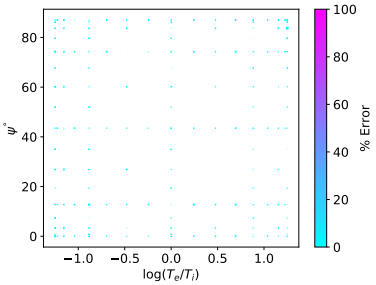
Figure 4.4: Histograms of training errors, represented as absolute percent error for $\frac{e\phi_w}{K_B T_e}$ a), absolute error for $\frac{e\phi_{MPS}}{K_B T_e}$ b), and absolute error for $\frac{e\phi_{DS}}{K_B T_e}$ c). The red solid line represents the mean value of the error and the red dashed line is one standard deviation of error.

as can be seen in Figure 4.5 a). In conclusion, the least square approximation method was able to accurately capture the repose surface of dimensionless plasma's floating wall potential $\frac{e\phi_w}{K_B T_e}$.

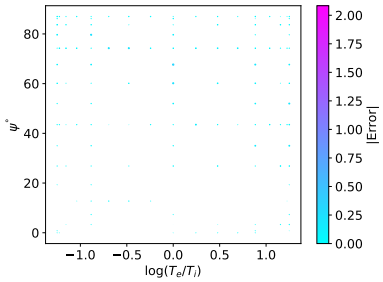
Training errors for the **MPS** and **DS** response surfaces were significantly higher. In addition, a large number of error outliers were observed as shown in figures 4.4 b) and 4.4 c). Figures 4.5 b) and 4.5 c) show that most of the high-valued errors are presented in regions where the exact value of the **RoI** is zero, and near the corners of the **SG**. This suggests three drawbacks of using the least square approximation for such response surfaces: first, the model failed to capture exactly zero values in the response surfaces; second, the model did over-fit the response in the regions of highly non-linear response; third, the regions of zeros affected the capability of the model to capture the non-linearity of the non-zeros regions. The latter can be clearly observed in Figure 4.3 e), where a peck appeared on the fitted surface. The second point can be seen in Figure 4.3 f) where two boundaries of high-valued potentials showed a "wavy" behavior.

In order to assess surrogate models' predictive capabilities, a uniformly spaced testing grid with points different than the sparse grid training points was used. The testing grid data points are shown in Figure 4.1. The error computed on the testing grid points is denoted as "testing error". Potentials "testing error" statistics are summarized in Table 4.1. Histograms of testing error distribution are shown in Figure 4.6. Testing errors computed on the testing grid are shown in Figure 4.7. While the general arguments addressed in the "training errors" discussion hold for "testing errors", the latter have higher values of errors' mean values and standard deviations, as the training process was completely agnostic to the testing data set. One exception is most of the high error points are located on the boundaries of the zeros non-zero regions.

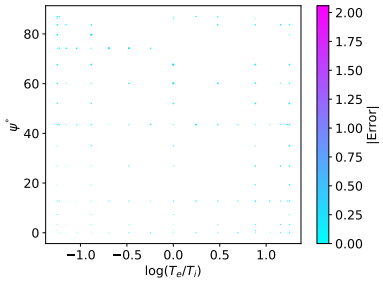
A major advantage of the least square approximation method is the computational efficiency of construction. The construction times for each surrogate model were in the range of $1-3ms$.



a) $\frac{e\phi_w}{K_B T_e}$ training absolute percent error

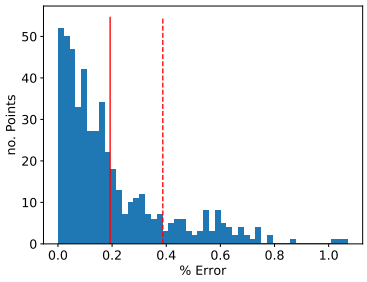


b) $\frac{e\phi_{MPS}}{K_B T_e}$ training absolute error

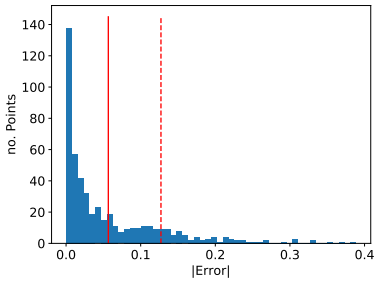


c) $\frac{e\phi_{DS}}{K_B T_e}$ training absolute error

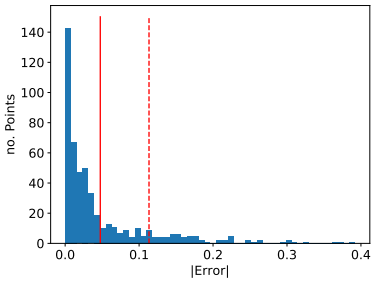
Figure 4.5: Potential surrogate model errors computed on data sparse grid, represented as absolute percent error for $\frac{e\phi_w}{K_B T_e}$ a); absolute error for $\frac{e\phi_{MPS}}{K_B T_e}$ b); and absolute error for $\frac{e\phi_{DS}}{K_B T_e}$ c). The color intensity and marker size represent the relative error magnitude.



a) $\frac{e\phi_w}{K_B T_e}$ testing absolute percent error

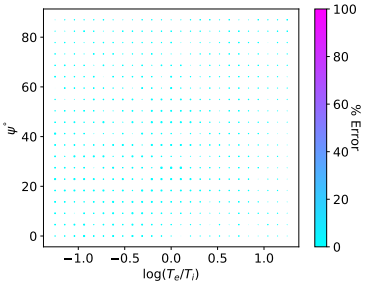


b) $\frac{e\phi_{MPS}}{K_B T_e}$ testing absolute error

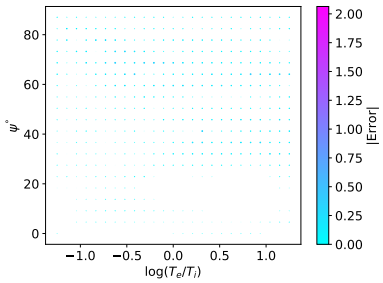


c) $\frac{e\phi_{DS}}{K_B T_e}$ testing absolute error

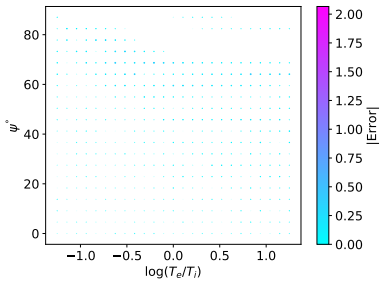
Figure 4.6: Histograms of testing errors, represented as absolute percent error for $\frac{e\phi_w}{K_B T_e}$ a); absolute error for $\frac{e\phi_{MPS}}{K_B T_e}$ b); and absolute error for $\frac{e\phi_{DS}}{K_B T_e}$ c). The red solid line represents the mean value of the error and the red dashed line is one standard deviation of error.



a) $\frac{e\phi_w}{K_B T_e}$ testing absolute percent error

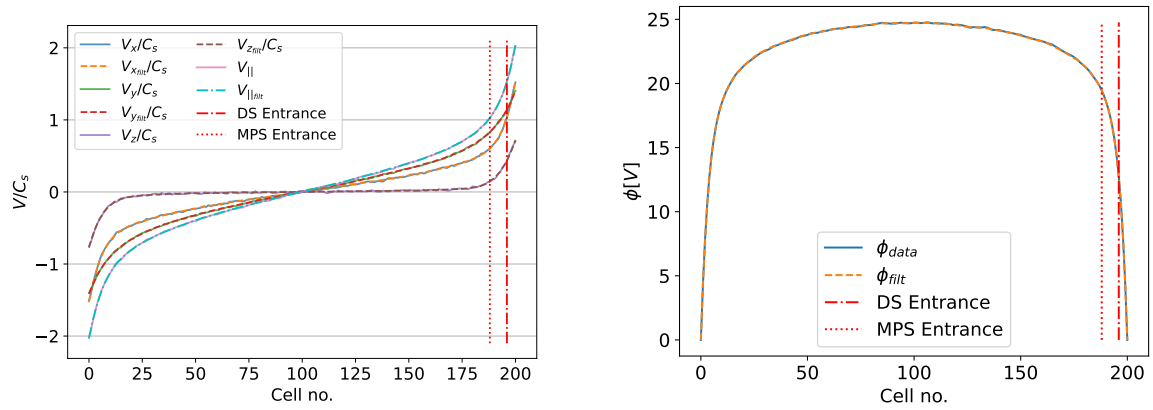


b) $\frac{e\phi_{MPS}}{K_B T_e}$ testing absolute error



c) $\frac{e\phi_{DS}}{K_B T_e}$ testing absolute error

Figure 4.7: Potential surrogate model "testing errors" computed on data sparse grid, represented as absolute percent error for $\frac{e\phi_w}{K_B T_e}$ a); absolute error for $\frac{e\phi_{MPS}}{K_B T_e}$ b); and absolute error for $\frac{e\phi_{DS}}{K_B T_e}$ c). The color intensity and marker size represent the relative error magnitude.



a) b)

Figure 4.8: Example of "Savitzky–Golay filter" applied to hPIC data at ($\log(T_e/T_i) = 0$, $\psi = 55^0$).

While prediction times for testing data points[2] were in the range of $10-50ms$.

4.3 Artificial Neural Network Surrogate model

The limitations of the global interpolation surrogate model method motivated exploring a different method that is capable of capturing the complexity of potentials response surfaces with reduced error. Thus, Artificial Neural Networks **ANN** techniques were considered. By utilizing **ANN**, a number of challenges arise, including: 1. **ANN** techniques are more sensitive to noise, 2. **ANN** is more likely to over-fit the response of training data points, 3. exactly zero values in the response surfaces can bias the overall fitting leading to lower prediction values and preventing the model from achieving sufficiently high accuracy, 4. **ANN**s requires a careful selection of the hyper-parameters (e.g., when the response surface has exactly zero values, the mean absolute percent error cannot be used as a loss function). To tackle such a problem several techniques were developed and implemented. For example, to reduce the effect of particles' noise on the final response surface, a digital signal filtration technique known as "Savitzky–Golay filter" [39] was applied to ions' fluid moments and potential profiles in the post-analysis procedures, the windows for filtration were tuned at $50\%N_{Cells}$ and $25\%N_{Cells}$ for the ions' moments and plasma potential profiles, respectively; and the filter polynomial orders are 9 and 11. An example of filtered data is shown in Figure 4.8.

In the case where the response surface does not contain zeros (e.g., ϕ_w and $\frac{e\phi_w}{K_B T_e}$), the **ANN** surrogate model was constructed using a general **ANN** architecture, which was built using Keras library [40], an open-source library that provides a Python based interface for **ANN**. The general

[2]Total of 500 data points.

Table 4.2: Summary of the general **ANN** model, showing the model's architecture and hyperparameters details.

Architecture		
Layer(s)	No. Activation Nodes	No. Activation Function
1× Input	2	ReLU
2× Hidden	512	ReLU
1× Output	1	Linear

Hyper Parameters					
Epochs	Loss Function	Optimizer	Learning rate	Momentum	Batch size
2,000	Huber	Adam	5×10^{-4}	0.1	32

ANN model consists of one input layer of shape 2 and two hidden layers with 512 activation nodes each per hidden layer and activation functions of type Rectified Linear Function **ReLU** [41], and one output layer of shape 1 and linear activation function. To ensure the model does not converge to a local minimum momentum of value 0.1 was added. Because the training data set size is relatively small, the learning rate was reduced to 5×10^{-4} and batch size was limited to 32, such setup is important to avoid sudden spikes in the loss function during the training. Finally, to avoid over-fitting and ensure the robustness of the model, the loss function was changed to Huber loss function [42]. Huber loss functions are less sensitive to outliers in the data compared to the **MSE** loss function. A summary of the general **ANN** model's architecture and hyperparameters values are tabulated in Table 4.2.

The problem that arises from the presence of a large number of exactly zero values in a probability distribution is well-known in statistics and usually denoted as the "excess zeros problem" [43]–[45], common techniques for dealing with such distributions in statistics are: the zero-inflated models (i.e., zero-inflated Poisson **ZIP** [46]); and Hurdle model [47]. In general, a zero-inflated model is a probability distribution that allows for frequent zeros in the model. On the other hand, Hurdle model represents the random variable using two parts, the first part is the probability of attaining a value of zero, and the second part models the non-zero values. In the current analysis, variations of both the zero-inflated model and Hurdle model were modified to fit within an **ANN** model. In the modified **ZIP** model, a classifier of type Support Vector Classifier **SVC**, which is a classifier based on **SVM**, was used to separate the zero from the non-zero outputs, where the zero output directly fed to the output layer and the non-zero output are fed to the **ANN** regression model, **SVC** model used a kernel function of type **RBF**. The **ANN** regression model contains one input layer with shape 2 (represents the inputs of the model ($\log(T_e/T_i)$), 8 hidden layers of size 512 each, and activation functions of ReLU, with a loss **MSE** function. The architecture and hyperparameters of the modified **ZIP** are summarized in tables 4.3 and 4.4 for the classifier and the regressor, respectively. The modified **ZIP** was constructed

Table 4.3: Summary of hyper-parameters of the classifier **SVC** sub-model used in the modified **ZIP** model.

Model	Kernel	degree	Tolerance	No. Iterations
SVC	"**RBF**"	7	1×10^{-4}	10,000

Table 4.4: Summary of the regressor **ANN** sub-model used in the modified **ZIP** model, showing the model's architecture and hyperparameters details.

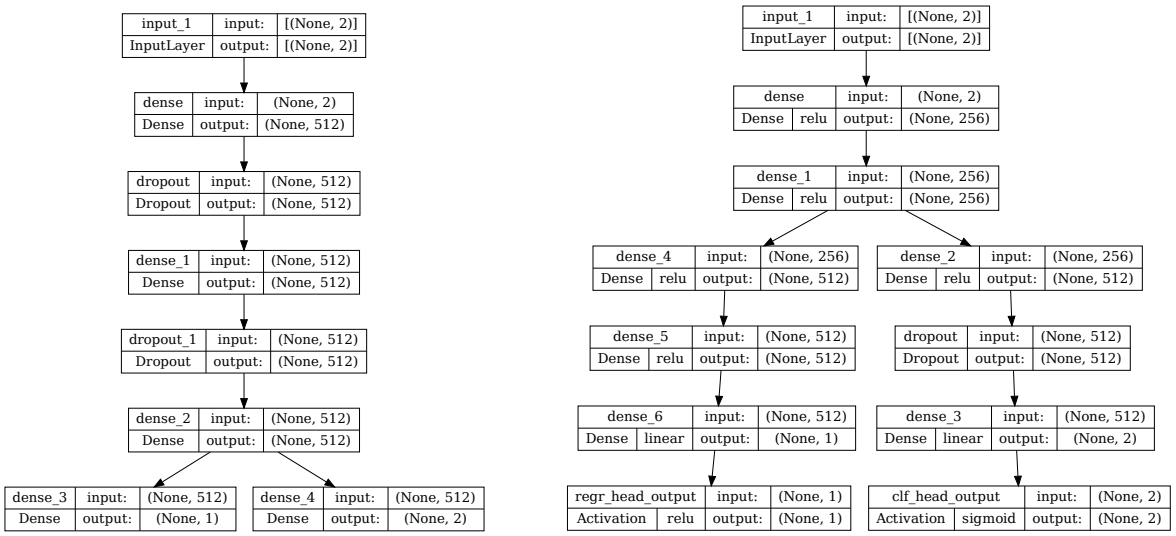
Architecture		
Layer(s)	No. Activation Nodes	No. Activation Function
1× Input	2	ReLU
8× Hidden	512	ReLU
1× Output	1	Linear

Hyper Parameters					
Epochs	Loss Function	Optimizer	Learning rate	Momentum	Batch size
50,000	**MSE**	Adam	1×10^{-4}	None	32

using scikit-lego meta models library [48] with regression and classification models built using scikit-learn (sklearn) library [49]. Huber loss function was not used in the modified **ZIP** model because it is not available in the scikit-learn library.

The main idea of Hurdle **ANN** model is to train two separate classification and regression sub-models simultaneously, where the two sub-models can share all or part of the **ANN** layers. However, each sub-model will have its own loss function and output layer. The main target of the classification sub-model is to predict if the response value is zero or non-zero, and the regression sub-model will predict the actual response value. The final output of the model can either be the prediction of the regressor alone or manually multiply it with the classification output to force zero values. While modified Hurdle and modified **ZIP** may seem to share many similarities, they are essentially different. While the classifier in the modified **ZIP** model filters out the zeros before feeding them to the regression model, Hurdle model regressor takes the data as is. Thus, arguably, the modified **ZIP** model can provide more accurate predictions since no bias caused by the zero values affects the regressor. On the other hand, the zeros bias on Hurdle regressor is not expected to be significant. This can be explained using the fact that the regressor and classifier share part or all of the activation nodes, thus when the regressor sub-model modifies activation nodes weights in the shared part it forces the regressor to achieve higher accuracy near the zero's responses. A detailed comparison of Hurdle and **ZIP** models can be found in [50].

Two modified Hurdle model based **ANN** architectures are proposed and will be referred to them as "Hurdle1" and "Hurdle2". The main difference between the two models is the portion



a) "Hurdle1" b) "Hurdle2"

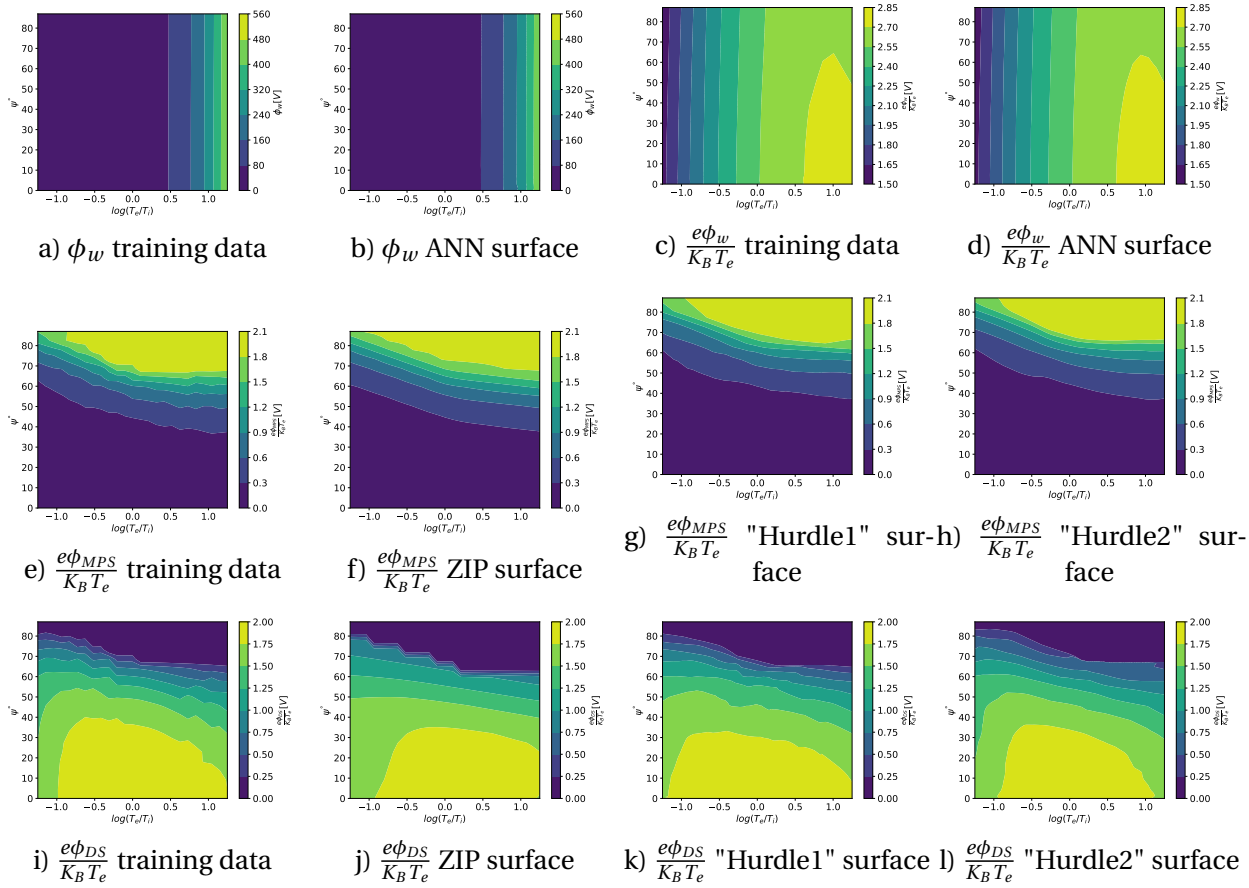
Figure 4.9: A schematic representation of the two variations of Hurdle model architectures developed in the current work. "Hurdle1" shown in a) regressor and classifier shares all layers. In "Hurdle2" shown in b) regressor and classifier shares part layers.

Table 4.5: Hyper parameters summary of the modified Hurdle models.

Parameter	Value/Flag
Regressor Loss Function	Huber loss
Classifier Loss Function	Sparse Categorical Cross Entropy loss
Batch Size	128
Optimizer	Adam
Learning rate	5×10^{-4}
Number of Epochs	15,000

of layers the sub-models share. In the "Hurdle1" model shown in 4.9 a) the regressor and classifier share the entire neural network except for the input and output layers. Whereas in the "Hurdle2" model shown in 4.9 b) the sub-models only share part of the neural network layers, called "bottleneck layers". Dropout layers were added to the models to prevent any possible over-fitting. A summary of Hurdle models hyper-parameters is shown in Table 4.5. Despite the undesired effect of increasing the batch size on convergence, it was important to increase it to 128 to prevent strong biases when the random batch consists of mainly zeros.

Training and testing data sets used for the **ANN**-based models training and testing are shown in Figure 4.1. While the training data set of the least square model could have potentially been used for the **ANN** models, they can lead to a significant error in training since the training data



a) ϕ_w training data b) ϕ_w ANN surface c) $\frac{e\phi_w}{K_B T_e}$ training data d) $\frac{e\phi_w}{K_B T_e}$ ANN surface

e) $\frac{e\phi_{MPS}}{K_B T_e}$ training data f) $\frac{e\phi_{MPS}}{K_B T_e}$ ZIP surface g) $\frac{e\phi_{MPS}}{K_B T_e}$ "Hurdle1" surface h) $\frac{e\phi_{MPS}}{K_B T_e}$ "Hurdle2" surface

i) $\frac{e\phi_{DS}}{K_B T_e}$ training data j) $\frac{e\phi_{DS}}{K_B T_e}$ ZIP surface k) $\frac{e\phi_{DS}}{K_B T_e}$ "Hurdle1" surface l) $\frac{e\phi_{DS}}{K_B T_e}$ "Hurdle2" surface

Figure 4.10: Comparison of potentials training data and **ANN** trained response surfaces. Response surfaces in figures [b), d)] were trained using the general **ANN** model summarized in Table 4.2; figures [f), j)] were trained using the modified **ZIP** model summarized in tables 4.4 & 4.3; and figures [g), k)] & [h), l)] were trained using the two variations of Hurdle model shown in Figure 4.9. To reveal possible over-fitting, response surfaces were created using a fine input grid of size 500×500.

points of a **SG** is denser on the corners of data ranges and more sparse near the center, making such set inappropriate for **ANN** training. Figure 4.10 summarizes the plasma potentials' original data response surfaces and **ANN** trained response surfaces[3]. Where general **ANN** model was applied to the total potential drop and its corresponding dimensionless response surfaces shown in figures 4.10 b) and 4.10 d). On the other hand, "Hurdle1", "Hurdle2", and **ZIP ANN** models were applied to the dimensionless **MPS** and **DS** response surfaces shown in figures 4.10 f)-4.10 h) and 4.10 j)-4.10 l).

Similar to the procedures followed in section 3.3, models' accuracies and predictive capabilities were assessed by evaluating "training error" and "testing errors". Absolute percent errors

[3]Note that response surfaces of the original data in Figure 4.10 is slightly different than the surfaces appears in the **LSF** model in Figure 4.3 as different training data sets were used.

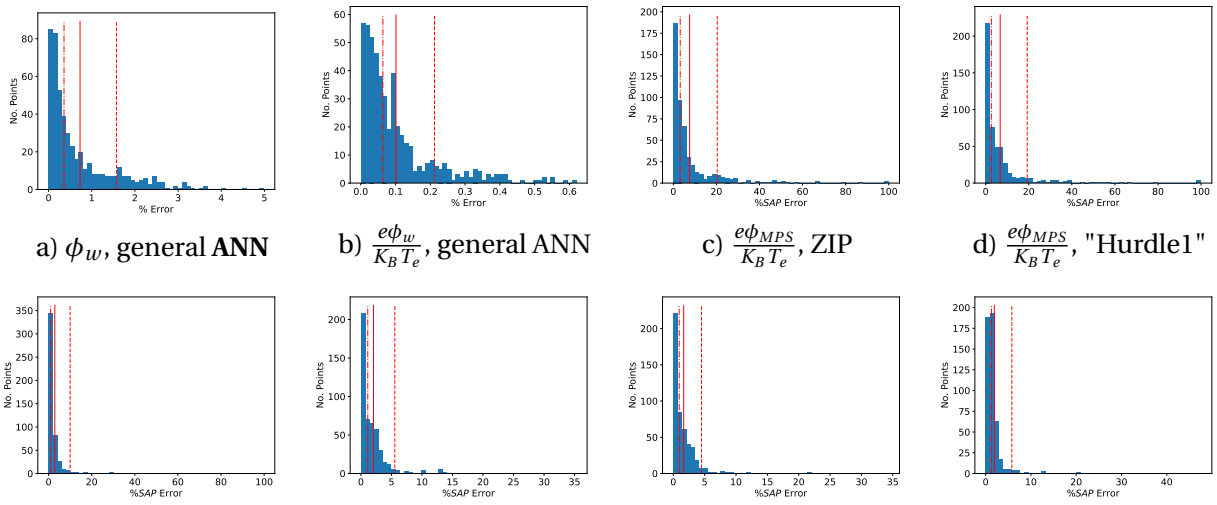
Table 4.6: **ANN** training errors. Normalized Root-Mean-Square Error **NRMSE** was calculated for all models. Absolute percent error was used for ϕ_w models assessments, and **SAMPE** error given by Equation 4.1 was used to assess other models.

RoI	Model	**NRMSE**	Errors		
			Mean	Max	stdev
$\frac{e\phi_w}{K_B T_e}$	General **ANN**	0.42637	0.10121	0.6199	0.1109
ϕ_w	General **ANN**	0.3729	0.7298	4.99	0.8435
$\frac{e\phi_{MPS}}{K_B T_e}$	modified **ZIP**	0.8773	7.56	87.73	12.8
	"Hurdle1"	0.0247	6.759	79	12.542
	"Hurdle2"	0.0121	2.891	50.7	7.093
$\frac{e\phi_{DS}}{K_B T_e}$	modified **ZIP**	0.1925	2.0564	35.297	3.496
	"Hurdle1"	0.0296	1.6586	34.35	2.835
	"Hurdle2"	0.0342	1.9178	47.5	3.868

were computed for the general **ANN** models. And, in the cases of "excess zeros" models, the error is calculated as Systematic Absolute Percent Error **SAPE** given by Equation 4.1, where f is the actual data value and $\tilde{f}$ is the predicted value and $\mathbf{p}$ is the input vector. The Normalized Root-Mean-Square Errors **NRMSE**s were calculated for all training methods. A summary of the "training" normalized root-mean-square errors and statistics of "training errors" are listed in Table 4.6. Histograms of errors' distributions are shown in Figure 4.11. While the training error of the floating wall potential is lower in the least square approximation model as Table 4.1 suggests, training errors of all other potentials' response surfaces were significantly lower when trained using the **ANN** models. Concluding that, in general, **ANN** models provide enhanced accuracy. In addition, Table 4.6 suggests that Hurdle models are more accurate than the modified **ZIP** models. Also, "Hurdle1" showed a lower error when used to train the **MPS** response surface. In contrast, the error was lower when "Hurdle2" was used for the **DS** case.

$$SAPE = \%e(\mathbf{p}) = \frac{|\tilde{f}(\mathbf{p}) - f(\mathbf{p})|}{|f(\mathbf{p})| + |\tilde{f}(\mathbf{p})|} \times 100\% \tag{4.1}$$

For the purpose of having a better insight into the distributions of training errors on the physical parameters space, errors are plotted on the uniform training grid as shown in Figure 4.10. Error grids show higher values on the boundaries of training batches. Such behavior is more visible for ϕ_w training error grid in Figure 4.12 b), where lines of errors surrounding groups of 32 data points (batch size) are visible. Such errors can be reduced by a proper shuffling of training data and using an increased batch size. Also, high-error data points are visible around the zero-valued data point, such errors can be interpreted as artifacts of response surface smoothing



a) ϕ_w, general **ANN** b) $\frac{e\phi_w}{K_B T_e}$, general ANN c) $\frac{e\phi_{MPS}}{K_B T_e}$, ZIP d) $\frac{e\phi_{MPS}}{K_B T_e}$, "Hurdle1"

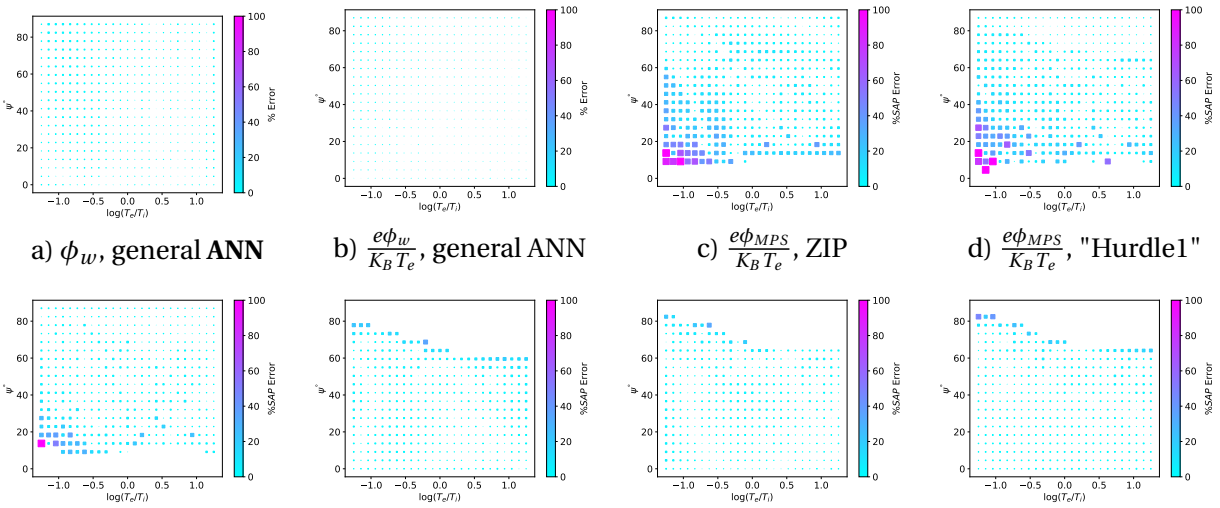
e) $\frac{e\phi_{MPS}}{K_B T_e}$, "Hurdle2" f) $\frac{e\phi_{DS}}{K_B T_e}$, ZIP g) $\frac{e\phi_{DS}}{K_B T_e}$, "Hurdle1" h) $\frac{e\phi_{DS}}{K_B T_e}$, "Hurdle2"

Figure 4.11: Histograms of the "training" errors associated with **ANN** surrogate models' constructions. Captions show the **RoI** along with the models' names. Absolute percent error was used for zero-free models in figures [a), b)]. Systematic absolute percent **SAP** error given by Equation 4.1 was used for the excess zeros problems figures c)-h). The solid lines represent the errors' mean, the dashed-dotted lines are the errors' median, and the dashed lines are one standard deviation from the mean.

procedures.

In order to test the predictive capabilities of the **ANN**-based models, testing errors were calculated for all models and response surfaces. **NRMSE** of testing data as well as testing errors' statistics are listed in Table 4.7. Comparing the testing errors of the **ANN** models and the least square approximation models' testing error in Table 4.1, it can be inferred that the **ANN** models have better predictive capabilities. Similar to training errors, Table 4.7 suggests that "Hurdle2" model has a better performance compared to "Hurdle1" model in the case of **MPS** response surface. Surprisingly, modified **ZIP** model showed excellent predictive capabilities with error's mean, median, and maximum values of 2.3%, 1.03%, and 13.1%, respectively. Detailed distributions' histograms of **ANN** testing errors are shown in Figure 4.13. Testing errors' distributions on testing grid data points are shown in Figure 4.14. Similar to training errors, the testing errors grid shown in Figure 4.13 a) shows an increased error on the boundaries of batches. Errors of the **MPS** models were found to be high near the zero non-zero regions' boundary, away from boundaries "Hurdle2" has a reduced error compared with other **ANN** models.

To select a surrogate model for each response to be used for the sensitivity analysis conducted in chapter 5 a selection criterion based on the following factor was used: (1) Training accuracy, which can be assessed based on training error and the number of outliers. (2) The ability of



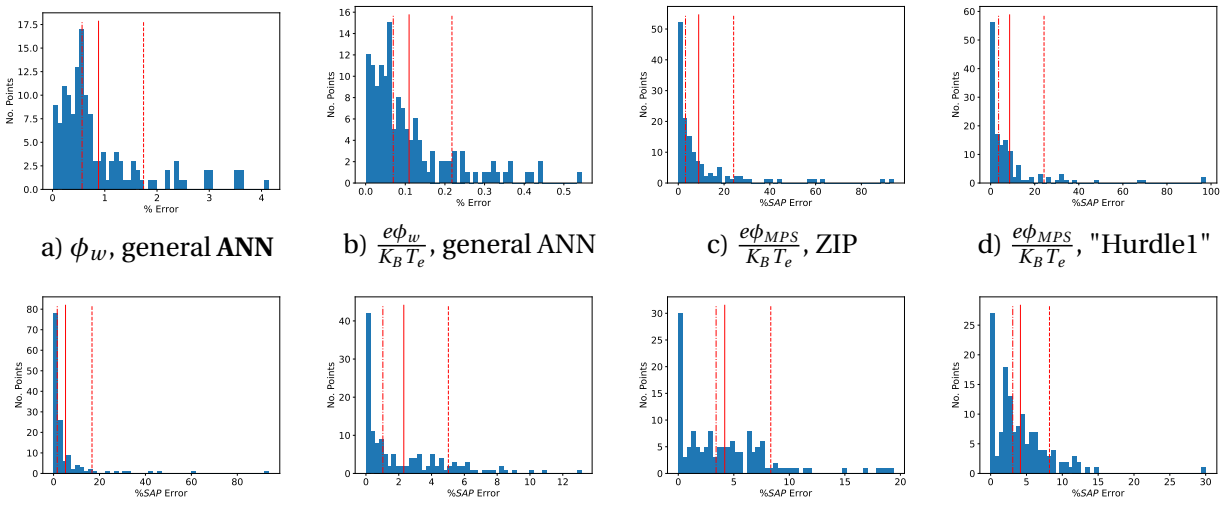
a) ϕ_w, general **ANN** b) $\frac{e\phi_w}{K_B T_e}$, general ANN c) $\frac{e\phi_{MPS}}{K_B T_e}$, ZIP d) $\frac{e\phi_{MPS}}{K_B T_e}$, "Hurdle1"

e) $\frac{e\phi_{MPS}}{K_B T_e}$, "Hurdle2" f) $\frac{e\phi_{DS}}{K_B T_e}$, ZIP g) $\frac{e\phi_{DS}}{K_B T_e}$, "Hurdle1" h) $\frac{e\phi_{DS}}{K_B T_e}$, "Hurdle2"

Figure 4.12: Distribution of **ANN** models' training errors shown in Figure 4.11 on the training grid.

Table 4.7: **ANN** testing errors. Normalized Root-Mean-Square Error **NRMSE** was calculated for all models. Absolute percent error was used for ϕ_w models' assessments, and **SAMPE** error given by Equation 4.1 was used to assess other models.

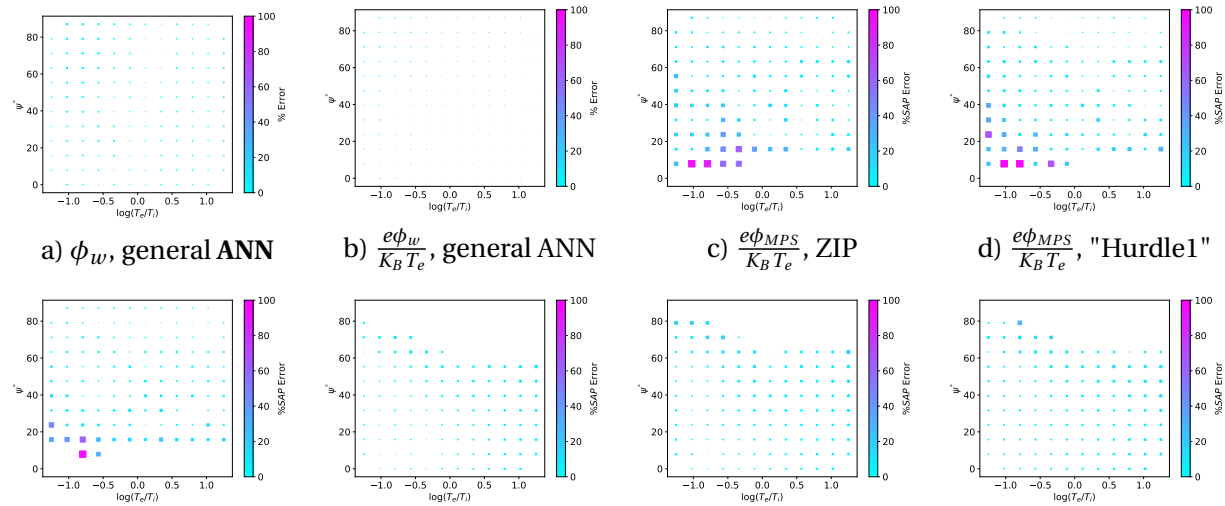
RoI	Model	**NRMSE**	Errors			
			Mean	Max	stdev	median
$\frac{e\phi_w}{K_B T_e}$	General **ANN**	0.44004	0.10962	0.54417	0.10811	0.06965
ϕ_w	General **ANN**	0.39723	0.88058	4.14324	0.86182	0.56529
$\frac{e\phi_{MPS}}{K_B T_e}$	modified **ZIP**	0.03608	8.90772	94.1215	15.2772	3.15342
	"Hurdle1"	0.02792	8.61349	97.7446	15.6129	3.6045
	"Hurdle2"	0.01356	5.20074	93.7956	11.5264	1.5914
$\frac{e\phi_{DS}}{K_B T_e}$	modified **ZIP**	0.0842	2.30791	13.1324	2.71752	1.0317
	"Hurdle1"	0.06748	4.18116	19.3978	4.1471	3.4044
	"Hurdle2"	0.07318	4.15528	30.0551	4.0520	3.0844



a) ϕ_w, general **ANN** b) $\frac{e\phi_w}{K_B T_e}$, general ANN c) $\frac{e\phi_{MPS}}{K_B T_e}$, ZIP d) $\frac{e\phi_{MPS}}{K_B T_e}$, "Hurdle1"

e) $\frac{e\phi_{MPS}}{K_B T_e}$, "Hurdle2" f) $\frac{e\phi_{DS}}{K_B T_e}$, ZIP g) $\frac{e\phi_{DS}}{K_B T_e}$, "Hurdle1" h) $\frac{e\phi_{DS}}{K_B T_e}$, "Hurdle2"

Figure 4.13: Histograms of the "testing" errors associated with **ANN** surrogate models' constructions. Captions show the **RoI** along with the models' names. Absolute percent error was used for zero-free models in figures [a), b)]. Systematic absolute percent **SAP** error given by Equation 4.1 was used for the excess zeros problems figures c)-h). The solid lines represent the errors' mean, the dashed-dotted lines are the errors' median, and dashed lines are one standard deviation from the mean.



a) ϕ_w, general **ANN** b) $\frac{e\phi_w}{K_B T_e}$, general ANN c) $\frac{e\phi_{MPS}}{K_B T_e}$, ZIP d) $\frac{e\phi_{MPS}}{K_B T_e}$, "Hurdle1"

e) $\frac{e\phi_{MPS}}{K_B T_e}$, "Hurdle2" f) $\frac{e\phi_{DS}}{K_B T_e}$, ZIP g) $\frac{e\phi_{DS}}{K_B T_e}$, "Hurdle1" h) $\frac{e\phi_{DS}}{K_B T_e}$, "Hurdle2"

Figure 4.14: Distribution of **ANN** models' "testing" errors shown in Figure 4.13 on the testing grid data points.

Table 4.8: Summary of **ANN** surrogate models construction times in [s]. And prediction time for 250,000 data points in[ms].

RoI	Model	Fitting Time [s]	prediction time [ms]
$\frac{e\phi_w}{K_B T_e}$	General **ANN**	64	~ 1
ϕ_w	General **ANN**	119	~ 1
$\frac{e\phi_{MPS}}{K_B T_e}$	modified **ZIP**	137	60
	"Hurdle1"	320	~ 2
	"Hurdle2"	190	~ 2
$\frac{e\phi_{DS}}{K_B T_e}$	modified **ZIP**	70	18
	"Hurdle1"	700	~ 2
	"Hurdle2"	400	~ 2

the model to mitigate particles' noise effect, which can be assessed by the smoothness of the surface, the location of outliers, and the extent of over-fitting. It should be noted that there is a trade-off between training accuracy and the ability for noise mitigation. For example, if a data point of value zero falls between two non-zero data points, an accurate fitting will be the one that captures the values of the three data points, however, an over-fitting free fitting will predict a non-zero value for the zero data point, leading to a 100% error at that point. (3) Models' predictive capabilities, which can be measured based on testing error. However, one should be cautious when calculating the testing error as the testing data set can potentially be affected by particles' noise. Based on such criteria, the general **ANN** surrogate model of ϕ_w and $\frac{e\phi_w}{K_B T_e}$ will be used for the **SA**. "Hurdle1" model will be used for $\frac{e\phi_{MPS}}{K_B T_e}$. While tables 4.6 and 4.7 show that "Hurdle2" model is more accurate for such a response surface, errors' distribution on the data grid in figures 4.14 d) and 4.14 e) suggests that "Hurdle2" model suffered from over-fitting. Finally, the modified **ZIP** surrogate model will be used for $\frac{e\phi_{DS}}{K_B T_e}$ response surfaces.

Finally, **ANN** models' construction (training) and prediction time are tabulated in 4.8. In general, the **LSF** models' training time was significantly shorter when compared with the **ANN** training time. On the other hand, the prediction time is slightly shorter for the **ANN** models. Table 4.8 shows that the training time of the modified **ZIP** models is generally shorter than Hurdle models, this can be explained by the fact that the training of classification models is more efficient when compared with regression models, and since a smaller amount of data is being passed to the regression model in the modified **ZIP** model, it converges faster. For the time differences in the two Hurdles models, it is a function of the models' complexity as Figure 4.9 shows.

A problem of interest is identifying the region on the parameter space where the Debye sheath disappears. Such a problem was identified by [15]. The classification part of the **ANN**

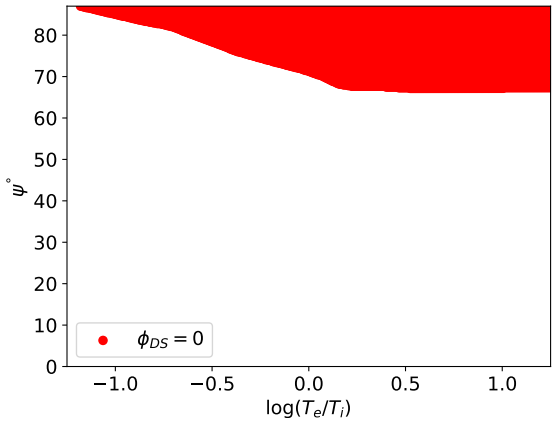


Figure 4.15: Region where Debye sheath disappears as predicted by the **ANN** classification model. The model was trained using **PIC** simulation for plasma with Hydrogen+ ions, electron-to-ion temperature ratios $\log(T_e/T_i)$ in the range $[-1.25, 1.25]$, magnetic field inclination angles with respect to surface normal ψ in the range $[0.0°, 87.0°]$, magnetic field strength of $B = 8\,[T]$, and plasma density of $n = 1 \times 10^{18}\,[m^{-3}]$.

model was used to identify such regions as shown in Figure 4.15.

For reference, the response surfaces of the potential drops across the **MPS** and **DS** are shown in Figure 4.16.

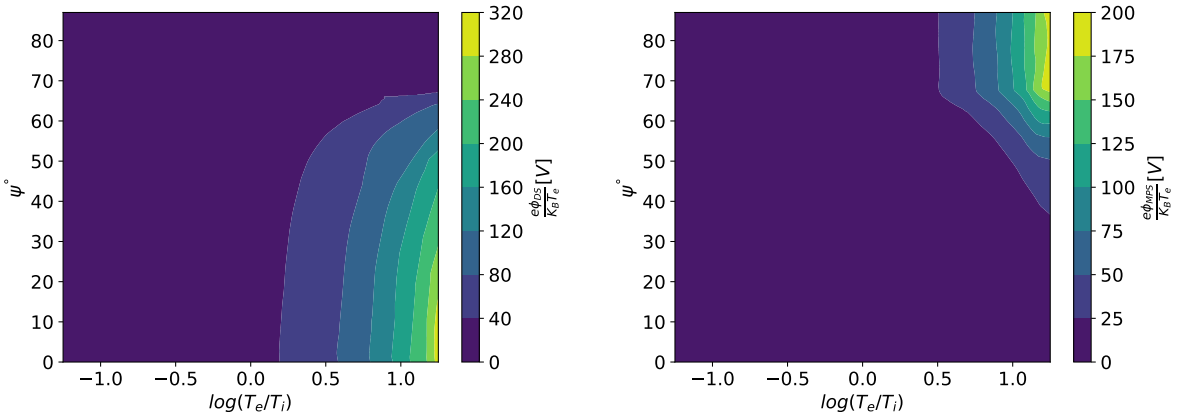
4.4 Main Takeaways

- In this chapter, the procedure of constructing surrogate models for floating wall potential ϕ_w, potential drop across the magnetic pre-sheath ϕ_{MPS}, and potential of the Debye sheath ϕ_{DS} is summarized.

- The main goals of constructing a surrogate model for the plasma potentials is to have a better understanding of plasma sheath structure, to understand its impact on the **IEAD**, and to provide a tool to efficiently compute the plasma's potential drops.

- Two main methods of construction were applied and compared, the least square approximation method introduced in chapter 3, and the artificial neural network **ANN** approach (i.e., Multi-Layer Perceptron **MLP**).

- Two **ANN** approaches for the excess-zero problems were introduced and applied to the construction of plasma potentials surrogate models. These approaches utilized modified versions of common methods in statistics for such problems, namely, zero-inflated Poisson **ZIP** and Hurdle models. The main approach for each model is summarized as follows:



a) Potentials drops across the Debye sheath b) Potentials drops across the magnetic pre-sheath

Figure 4.16: Response surfaces of the potentials drops across both the **DS** and the **MPS** computed using the **ANN** surrogate model.

  - Zero-Inflated Poisson **ZIP**: uses a classifier to filter zero values before passing it to the regressor, or by defining a custom loss function to optimize prediction in the excess zero regions. For the current work supported vector classifier is used.
  - Hurdle: train a regressor and classifier simultaneously, where the regressor and classifier share part or all of the **ANN**'s architecture.

- To overcome challenges associated with using **PIC** generated training data, several strategies were adopted in the **ANN** model construction. Including, using a digital signal filter to reduce the effect of particles' noise, and utilizing Huber loss function to avoid data over-fitting.

- While the artificial neural network surrogate models outperformed the models constructed using the least square approximation method in terms of accuracy and predictive capabilities, it has several drawbacks when compared to the latter. First, it requires a careful selection of hyperparameters. Second, it is more sensitive to particles' noise and more prone to over-fit the training data. Third, construction is several order-of-magnitude more computationally intensive.

- The classifier of the ANN model was used to identify the regions where the Debye sheath disappears, such a problem is important for plasma under the influence of a grazing magnetic field, for example, near the divertor of a Tokamak.

Chapter 5

Sensitivity Analysis (SA) of IEAD's moments and Plasma Potentials

5.1 Introduction

Uncertainty Quantification **UQ** and Sensitivity Analysis **SA** are two essential procedures in scientific computing. In general, **UQ** can be defined as the area of research concerned with identifying, quantifying, and reducing uncertainties associated with models, numerical methods, experiments, and predicting Response of Interest **RoI** [29]. The main goal of **UQ** is to assess the performance of the system when part of the inputs is uncertain. **SA** on the other hand maps the relation between scientific model (computational, empirical, semi-empirical, etc) inputs and the **RoI**. The main goal of **SA** is to identify and rank input parameters with the most significant effect on **RoI**. **UQ** and **SA** are tightly connected. For example, input parameters that have the largest impact on **RoI** and are highly uncertain are expected to dominate the effect on the **RoI** [51]. In general, **SA** has two main classes: Local Sensitivity Analysis **LSA** and Global Sensitivity Analysis **GSA**. **LSA** aims to obtain sensitivity information about a specific point within the input parameter space. Such analysis aims for eliminating the least significant parameters for later **UQ** analysis, or unrevealing the most impact-full input parameters specifically for the inputs of interest. On the other hand, **GSA** explores the entire space of the input parameters [52]. **SA** methods can be classified in many different forms, [53] reviewed some of the methods and classified them as follows: (1) perturbation and derivative methods, (2) multi-starts perturbation methods, (3)correlation and regression analysis methods, (4) regional sensitivity analysis methods, (5) variance-based methods, (6) density-based methods. Method no.1 (perturbation) is utilized in this work for **LSA** in section 5.3.2. Variance-based methods are utilized for **GSA** in section 5.3.1.

PIC simulations often suffer from particle noise and are computationally expensive. Thus,

limited work was found in the literature concerning **UQ** and **SA** on **PIC** codes, and constructing a surrogate model is even more challenging. An example of performing **SA** and utilizing a surrogate mode is [54], in which the authors used Sobols' indices as a method of **SA** to identify the most critical numerical parameters of **PIC** simulations, aiming to reduce the total computational time by eliminating trial-and-error procedure of choosing numerical parameters of a **PIC** simulation used in the context of particle accelerator systems. In their analysis, a surrogate based on Polynomial Chaos Expansion **PCE** was constructed. Using **PCE** to construct the surrogate model can reduce the cost of computing Sobols' indices. Also, [55] performed **SA** in the context of chemical reactor engineering using the **PIC** code MFiX-PIC. On the other hand, **UQ** is usually used in validation and verification procedures. For example [56] performed parametric **UQ** as a part of the validation and benchmark of two **PIC** codes; EDIPIC [57] and LSP [58], applied to a short parallel-plate glow discharge. Also, [59] applied **UQ** as a part of **PIC** validation and verification. They identified two sources of uncertainties: the uncertainty related to particle noise (numerical error) and parametric uncertainty of the **PIC** inputs. To estimate the numerical error, they used a multi-fitting scheme known as Stochastic Richardson Extrapolation-based Error Quantification **StREEQ** [60]. Such a method was developed specifically for numerical error estimation for stochastic codes. [61] explored the propagation of the noise in a **PIC** cycle by performing **GSA** of the parameters at each stage of the cycle at different numerical input parameter sets.

A variance-based method, known as Sobols' indices, was used to perform global sensitivity analysis on two main problems, **IEAD**s' moments as well as potentials' drop across the plasma domain. The **IEAD** surrogate model constructed in chapter 3 was employed to supply the samples necessary for sensitivity indices computations. Sensitivity indices of **IEAD** moments, namely, ions' mean energy of impact, ions' mean impact angle, and their associated standard deviations were computed with respect to the most influential parameters, electrons-to-ions temperature ratios T_e/T_i, and magnetic field inclination angle ψ. Then the analysis was extended to include additional parameters: magnetic field strength B and plasma density n. Similarly, the **ANN**-based surrogate models of plasma potentials were used to compute the effects of T_e/T_i and ψ on floating wall potential, the potential drop across the **DS**, and **MPS**. In addition, Local Sensitivity Analysis **LSA** was performed on **WEST** like conditions using One at a Time **OAT** method. The goal of the **LSA** is to explore the effect of an extended parameter space on the responses of interest.

Section 5.2 reviews the methods used for the **GSA** and **LSA**. The results of the **GSA** and **LSA** are reported in sections 5.3.1 and 5.3.2, respectively.

5.2 Methodologies

5.2.1 Sobol's Indices

Among the classes of methods available for **GSA**, a variance-based method known as Sobols' indices has been adopted in this work [52]. The basic idea is to decompose the variance of **RoI** into fraction contributions from inputs. Choosing Sobols' indices can be justified by: (i) simplicity of the method, the method is straightforward to implement and interpret since the output appears as a percentage of input contribution to the output of **RoI**. (ii) The method allows for interaction between the inputs. iii. This class of methods can deal with non-linear **RoIs**.

First order Sobols' indices (also known as "main effect indices") measure the direct contribution of each input toward the final **RoI**, without taking into account the interaction between the input parameters. Given a response function y for input vector $\mathbf{p}$ with n inputs, such that $y(\mathbf{p}) : \mathbb{R}^n \rightarrow \mathbb{R}$ such that $\mathbf{p} \in \mathbb{R}$, then first order Sobol index is given by Equation 5.1. In which, $\mathbb{V}(y)$ is the variance of y and $\mathbb{V}\left[\mathbb{E}\left(y \mid p_i\right)\right]$ is the variance of the conditional expectation of y given the fixed input parameter p_i.

$$S_i := \frac{\mathbb{V}\left[\mathbb{E}\left(y \mid p_i\right)\right]}{\mathbb{V}(y)}, \quad i = 1, ..., n \tag{5.1}$$

The second-order Sobols' indices (also known as "second-order interaction"), measure the effect of varying two parameters (p_i, p, j) simultaneously, along with their individual effect, and is defined by Equation 5.2. Such that $\mathbb{V}\left[\mathbb{E}\left(y \mid p_i, p_j\right)\right]$ is the conditional expectation of y given the fixed input parameters p_i, p_j. The second-order indices are usually represented in a 2×2 matrix. Higher order indices follow the same definition of 5.2, however, have not been used in this analysis. In practice, the number of higher order indices scales with $2^n - 1$. Accordingly, if the number of input parameters is large, higher order indices become computationally extensive and one can use the total effect indices instead.

$$S_{ij} := \frac{\mathbb{V}\left[\mathbb{E}\left(y \mid p_i, p_j\right)\right]}{\mathbb{V}(y)} - S_i - S_j, \quad i, j = 1, ..., n, \quad i \neq j \tag{5.2}$$

Finally, the total effect indices measure the contribution towards **RoI** using variances of the p_i, including all variances caused by interactions of any order with any other input parameter. Total effect variance is defined by Equation 5.3. Such that, $\mathbb{E}\left(y \mid \mathbf{p}_{\sim i}\right)$ is the conditional expectation of y for all fixed parameters except for p_i. Total effect indices are the most informative among the order of Sobols' indices.

$$S_{T_i} := 1 - \frac{\mathbb{V}\left[\mathbb{E}\left(y \mid \mathbf{p}_{\sim i}\right)\right]}{\mathbb{V}(y)}, \quad i = 1, ..., n \tag{5.3}$$

It should be noted that Sobols' indices are a uni-variant method and the indices for a two-dimensional response (e.g., IEAD) can be evaluated by computing Sobols' indices pixel-wise across all input parameters inputs, similar to Figure 3.5.

For the 2D cases, relations between different orders of Sobol indices are summarized in equations 5.4 & 5.6. Such a simple relation exists because in 2D first and second order indices can fully describe sensitivity response and the total effect indices are redundant information.

$$S_1 + S_2 + S_{12} = 1, \tag{5.4}$$

$$S_{T_1} = S_1 + S_{12} = 1 - S_2, \tag{5.5}$$

$$S_{T_2} = S_2 + S_{12} = 1 - S_1, \tag{5.6}$$

In general, the first-order indices and higher order indices can be related according to Equation 5.14, so when the interaction is minimal, the first-order indices are expected to add up to unity. If the sum exceeded unity then an error/noise is affecting the data and more samples are needed. On the other hand, the total effect indices can exceed unity as can be read in Equation 5.15. If the sum of the total effect indices is unity, then the model is purely additive and no significant interaction between the inputs is present.

The computations of Sobols' indices are as follows: To compute Sobols' indices, N samples of n quasi-random input values[1] have been drawn from the physical input space. The sampling technique of choice was Sobol's sequence[2] with $n = 4$ for the 2D cases and $n = 8$ for the 4D cases. Using Sobol's sequence to sample from the input space is attractive because it can sample $N \times 2d$ list efficiently, with d being the number of input parameters. In addition, Sobol's sequence ensures a low discrepancy random sample [62]. It should be indicated that Sobol's sequence can only sample a number of samples of base 2 and using MATLAB *'qrandstream'* will truncate the sequence to N, leading to a slight increase in the discrepancy of the random sample.

Sobols' indices computation procedure as described by [52], and summarized as follows: First step is to generate a $N \times 2n$ matrix of random numbers M within the range of the input parameters, such that, $M \in \mathbb{R}^{N \times 2n}$. With N is the sample size and n is the number of input parameters (In the current analysis $n = 2, 4$). Such that, the i^{th} and $(n+i)^{th}$ element corresponds to the i^{th} parameter.

[1] Not to be mixed with pseudo-random numbers.
[2] Not to be mixed with Sobol indices.

$$M = \begin{bmatrix} p_1^{(1)} & p_2^{(1)} & \cdots & p_n^{(1)} & p_{n+1}^{(1)} & \cdots & p_{2n}^{(1)} \\ p_1^{(2)} & p_2^{(2)} & \cdots & p_n^{(2)} & p_{n+1}^{(2)} & \cdots & p_{2n}^{(2)} \\ \vdots & \vdots & \cdots & \vdots & \vdots & \cdots & \vdots \\ p_1^{(N)} & p_2^{N} & \cdots & p_n^{(N)} & p_{n+1}^{(N)} & \cdots & p_{2n}^{(N)} \end{bmatrix} \tag{5.7}$$

Matrix M can be rewritten as two matrices of $N \times n$ elements such that $A \in \mathbb{R}^{N \times n}$ and $B \in \mathbb{R}^{N \times n}$ as follows:

$$A = \begin{bmatrix} p_1^{(1)} & \cdots & p_n^{(1)} \\ \vdots & \ddots & \vdots \\ p_1^{(N)} & \cdots & p_n^{(N)} \end{bmatrix}, \qquad B = \begin{bmatrix} p_{n+1}^{(1)} & \cdots & p_{2n}^{(1)} \\ \vdots & \ddots & \vdots \\ p_{n+1}^{(N)} & \cdots & p_{2n}^{(N)} \end{bmatrix} \tag{5.8}$$

Matrices A & B are then used to construct n additional matrices $C_i \in \mathbb{R}^{N \times n}$, $i = 1, \cdots, n$. Matrices C_i are constructing based on matrix B with the i^{th} column replaced by elements from matrix A:

$$C_i = \begin{bmatrix} p_{n+1}^{(1)} & \cdots & p_i^{(1)} & \cdots & p_{2n}^{(1)} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ p_{n+1}^{(N)} & \cdots & p_i^{(N)} & \cdots & p_{2n}^{(N)} \end{bmatrix} \tag{5.9}$$

The response $y(\mathbf{P})$ of each input parameters vector (i.e., row) in the matrices A, B, and C_i is evaluated (either by simulations or using a surrogate model), leading to $(2+n)$ $N \times 1$ response vectors:

$$\mathbf{y}_A = y(A) := \begin{bmatrix} y\left(\mathbf{P}^{(1)}\right) \\ y\left(\mathbf{P}^{(2)}\right) \\ \vdots \\ y\left(\mathbf{P}^{(N)}\right) \end{bmatrix}, \qquad \mathbf{y}_B = y(B) := \begin{bmatrix} y\left(\bar{\mathbf{P}}^{(1)}\right) \\ y\left(\bar{\mathbf{P}}^{(2)}\right) \\ \vdots \\ y\left(\bar{\mathbf{P}}^{(N)}\right) \end{bmatrix}, \qquad \mathbf{y}_C = y(C) := \begin{bmatrix} y\left(\bar{\mathbf{P}}^{(1)}|_i\right) \\ y\left(\bar{\mathbf{P}}^{(2)}|_i\right) \\ \vdots \\ y\left(\bar{\mathbf{P}}^{(N)}|_i\right) \end{bmatrix}, \tag{5.10}$$

Where $\mathbf{p}^{(j)} := \left(p_1^{(j)}, p_2^{(j)}, \cdots, p_n^{(j)}\right)$, $\bar{\mathbf{p}}^{(j)} := \left(\bar{p}_{n+1}^{(j)}, \bar{p}_{n+2}^{(j)}, \cdots, \bar{p}_{2n}^{(j)}\right)$ and $\bar{\mathbf{p}}^{(j)}|_i := \left(\bar{p}_{n+1}^{(j)}, \bar{p}_{n+2}^{(j)}, \cdots, p_i^j, \cdots, \bar{p}_{2n}^{(j)}\right)$ with the i^{th} entry replaced by $p_i^{(j)}$ from matrix A.

After computing the responses of matrices A, B and C, the first and total order Sobols' indices can be evaluated. First order Sobol's index with respect to the i^{th} input parameter is given by:

$$S_i = \frac{(1/N)\sum_{j=1}^{N} y_A^{(j)} y_{C_i}^{(j)} - f_0^2}{(1/N)\sum_{j=1}^{N} \left(y_A^{(j)}\right)^2 - f_0^2}, \quad i = 1, \cdots, n, \tag{5.11}$$

With f_0^2 given by:

$$f_0^2 := \left(\frac{1}{N} \sum_{j=1}^{N} y_A^{(j)} \right)^2 \tag{5.12}$$

Similarly, total order Sobol's index with respect to the i^{th} input parameter can be calculated by:

$$S_{T_i} = 1 - \frac{(1/N) \sum_{j=1}^{N} y_B^{(j)} y_{C_i}^{(j)} - f_0^2}{(1/N) \sum_{j=1}^{N} \left(y_A^{(j)} \right)^2 - f_0^2}, \quad i = 1, \cdots, n, \tag{5.13}$$

Sample size used for the current analysis was $N = 50,000$ as discussed in section 5.3.1. Based on this method, the number of simulations needed for the **GSA** is $N(2+n)$, with $n = 2$ for the 2D cases and $n = 4$ for the 4D case, the total number of simulations required to be run are $200,000$ and $300,000$, respectively. Thus, a surrogate model was a necessity.

$$\sum_{i=1}^{d} S_i + \sum_{i<j}^{d} S_{ij} + \cdots + S_{12\ldots d} = 1 \tag{5.14}$$

$$\sum_{i=1}^{d} S_{T_i} \geq 1 \tag{5.15}$$

While the Sobols' indices were hardly coded for most of the work, for some analysis the sensitivity analysis library in python **SALib** was used [63], [64] due to its parallel capabilities, which allowed for more extensive analysis.

5.2.2 One At a Time (OAT)

While **GSA** ranks the influence of parameters on the **RoI** over a wide range of inputs, **LSA** is concerned with a limited input range, usually around one point of interest. The motivations to perform **LSA** in the current analysis are summarized as follows: To get an insight about the parameters that will be included in future **GSA**, especially in the case of multi-species plasma, to have a more profound understanding of **RoI**s' sensitivities for input parameters from an actual magnetic confinement device, and to determine which parameters should be included in future **UQ** analysis (i.e., parameters that are both highly uncertain and sensitive).

LSA was applied to a tungsten divertor in steady-state of **WEST** [65]. Plasma parameters of $He+/He++$ discharges and from probe data obtained at 5 points along the Tokamak divertor were used as inputs. The method adopted for **LSA** is One At a Time **OAT** local sensitivity analysis method (i.e., also known as linear perturbation theory), which is the most common method

for local sensitivity analysis [3]. **OAT** ranks the relative influence of the i^{th} input parameter x_i on the **RoI** denoted as f. Then the sensitivity index S, for each input parameter is calculated by evaluating the partial derivative of the **RoI** concerning each input parameter as shown in Equation 5.16 [66], [52].

$$S_i^{OAT} = \frac{\partial y}{\partial x_i}, i = 1, 2, ..., n \tag{5.16}$$

Since an analytical solution is not available for the **RoI**s. Finite difference methods were used to calculate the partial derivatives numerically. In this work, forward 5.17 and central 5.18 finite difference methods were used and compared. Where, in equations 5.17 & 5.18, h is the perturbation value of the i^{th} input parameter and f is the **RoI**. h has been estimated based on the expected uncertainties of the measurement. In order to ensure dimensionless values of S, and to facilitate the comparison of sensitivity indices, it is common to normalize equations 5.17 and 5.18 as shown in equations 5.19 and 5.20. Although it is common to use different perturbation values for each finite difference scheme. It has been fixed in this analysis for simplicity. Table 5.1 summarize perturbations values for each physical parameter of interest used for the **OAT** analysis.

$$f^{'}(x) = \frac{f(x+h) - f(x)}{h} + \mathcal{O}(h) \tag{5.17}$$

$$f^{'}(x) = \frac{f(x+h) - f(x-h)}{2h} + \mathcal{O}(h^2) \tag{5.18}$$

$$S_{i,norm}^{OAT} = \frac{(f(x_i+h) - f(x_i))/f(x_i)}{h/x_i} \tag{5.19}$$

$$S_{i,norm}^{OAT} = \frac{(f(x_i+h) - f(x_i-h))/f(x_i)}{2h/x_i} \tag{5.20}$$

5.3 Results

5.3.1 Global Sensitivity Analysis (GSA) of IEAD's Moments and plasma potentials

The Global Sensitivity Analysis **GSA** method described in section 5.2.1 was applied to two problems, moments of the **IEAD**s problem described in chapter 3, and the plasma potentials problem

[3] It is the **LSA** version of Morris screening method

Table 5.1: Chosen perturbation values for **WEST** physical parameters used for **OAT** sensitivity analysis.

Parameter	**Perturbation**
B (Magnetic field) $[T]$	0.5
ψ (Magnetic angle) [Degree]	3
Te (Electrons' Temperature) $[eV]$	10%
Ti (Ions' Temperature) $[eV]$	10%
He++ Density $[n/m^3]$	One order of magnitude
He+ Density $[n/m^3]$	One order of magnitude

introduced in chapter 4. The goal of the analysis is to unravel the effects of physical parameters (i.e., electron-to-ion temperature ratio T_e/T_i, plasma density n, magnetic field strength B, and magnetic field inclination ψ) on the ions' energies and angles moments (i.e., mean and variance) at **PSI** of a Hydrogen plasma, for the first problem. And to compute the effects of physical parameters (i.e., electron-to-ion temperature ratio T_e/T_i, and magnetic field inclination angle ψ) on plasma potential for the second problem. More specifically compute the effect of physical parameters on floating wall potentials (ϕ_w, $\frac{e\phi_w}{K_B T_e}$), the potentials' drop across the **MPS** (ϕ_{MPS}, $\frac{e\phi_{MPS}}{K_B T_e}$), the potentials' drop across the **DS** (ϕ_{DS}, $\frac{e\phi_{DS}}{K_B T_e}$), and the effect on normalized potential drops in the **MPS** and **DS** with respect to the total potential drops ($\frac{\phi_{MPS}}{\phi_w}$, $\frac{\phi_{DS}}{\phi_w}$).

The general workflow of the **GSA** follows is shown in Figure 3.2, while the workflow shows the procedures for the **IEAD**, the method remains the same for the plasma potentials problem. The workflow is summarized as follows. Following input space sampling, the **RoI**s for both, moments of ions' energies and angles, and plasma potentials were evaluated using the **IEAD** surrogate model described in chapter 3, and the plasma potentials surrogate model described in chapter 4. Then Sobols' indices were evaluated along with the associated indices' errors. More samples were added until the maximum error of the Sobols' indices is within the "acceptable" range. Table 5.2 shows the errors of Sobols' indices for both 2D and 4D cases computed for the **IEAD**'s moments. At $N = 50,000$ the error is 0.02% for the 2D cases and 0.35% for the 4D cases, thus, for later analysis the number of samples used in Sobols' indices calculations was fixed at $N = 50,000$.

First and second-order Sobols' indices[4] of plasma potentials (with variable electron-to-ion temperature ratios T_e/T_i and the magnetic field inclination angles ψ) are shown in Figure 5.1 and the values of the indices are tabulated in Table 5.3, where S1 denotes the first order effect with respect to electrons-to-ions temperature ratios T_e/T_i, S2 denotes the first order effect with respect to magnetic field inclination angle with respect to surface normal ψ, and S12 denoted the second order sensitivity index, showing the interactions of the input parameters.

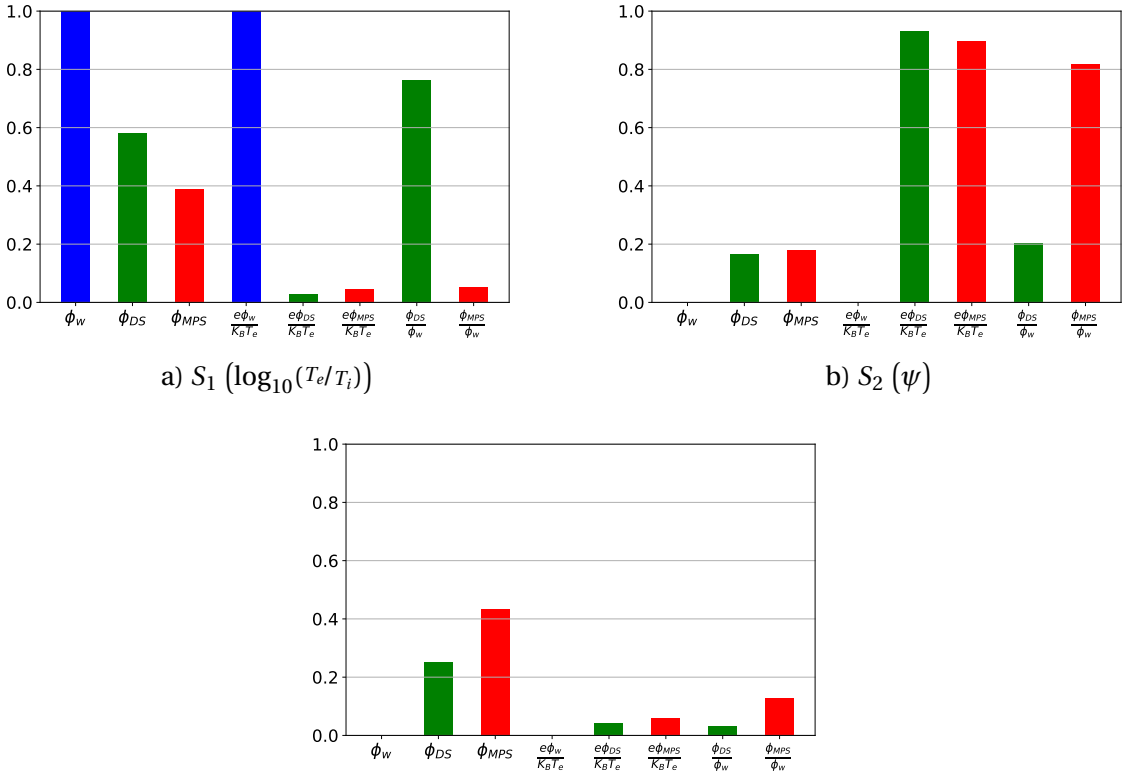
[4]Recall that the total index in for two dimensional problems is the sum of first and second order indices as given by Equation 5.4.

Table 5.2: Maximum error of Sobol indices for a different N number of samples. Error for each number of samples was evaluated with respect to the previous sample size, with $N = 100$ have been used as a reference sample size. The error in the 2D cases is that of the first-order Sobol indices, whereas the error in the 4D cases is that of the first and total Sobol indices.

N	2D Max. Error	4D Max. Error
500	0.0324	0.2266
1000	0.0123	0.0409
2000	0.0126	0.0419
5000	0.0068	0.0136
10,000	0.0013	0.0081
20,000	0.0012	0.0031
50,000	0.0002	0.0035

Table 5.3: First and second orders Sobols' indices for plasma potentials, ranking the influence of physical parameters $\left(\log_{10}(T_e/T_i), \psi\right)$ on floating wall potentials (ϕ_w, $\frac{e\phi_w}{K_B T_e}$), the potentials' drop across the **MPS** (ϕ_{MPS}, $\frac{e\phi_{MPS}}{K_B T_e}$), the potentials' drop across the **DS** (ϕ_{DS}, $\frac{e\phi_{DS}}{K_B T_e}$), and the effect on normalized potential drops in the **MPS** and **DS** with respect to the total potential drops ($\frac{\phi_{MPS}}{\phi_w}$, $\frac{\phi_{DS}}{\phi_w}$). Sensitivity indices were computed using a large number of samples $N = 50,000$. Responses were computed using the surrogate model developed in chapter 4.

Sobol's Index	Parameter	Potential							
		ϕ_w	ϕ_{DS}	ϕ_{MPS}	$\frac{e\phi_w}{K_B T_e}$	$\frac{e\phi_{DS}}{K_B T_e}$	$\frac{e\phi_{MPS}}{K_B T_e}$	$\frac{\phi_{DS}}{\phi_w}$	$\frac{\phi_{MPS}}{\phi_w}$
S1	$\log_{10}(T_e/T_i)$	1	0.5824	0.388	0.9989	0.0277	0.0453	0.7634	0.0534
S2	ψ	0	0.1672	0.1794	0.0011	0.93	0.8969	0.2045	0.8182
S12	$\left(\log_{10}(T_e/T_i), \psi\right)$	0	0.2505	0.4326	0.0001	0.0423	0.0579	0.0322	0.1292

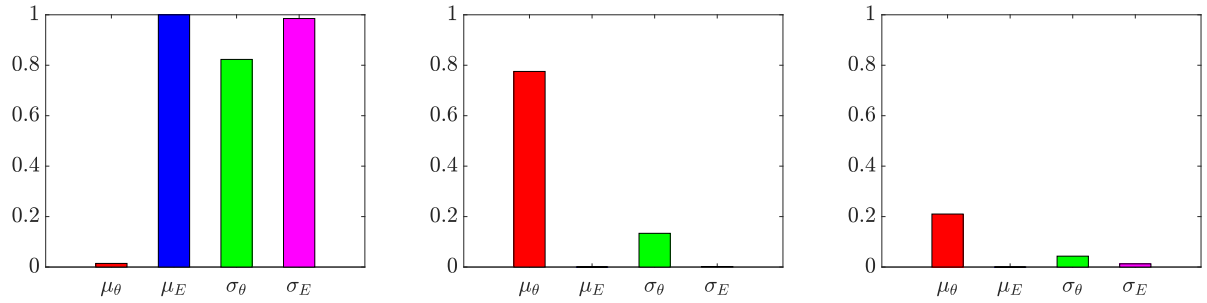


a) $S_1\left(\log_{10}(T_e/T_i)\right)$

b) $S_2\left(\psi\right)$

c) $S_{12}\left(\log_{10}(T_e/T_i), \psi\right)$

Figure 5.1: First and second orders Sobols' indices for the plasma potentials; showing the effect of electron-to-ion temperature ratios T_e/T_i (a)), magnetic field inclination angles ψ (b)) and the combined effect (second/total effect Sobols' indices) (c)) on plasma potential of a Hydrogen plasma. Using a number of samples $N = 50,000$. **RoI**s were computed using the surrogate models developed in chapter 4.



a) S_1 $\left(\log_{10}(T_e/T_i)\right)$ b) S_2 $\left(\psi\right)$ c) S_{12} $\left(\log_{10}(T_e/T_i), \psi\right)$

Figure 5.2: First and second orders Sobols' indices for the 2D cases; showing the effect of electron-to-ion temperature ratios T_e/T_i (a)), magnetic field inclination angles ψ (b)), and the combined effect (second/total effect Sobols' indices) (c)) on ions' energies and angles moments at **PSI** of a Hydrogen plasma. Using a number of samples $N = 50,000$.

Figure 5.1 a) shows the effect of electron-to-ion temperature ratios T_e/T_i on the different potential drop values. The floating wall potential is 100% affected by T_e/T_i as Equation 2.13 reads. Potential drops across **MPS** and **DS** does add up to a value very close to unity, suggesting that in some cases Equation 2.22 does not apply, and the sum of potentials exceeds unity. When computing S1 for the dimensionless cases, T_e/T_i has little effect on **MPS** and **DS**, since they are no longer a function of T_i and only ψ has a significant effect as Equation 2.21 suggests. A strong interaction between the input parameters was observed for **MPS** and **DS** in Figure 5.1 c) since both of them are a function of T_e, with a much lower effect observed for their dimensionless values as the effect of T_e is neutralized, with the remaining effect is believed to occurs in the regions where the **DS** does not exist. In addition, little to no effect on ϕ_w was correlated to ψ as Figure 5.1 b). Finally, the effects of the normalized **DS** and **MPS** values will be used to explain the Sobols' indices on the **IEAD**s' moments.

First and second-order Sobols' indices of the 2D cases (with variable electron-to-ion temperature ratios T_e/T_i and the magnetic field inclination angles ψ) are shown in Figure 5.2 and the total effect Sobols' indices values for 2D and 4D cases are tabulated in Table 5.4. Recall that for the 4D cases, in addition to the 2D parameters, the effect of magnetic field strength B and plasma density n were added. 4D first order Sobols' indices for the ions' energies and angles moments are shown in Figure 5.3 and the total effect indices are shown in Figure 5.4.

From Figure 5.2, it can be noted that the electron-to-ion temperature ratios T_e/T_i significantly affect the mean ions' energy and the spread of the energy (can be interpreted as ions' temperature) at **PSI**. This agrees with Equation 2.13 which governs the total potential drop from the bulk plasma to the wall surface. In more detail, as T_e/T_i increases $\frac{e\phi_w}{K_B T_e}$ increases, thus, increasing the energy of ions impacting the wall and decreasing its spread as the sheath accelerates the ions toward the wall to get a beam-like structure (sheath cooling effect). This also agrees with Figure

Table 5.4: Total effect Sobols' indices for 4D cases ranking the influence of T_e/T_i, n, B and ψ on **IEAD**'s moments. Indices for 2D cases are provided for comparison. Indices were computed using a number of samples $N = 50,000$.

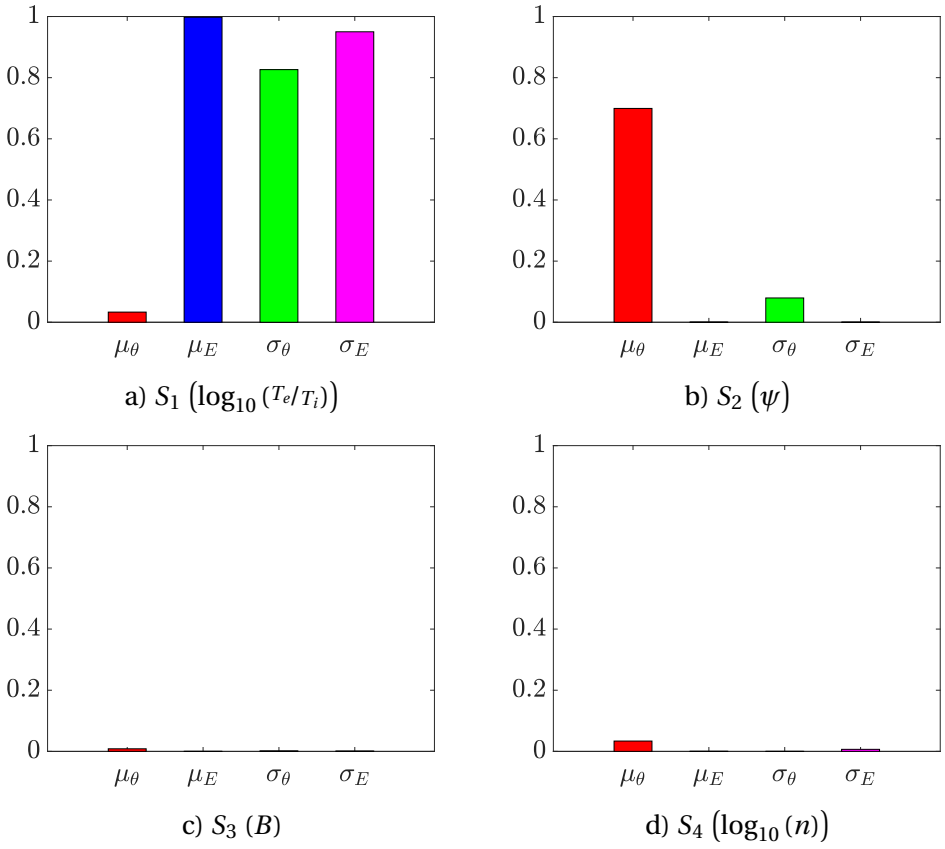
Sobol index	S_{T_1}		S_{T_2}		S_{T_3}	S_{T_4}
Parameter	$\log_{10}(T_e/T_i)$		ψ		B	$\log_{10}(n)$
Cases	2D	4D	2D	4D	4D	4D
Moments						
μ_θ	0.22	0.24	0.99	0.90	0.02	0.07
μ_E	1.00	1.00	0.0	0.0	0.0	0.0
σ_θ	0.87	0.88	0.18	0.17	0.01	0.04
σ_E	1.00	0.99	0.01	0.01	0.01	0.04

5.1 where it can be noted that the mean and standard deviation of ions' energy at the **PSI** are directly correlated with the total potential drop in plasma. Comparing figures 5.2 and 5.1 it can be noted that the mean angle of impact follows the same correlation behavior as $\frac{\phi_{MPS}}{\phi_w}$ indicating that the mean angle of impact is a function of the portion of the **MPS** drops to the total plasma's potential drop. Similarly, the spread of the impact angle can be correlated with the **DS** portion of the potential drop.

First and total orders Sobols' indices for **IEAD**s' moments in the 4D cases are shown in figures 5.3 and 5.4. This shows similar dependencies of the ions' energy and angle moments as the 2D cases, also reviles a small dependency on plasma density n and almost negligible dependency on magnetic field strength. To gain a deeper understanding of the nature of variation in ions' energy and angle moments with respect to the most influential input parameters (i.e., 2D parameters, electron-to-ion temperature rations T_e/T_i, and magnetic field inclination angle ψ.), the dependencies of **IEAD**s moments on the physical parameters space are plotted in Figure 5.5.

Figure 5.5 shows a strong dependency of **IEAD** means energies μ_E on electrons-to-ions temperature ratios. This can be fully explained using Equation 2.13. Total potential drop ϕ_w is proportional to T_e and $\ln(T_e/T_i)$, thus, the acceleration of ions toward the wall will increase as T_e increases. Note that the electrons-to-ions temperature ratio has been varied by keeping the ions' temperature T_i constant and changing the electrons' temperature T_e. Accordingly, most of the effect on μ_E is in fact due to T_e. The behavior of ions' energy at the **PSI** is also in good agreement with the response surface of the floating wall potential ϕ_w shown in Figure 4.10 b).

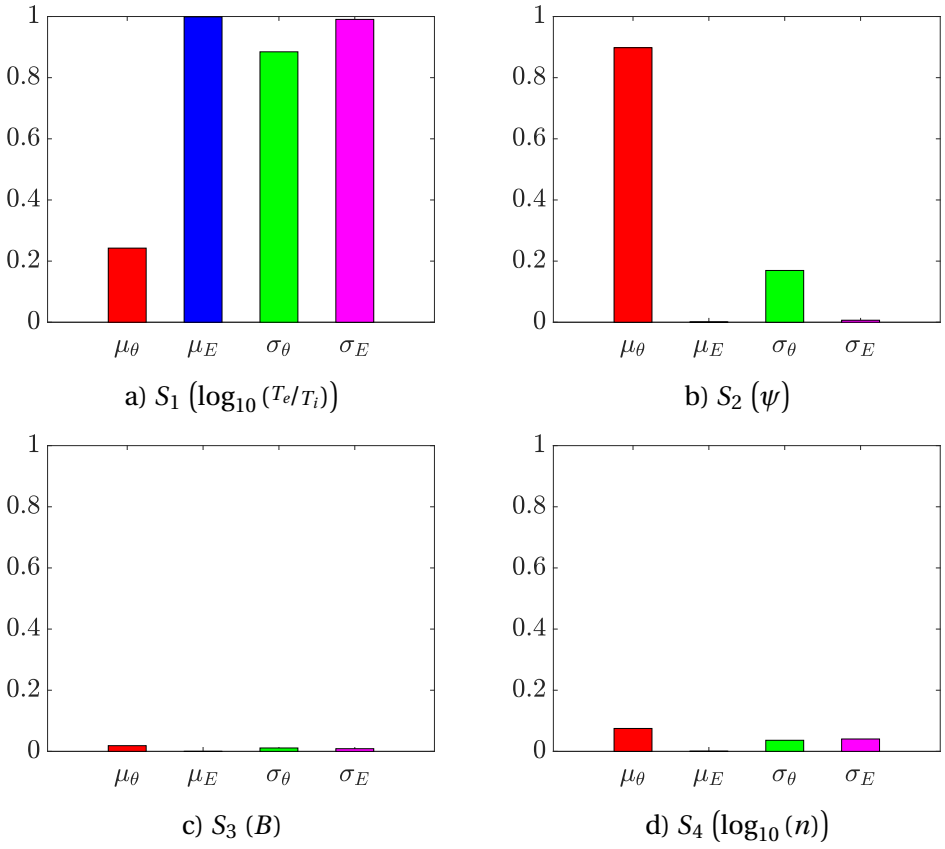
Figure 5.5 b) shows that the energy spread (represented by standard divination of energies σ_E) is directly affected by the electrons-to-ions temperature ratios T_e/T_i with a slight effect from the magnetic field inclination angle ψ. Such an effect is more obvious at high temperature ratios.



a) S_1 $\left(\log_{10}\left(T_e/T_i\right)\right)$ b) S_2 $\left(\psi\right)$

c) S_3 $\left(B\right)$ d) S_4 $\left(\log_{10}\left(n\right)\right)$

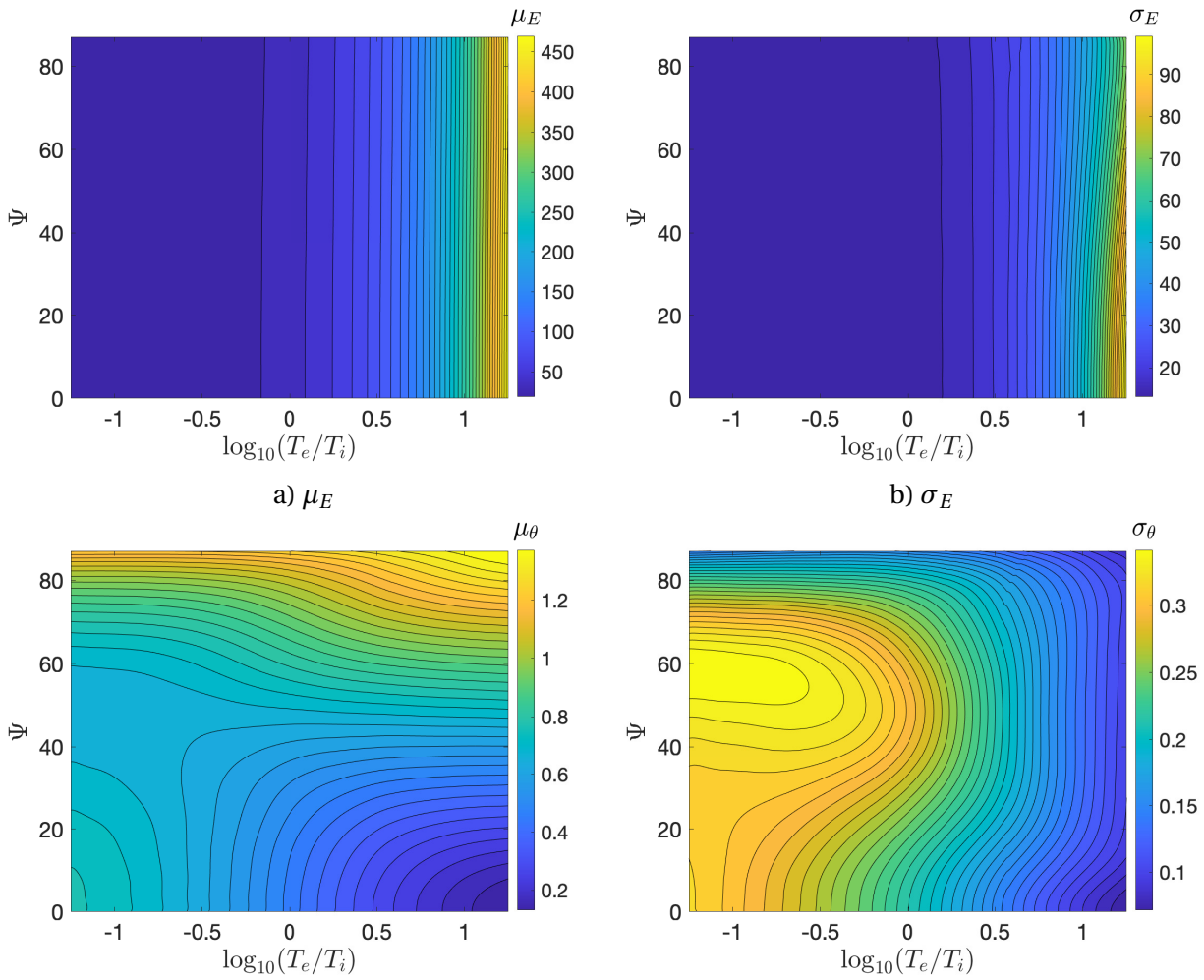
Figure 5.3: First order Sobols' indices for the 4D cases; ranking the effect of electron-to-ion temperature ratios $nicefracT_eT_i$ (a)), magnetic field inclination angles ψ (b)), magnetic field strength B (c)) and plasma density n (d)) on ions' energies and angles moments at **PSI**. Using a number of samples $N = 50,000$.



a) S_1 $\left(\log_{10}(T_e/T_i)\right)$ b) S_2 (ψ)

c) S_3 (B) d) S_4 $\left(\log_{10}(n)\right)$

Figure 5.4: Total order Sobols' indices for the 4D cases; ranking the effect of electron-to-ion temperature ratios T_e/T_i (a)), magnetic field inclination angles ψ (b)), magnetic field strength B (c)), and plasma density n (d)) on ions' energies and angles moments at **PSI**. Using a number of samples $N = 50,000$.



a) μ_E b) σ_E

c) μ_θ d) σ_θ

Figure 5.5: Variation of ions' energies and angles moments at **PSI** with respect to the two-dimensional space of the most impactful physical parameters, namely, the electron-to-ion temperature ratios T_e/T_i and magnetic field inclination angle ψ, within ranges defined in Table 3.2. Values were evaluated using the moments surrogate model in chapter 3 using 250×250 points grid.

Comparing Figure 5.5 b) with **DS** and **MPS** dependencies on the physical parameters shown in Figure 4.16 it can be noted that the spread of energy tends to decrease in the regions where the **MPS** drop is strong. Meaning that, in such regions, ions will keep accelerating in a direction parallel to the magnetic field lines, and the plasma cooling effect is more obvious.

The relation of **IEAD**s angles' moments θ shown in figures 5.5 c) and 5.5 d) with respect to physical parameters T_e/T_i & ψ is more complicated. In general, for the same value of T_e/T_i the mean values of the ions' angle of impact θ increase with ψ. This effect becomes more significant as T_e/T_i increases. As discussed in chapter 2 the initial distribution of the ions in the plasma domain can be assumed to be Maxwellian, with the ions' temperature T_i representing the spread of the ions. Such distribution is then accelerating along the magnetic field lines in the presheath. Magnetic pre-sheath then accelerates ions parallel with the surface normal. Finally, the Debye sheath accelerates ions toward the surface. Comparing Figure 5.5 c) with Figure 4.15 it can be noted that the ions' angle of impact follows the magnetic field inclination in the regions where **DS** disappears (i.e., top row in Figure 5.5 c)). On the other hand, ions' mean angles of impact are close to 45° for $\log_{10}(T_e/T_i) < -0.5$ and $\psi < 70°$, that is the angle of impact for ions with thermal distributions. Finally, the region in the right lower corner (i.e., $\log_{10}(T_e/T_i) > 0$ and $\psi < 60°$) is dominated by the effect of the **DS** potential drop shown in Figure 4.16 a) where ions follow a direction normal to surface normal.

The standard deviation of the **IEAD**s' angles of impact at the **PSI** (also correlated with the ions' temperature) is shown in Figure 5.5 d). Comparing the standard deviations of ions' angle of impact with the regions in the physical input parameter space where the **MPS** or **DS** dominates as shown in Figure 4.16, it can be concluded that the standard deviation of ions' angle of impact tends to be higher in the regions where the potential drop across the **MPS** and the **DS** are comparable. Where, in such regions, the **MPS** will accelerate ions toward magnetic field lines, followed by the **DS** will try to bend ions' flow speed toward the surface normal direction. The combined effect will result in a larger spread of ions' angle of impact, thus, a higher standard deviation. Such an effect cannot be seen in the regions where the **MPS** or **DS** disappears.

5.3.2 Local Sensitivity Analysis (LSA) on WEST Like Conditions

While the Global Sensitivity Analysis (**GSA**) performed in section 5.3.1 ranks the influence of input parameters on the **RoI**s for a wide range of inputs, the Local Sensitivity Analysis (**LSA**) is concerned with the effects of input parameters around a specific point, or a limited set of point in the input parameter space. The main goal of performing **LSA** is to acquire more information about the effect of extended input parameter space on a specific point of interest. Because **LSA** can be computed using very few simulations when compared with **GSA**, the dimensions of input

Table 5.5: Numerical parameters for **WEST OAT** simulations.

parameter	value
Domain Size $[m]$	0.005
Number of Cells	1000
Total Time $[s]$	5×10^{-6}
dt $[s]$	2×10^{-10}
Number of Time Steps	25,000
Number of Particles	2×10^{7}
Particles Per Cell	20,000

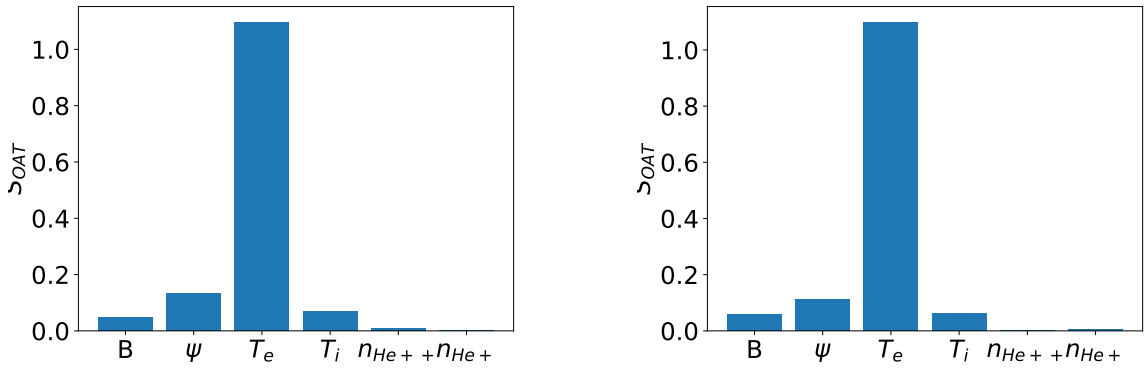
parameters space can be significantly increased. Accordingly, the **LSA** will serve as a guide on the input parameters that should be included in future **GSA** and surrogate model construction.

The problem chosen for the **LSA** is points on the divertor of **WEST** [65]. And the **RoI**s chosen for this analysis are total potential drop across the domain ϕ_w, and Moments of the **IEAD** for both of the species $He+$ and $He++$, including, the mean ions' energies and angles, the standard deviation of ions' energies and angles, as well as the most probable values of the moments. hPIC2 [5] was used to generate the **RoI**s of the targeted points and simulations. The numerical parameters for all points are summarized in Table 5.5. Numerical parameters were chosen to satisfy the following conditions: the domain size is at least 1000 Debye lengths and larger than the Larmor radii of $He+, He++$ with $dx = \lambda_{D_{min}}$, time-step is at max $dt = {}^{\lambda_{D_{min}}}/_{3v_{th_{max}}}$ and the number of time-steps was chosen such that the total time exceeded the maximum ions' transit time of all points.

The **OAT** indices averaged out over the 5 points of interest at the **WEST**'s divertor, showing the influences of the input parameters tabulated in 5.1 on the total potential drop in the domain were calculated and summarized in Figure 5.6. Detailed indices per point are shown in Figure B.1, Appendix B. As one can read from equations 2.21 & 2.13, the floating wall potential is a strong function of the electrons' temperature T_e, with ions temperature T_i and the magnetic field inclination angle ψ having a lower effect. Unlike what Equation 2.22 suggests, the total potential drop can have a higher value compared with the sum of **DS** and **CS**, thus, ψ can have a higher impact on ϕ, especially in the presence of species with a charge higher than 1. On the other hand, the densities of the species and the magnetic field strength B have little to no impact.

The analysis was extended to include the impact of the input parameters given by Table 5.1 on **IEAD** moments. Results of the **SA** indices of the **IEAD** moments are shown in figures 5.7 for the $He+$ species and 5.8 for $He++$. The indices were computed using the central finite difference approach given by Equation 5.20[5]. Detailed indices for each point at the divertor are shown in figures B.4 to B.7 in Appendix B.

[5]Results of the forward difference method given by Equation 5.19 are shown in figures B.2 & B.3 in Appendix B.



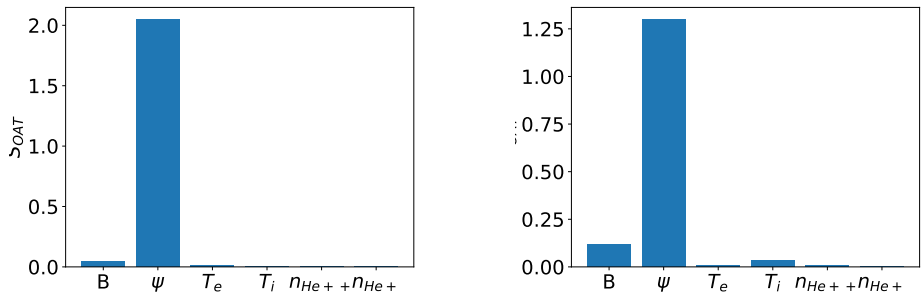
a) Forward Difference

b) Center Difference

Figure 5.6: Average **OAT** indices, ranking the influence of **WEST** plasma parameters near divertor on the *floating wall potential* ϕ_w. Figure 5.6 a) shows indices computed using the forward difference method given by Equation 5.19, and 5.6 b) shows center difference method shown in Equation 5.20. The **OAT** indices were calculated with respect to magnetic field strength B and its corresponding angle ψ, electrons' temperature T_e, ions' temperature T_i, and the densities of $He+$ and $He++$.

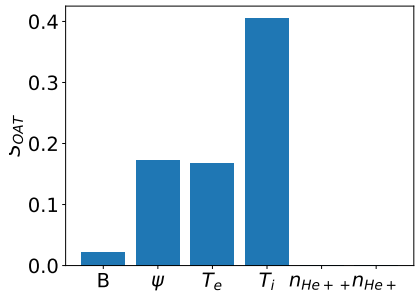
Figure 5.6 shows the **OAT** sensitivity indices of the floating wall potential ϕ_w across the domain with respect to electron temperatures T_e, ion temperature T_i, magnetic field strength B and inclination angle ψ, and the density of the Helium species (n_{He+}, n_{He++}). Sensitivity indices were computed using the central and forward finite difference methods and both are shown in the figure to demonstrate the consistency of the method. Figure 5.6 shows a good agreement with equations 2.13 & 2.21, such that, ϕ_w is a strong function of the electrons' temperatures T_e, followed by the ions temperature T_i and the magnetic field inclination angle ψ. Although Equation 2.22 suggests that the potential drop in the **MPS** ϕ_{MPS} is part ϕ_w, **LSA** suggest that in some cases it does add to the total potential drop (i.e., $\phi_{MPS} + \phi_{DS} > \phi_{w_{theory}}$). The magnetic field strength B was found to have a smaller effect on ϕ_w. The densities of the species were found to have no effect on the floating wall potential ϕ_w.

The **OAT** sensitivity indices for $He+$ and $He++$ **IEAD**s' moments are shown in figures 5.7 and 5.8, respectively. For both species, the magnetic field inclination angle ψ was found to have the most significant effect on the ions' mean angle of impact μ_θ, such observation is consistent with the results of section 5.3.1. However, the standard deviation of ions' angle of impact σ_θ also shows similar dependencies, the reason behind the deviation from what Figure 5.2 b) suggests can be referred to the fact that the input parameters of the **LSA** points fall near the region where the **DS** disappears (i.e., $\psi \sim 80°$, $log_{10}(T_e/T_i) \sim 1$), where ions tend to follow the magnetic field lines as discussed in section 5.3.1. In fact, analysis of the point-wise **OAT** indices shown in figures B.4 and B.5 shows that indices of σ_θ dramatically change with changing the point. Finally,

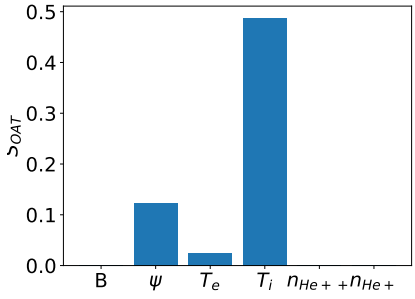


a) $He+$, $\bar{\theta}$

b) $He+$, σ_θ



c) $He+$, $\bar{E}$



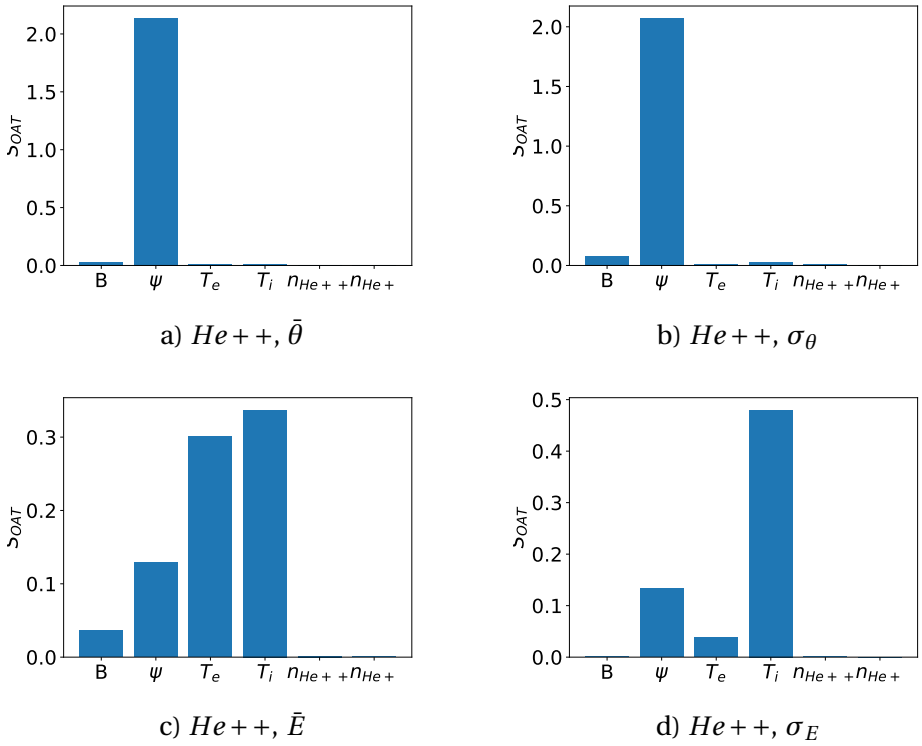
d) $He+$, σ_E

Figure 5.7: Average **OAT** indices, ranking the influence of **WEST** plasma parameters near divertor on $He+$ **IEAD** moments. Indices were calculated using center finite difference method given by Equation 5.20.

electrons' temperature T_e, ions' temperature T_i, and ions' species densities showed little to no effect on μ_θ and σ_θ.

The effect of electrons and ions temperature on the ions' energies mean μ_E and standard deviation σ_E were also analyzed. Figures 5.7 d) and 5.8 d) show strong dependencies on ions' temperatures, which is expected since the initial spread of ions is a direct function of their initial temperature as discussed in chapter 2. While a significant effect of sheath cooling is expected, electron temperatures for data points of interest have relatively small values, thus, potential drops also have small values. On the other hand, μ_E showed strong dependencies of both, electrons and ions temperatures, the latter has two contributions toward μ_E, one resulting from ions' thermal energy and the other due to its effect on the floating wall potential. On the other hand, the electrons' temperature is directly affecting the floating wall potential. This also explains the reason behind the higher impact of T_e on higher Z specie in Figure 5.8 c)[6].

[6]Higher **OAT** indices for ions' temperature should not be interpreted as a higher impact of ions' temperature when compared with electron temperature. The discrepancy is in fact a result of perturbation values choice in Table 5.1.



a) $He++$, $\bar{\theta}$

b) $He++$, σ_θ

c) $He++$, $\bar{E}$

d) $He++$, σ_E

Figure 5.8: Average **OAT** indices, ranking the influence of **WEST** plasma parameters near divertor on $He++$ **IEAD** moments. Indices were calculated using center finite difference method given by Equation 5.20.

5.4 Main Takeaways

The main goals of chapter 5 are summarized as follows. First, to demonstrate the robustness of the surrogate models constructed in chapters 3 and 4. Second, to estimate the effects of electrons-to-ions temperature ratios T_e/T_i and magnetic field inclination angle ψ on plasma potentials as well as the **IEAD** moments. The analysis for the latter was extended to include additional parameters, the magnetic field strength B, and plasma density n. Third, to explore potential additional parameters that can be included in future surrogate models construction and uncertainty quantification for magnetically confined fusion problems. The first and second goals were achieved by performing Global Sensitivity Analysis **GSA** with samples computed using the surrogate models. The third goal was met by performing local sensitivity analysis using actual data measured on a fusion device divertor.

Global Sensitivity Analysis **GSA** based on Sobols' indices method (a variance-based method) using data computed using potentials surrogate models revealed strong dependencies of floating wall potential ϕ_w on electrons-to-ions temperature ratios T_e/T_i with no dependencies on the magnetic field inclination angle ψ. The potential drop across the Debye sheath **DS** and the magnetic pre-sheath **MPS** showed a moderate dependency on T_e/T_i with a lower effect from ψ. However, the second-order indices revealed a significant effect of input parameters interaction, suggesting the existence of regions in the input parameter space where **MPS** is more significant. Such a region was associated with **DS** disappearance.

First and total orders Sobols' indices were computed for **IEAD** moments with data sampled using the **IEAD**'s surrogate model. Revealing that the electron-to-ion temperature ratio and the magnetic field inclination are the two parameters whose variation most affects the **IEAD**. On the other hand, the results suggest that the other parameters, magnetic field strength B and plasma density n have little to no effect on the **IEAD**. In terms of ions' impact angle and kinetic energy moments, it was observed that ions' mean angle of impact is mainly affected by the magnetic field angle ψ flowed by electrons-to-ions temperature ratios T_e/T_i, whereas the mean energy of impact is exclusively a function of electrons-to-ions temperature ratios T_e/T_i. With respect to standard deviation, the energy of impact standard deviation is primarily a function of electrons-to-ions temperature ratios T_e/T_i, whereas the angle standard deviation is affected by electrons-to-ions temperature ratios T_e/T_i followed by the magnetic field inclination ψ. A deeper analysis of the results revealed that the means and standard deviation of ions' impact energy is directly correlated with the floating wall potential values, whereas ions' mean angle of impact is correlated with the ratio of the potential drop across the **MPS** with respect to the floating wall potential ϕ_{MPS}/ϕ_w. Similarly, the standard deviation of ions' angle of impact is correlated with the ratio of the potential drop across the **DS** with respect to the floating wall potential ϕ_{DS}/ϕ_w.

Finally, a Local Sensitivity Analysis **LSA** based on One At a Time **OAT** method was performed using data measured at **WEST** divertor. The main motivations for performing such a **LSA** are the following: Rank the importance of input parameters on **IEAD**s to give guidance on the parameters that can be in any future **UQ** analysis around these points, thus, reducing the computational cost for such analysis. (ii) Give a general insight of the parameter space that should be included in a higher parameter space order **GSA** and surrogate model construction. (iii) Have a better understanding of **IEAD** behavior in a real Tokamak application. While **LSA** results are in general consistent with the one obtained in the **GSA**, the presence of additional input parameters showed that the standard deviations of ions' impact energy are mainly a function of the ions' temperature T_i, while the mean value is a function of electron temperature. Suggesting a need to separate the two temperatures in future analysis. In addition, the existence of multi-species did not show a significant effect from their densities, however, the charge significantly affected their individual **IEAD**s moments.

Chapter 6

Conclusions

6.1 Conclusions

A strategy based on Least Square Approximation **LSA** for constructing a surrogate model for Ion Energy-Angle Distribution **IEAD** produced by hPIC in high-dimensional parameter space was introduced. The **LSA** strategy along with **ANN** strategy were also used to construct a surrogate model for the plasma sheath potentials. Up to four input parameters relevant to magnetic confinement fusion devices were used: the electron-to-ion temperature ratio T_e/T_i, the magnetic field inclination angle with respect to surface normal ψ, magnetic field strength B, and plasma density n. Training and testing data sets were generated using hPIC simulations of Hydrogen plasma. The **LSA** model utilized a sparse grid to efficiently sample the input space, thus reducing the model construction cost. The **IEAD**s were normalized via coordinates transformation to overcome the challenge of variable support of the **IEAD**. This also allowed for using a nonuniform binning of the **IEAD**. Then, a bin-by-bin least square data fitting was used. The **IEAD** surrogate model showed L_2 norm errors within acceptable ranges and is several orders of magnitudes more efficient compared to the PIC simulations. In addition to the **LSA** method, two **ANN** architectures introduced for the excess-zero problem were applied to the plasma potentials problem. While the **ANN** based models resulted in lower errors when compared to the **LSA**, they are more prone to over-fitting, more computationally intensive, and require more supervision since a careful selection of hyperparameters is required. The classifier of the **ANN** model was used to identify the regions where the Debye sheath disappears, such a problem is important for plasma under the influence of a grazing magnetic field, for example, near the divertor of a Tokamak.

A Global Sensitivity Analysis **GSA** based on Sobols' indices method (a variance-based method) was performed on the moments of the **IEAD** and the sheath potentials drop. The main goals of the **GSA** are to demonstrate the robustness of the surrogate models, and to unfold the impact of the input parameters on the plasma sheath structure, and subsequently on the **IEAD**'s moments.

The samples necessary for computing Sobols' indices were drawn from the surrogate models introduced in chapters 3 and 4. The main findings of the **GSA** are the following: Electron-to-ion temperature ratio T_e/T_i dominates the effect on the floating wall potential ϕ_w and ion's mean and standard-deviation of impact energy ($\overline{E}$, σ_E), and has a significant effect on potential drop across the **DS** and **MPS** as well as impact angle ($\overline{\theta}$, σ_θ). A direct correlation was observed between ($\overline{E}$, σ_E) & ϕ_w; $\overline{\theta}$ & ϕ_{MPS}/ϕ_w; and σ_θ & ϕ_{DS}/ϕ_w. Local Sensitivity Analysis **LSA** based on One-At-a-Time **OAT** method was performed using data measured at WEST divertor. This analysis aimed to explore an extended parameter space for future surrogate model constructions. The **LSA**'s results were consistent with the **GSA**'s. However, some results suggested that ion temperature should be treated as a separate input.

6.2 Future Work

The proposed surrogate model approach was computationally efficient in reconstructing **IEAD** in high-dimensional parameter space, especially for applications related to **UQ** and **SA**. Several improvements to the surrogate model can be made and can be classified into two main categories: model's related improvements and method's related improvements. The model's related improvements include: (i) Extend the input parameter space to include more parameters relevant to magnetic confined fusion devices, including: multi-ions species, ions' source shape, etc. (ii) Include more **RoI**s like floating wall potential, ions' moments (e.g., density, flow speed, temperatures, etc), and boundary conditions. (iii) Automate the current analysis to facilitate **IEAD** reconstruction for future purposes. The method's related improvement includes: (i) A cross-validation data set different than the training and testing data set can be used to refine the selection of surrogate model components, such as the polynomial basis functions with their associated approximations, thus, reducing potential over-fitting during the **LSF** procedures. (ii) Explore other surrogate model construction techniques (e.g., machine learning and polynomial chaos expansion **PCE**), which could replace either the bin-by-bin fitting procedures or fitting the whole **IEAD** histogram. (iii) Incorporate a surrogate model for **IEAD** variance to provide a tool to estimate the noise at each point of the parameter space. (iv) Constructed a surrogate model for the processes at **PSI** rather than the complex **IEAD**. The latter requires coupling of hPIC with a plasma material interaction code such as rustBCA. The latter has a drawback of the limited applicability of the surrogate model since the output will also be unique to the wall material.

Due to the fact that artificial neural networks are known to be versatile, the **ANN** methods used in chapter 4 are expected to perform just as well when extending the dimensionality of the parameters space, with the drawback of the increased computational cost of training as more complex neural network architectures will be needed. Input parameters candidates

necessary for fusion plasma modeling include ions' densities of multi-species plasma, magnetic field strength, domain size, collision cross-sections, etc. In addition, since **IEAD**s are usually zero excess histograms, the excess zero **ANN** models are a good candidate for the modeling of Ion Energy-Angle Distributions (**IEAD**) when coupled with a more sophisticated regression algorithm.

While performing **GSA** on the entire input space provides a comprehensive understanding of the model behavior and the contribution of each input variable to the output variability over the full range of input values. It is useful to apply the **GSA** on sub-domains of the input parameters space. This can be justified by noting that the **IEAD** can be a strongly non-linear function with some input parameters, and some regions in the input space are dominated by a unique physical behavior. For example, at grazing magnetic field (i.e., $\psi > 60^{\circ}$) the **DS** can disappear and the **MPS** dominates the potential drop, thus will affect the **IEAD** differently. This phenomenon was observed in Figure 4.15 and was noted earlier by [15]. Another example, at $T_e/T_i >> 1$ an ion acoustic instability can be triggered [67] leading to a significant alternation to the **IEAD**.

References

[1] G. Federici, C. Skinner, J. Brooks, *et al.*, "Plasma-material interactions in current tokamaks and their implications for next step fusion reactors," *Nuclear Fusion*, vol. 41, no. 12, pp. 1967–2137, Dec. 2001. DOI: 10.1088/0029-5515/41/12/218. [Online]. Available: https://doi.org/10.1088/0029-5515/41/12/218.

[2] J. N. Brooks, A. Hassanein, A. Koniges, *et al.*, "Scientific and computational challenges in coupled plasma edge/plasma-material interactions for fusion tokamaks," *Contributions to Plasma Physics*, vol. 54, no. 4-6, pp. 329–340, 2014. DOI: https://doi.org/10.1002/ctpp.201410014. eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/ctpp.201410014. [Online]. Available: https://onlinelibrary.wiley.com/doi/abs/10.1002/ctpp.201410014.

[3] J. T. Drobny and D. Curreli, "Rustbca: A high-performance binary-collision-approximation code for ion-material interactions," *Journal of Open Source Software*, vol. 6, no. 64, p. 3298, 2021.

[4] R. Khaziev and D. Curreli, "Hpic: A scalable electrostatic particle-in-cell for plasma–material interactions," *Computer Physics Communications*, vol. 229, pp. 87–98, 2018, ISSN: 0010-4655. DOI: https://doi.org/10.1016/j.cpc.2018.03.028. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0010465518301012.

[5] L. Meredith, M. Rezazadeh, M. Huq, *et al.*, "Hpic2: A hardware-accelerated, hybrid particle-in-cell code for dynamic plasma-material interactions," *Computer Physics Communications*, vol. 283, p. 108 569, 2023, ISSN: 0010-4655. DOI: https://doi.org/10.1016/j.cpc.2022.108569. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0010465522002880.

[6] L. Tonks and I. Langmuir, "A general theory of the plasma of an arc," *Phys. Rev.*, vol. 34, pp. 876–922, 6 Sep. 1929. DOI: 10.1103/PhysRev.34.876. [Online]. Available: https://link.aps.org/doi/10.1103/PhysRev.34.876.

[7] D. Bohm, "The characteristics of electrical discharges in magnetic fields," *Qualitative Description of the Arc Plasma in a Magnetic Field*, 1949.

[8] F. C. Francis, *Introduction to plasma physics and controlled fusion*, 2016.

[9] M. A. Lieberman and A. J. Lichtenberg, *Principles of plasma discharges and materials processing*. John Wiley & Sons, 2005.

[10] P. C. Stangeby *et al.*, *The plasma boundary of magnetic fusion devices*. Institute of Physics Pub. Philadelphia, Pennsylvania, 2000, vol. 224.

[11] K.-U. Riemann, "The bohm criterion and sheath formation," *Journal of Physics D: Applied Physics*, vol. 24, no. 4, p. 493, 1991.

[12] A. A. Vlasov, "The vibrational properties of an electron gas," *Soviet Physics Uspekhi*, vol. 10, no. 6, p. 721, 1968.

[13] R. Khaziev and D. Curreli, "Ion energy-angle distribution functions at the plasma-material interface in oblique magnetic fields," *Physics of Plasmas*, vol. 22, no. 4, p. 043 503, 2015. DOI: https://doi.org/10.1063/1.4916910.

[14] R. Chodura, "Plasma–wall transition in an oblique magnetic field," *The Physics of Fluids*, vol. 25, no. 9, pp. 1628–1633, 1982. DOI: 10.1063/1.863955. eprint: https://aip.scitation.org/doi/pdf/10.1063/1.863955. [Online]. Available: https://aip.scitation.org/doi/abs/10.1063/1.863955.

[15] P. Stangeby, "The chodura sheath for angles of a few degrees between the magnetic field and the surface of divertor targets and limiters," *Nuclear Fusion*, vol. 52, no. 8, p. 083 012, Jul. 2012. DOI: 10.1088/0029-5515/52/8/083012. [Online]. Available: https://doi.org/10.1088/0029-5515/52/8/083012.

[16] K.-U. Riemann, "Theory of the collisional presheath in an oblique magnetic field," *Physics of plasmas*, vol. 1, no. 3, pp. 552–558, 1994.

[17] E. Ahedo, "Structure of the plasma-wall interaction in an oblique magnetic field," *Physics of Plasmas*, vol. 4, no. 12, pp. 4419–4430, 1997.

[18] D. Moseev and M. Salewski, "Bi-maxwellian, slowing-down, and ring velocity distributions of fast ions in magnetized plasmas," *Physics of Plasmas*, vol. 26, no. 2, p. 020 901, 2019. DOI: 10.1063/1.5085429. eprint: https://doi.org/10.1063/1.5085429. [Online]. Available: https://doi.org/10.1063/1.5085429.

[19] J. P. Boris *et al.*, "Relativistic plasma simulation-optimization of a hybrid code," in *Proc. Fourth Conf. Num. Sim. Plasmas*, 1970, pp. 3–67.

[20] C. K. Birdsall, "Particle-in-cell charged-particle simulations, plus monte carlo collisions with neutral atoms, pic-mcc," *IEEE Transactions on plasma science*, vol. 19, no. 2, pp. 65–85, 1991.

[21] M. K. Stoyanov and C. G. Webster, "A dynamically adaptive sparse grids method for quasi-optimal interpolation of multidimensional functions," *Comput. Math. Appl.*, vol. 71, no. 11, pp. 2449–2465, Jun. 2016, ISSN: 0898-1221. DOI: 10.1016/j.camwa.2015.12.045. [Online]. Available: https://doi.org/10.1016/j.camwa.2015.12.045.

[22] J. Wesson, "Tokamaks," in (International series of monographs on physics), 3rd, International series of monographs on physics. Oxford University Press, 2004, p. 35, ISBN: 0198509227.

[23] Y. YAMAMURA and H. TAWARA, "Energy dependence of ion-induced sputtering yields from monatomic solids at normal incidence," *Atomic Data and Nuclear Data Tables*, vol. 62, no. 2, pp. 149–253, 1996, ISSN: 0092-640X. DOI: https://doi.org/10.1006/adnd.1996.0005. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0092640X96900054.

[24] W. Möller, "The behaviour of hydrogen atoms implanted into metals," *Nuclear Instruments and Methods in Physics Research*, vol. 209-210, pp. 773–790, 1983, ISSN: 0167-5087. DOI: https://doi.org/10.1016/0167-5087(83)90882-7. [Online]. Available: https://www.sciencedirect.com/science/article/pii/0167508783908827.

[25] S. Brezinsek, J. Coenen, T. Schwarz-Selinger, *et al.*, "Plasma–wall interaction studies within the EUROfusion consortium: Progress on plasma-facing components development and qualification," *Nuclear Fusion*, vol. 57, no. 11, p. 116 041, Aug. 2017. DOI: 10.1088/1741-4326/aa796e. [Online]. Available: https://doi.org/10.1088/1741-4326/aa796e.

[26] Y. Yamamura, N. Matsunami, and N. Itoh, "Theoretical studies on an empirical formula for sputtering yield at normal incidence," *Radiation Effects*, vol. 71, no. 1-2, pp. 65–86, 1983. DOI: 10.1080/00337578308218604. [Online]. Available: https://doi.org/10.1080/00337578308218604.

[27] E. Thomas, R. Janev, and J. Smith, "Scaling of particle reflection coefficients," *Nuclear Instruments and Methods in Physics Research Section B: Beam Interactions with Materials and Atoms*, vol. 69, no. 4, pp. 427–436, 1992, ISSN: 0168-583X. DOI: https://doi.org/10.1016/0168-583X(92)95298-6. [Online]. Available: https://www.sciencedirect.com/science/article/pii/0168583X92952986.

[28] S. Koziel and A. Pietrenko-Dabrowska, *Performance-driven surrogate modeling of high-frequency structures*. Springer, 2020.

[29] R. Smith, *Uncertainty Quantification: Theory, Implementation, and Applications* (Computational Science and Engineering). Society for Industrial and Applied Mathematics, 2013, ISBN: 9781611973211. [Online]. Available: https://books.google.com/books?id=4c1GAgAAQBAJ.

[30] D. Gottlieb and S. A. Orszag, *Numerical Analysis of Spectral Methods*. Society for Industrial and Applied Mathematics, 1977. DOI: 10.1137/1.9781611970425. eprint: https://epubs.siam.org/doi/pdf/10.1137/1.9781611970425. [Online]. Available: https://epubs.siam.org/doi/abs/10.1137/1.9781611970425.

[31] M. D. Buhmann, *Radial Basis Functions: Theory and Implementations* (Cambridge Monographs on Applied and Computational Mathematics). Cambridge University Press, 2003. DOI: 10.1017/CBO9780511543241.

[32] B. Fornberg, E. Larsson, and N. Flyer, "Stable computations with gaussian radial basis functions," *SIAM Journal on Scientific Computing*, vol. 33, no. 2, pp. 869–892, 2011. DOI: 10.1137/09076756X. eprint: https://doi.org/10.1137/09076756X. [Online]. Available: https://doi.org/10.1137/09076756X.

[33] C. Rasmussen and C. Williams, *Gaussian Processes for Machine Learning* (Adaptive Computation and Machine Learning series). MIT Press, 2005, ISBN: 9780262182539. [Online]. Available: https://books.google.com/books?id=H3aMEAAAQBAJ.

[34] C. Cortes and V. Vapnik, "Support-vector networks," *Machine learning*, vol. 20, no. 3, pp. 273–297, 1995.

[35] J. A. Suykens and J. Vandewalle, "Least squares support vector machine classifiers," *Neural processing letters*, vol. 9, no. 3, pp. 293–300, 1999.

[36] H. Drucker, C. J. C. Burges, L. Kaufman, A. Smola, and V. Vapnik, "Support vector regression machines," in *Advances in Neural Information Processing Systems*, M. Mozer, M. Jordan, and T. Petsche, Eds., vol. 9, MIT Press, 1996. [Online]. Available: https://proceedings.neurips.cc/paper/1996/file/d38901788c533e8286cb6400b40b386d-Paper.pdf.

[37] O. Kramer, "Scikit-learn," in *Machine learning for evolution strategies*, Springer, 2016, pp. 45–53.

[38] M. Stoyanov, "User Manual: TASMANIAN Sparse Grids v4.0," Oak Ridge National Laboratory, One Bethel Valley Road, Oak Ridge, TN, 37831, Tech. Rep. ORNL/TM-2015/596, 2015. [Online]. Available: https://tasmanian.ornl.gov/documents/UserManual.pdf.

[39] A. Savitzky and M. J. Golay, "Smoothing and differentiation of data by simplified least squares procedures.," *Analytical chemistry*, vol. 36, no. 8, pp. 1627–1639, 1964.

[40] F. Chollet *et al.*, *Keras*, https://keras.io, 2015.

[41] M. M. Lau and K. H. Lim, "Review of adaptive activation function in deep neural network," in *2018 IEEE-EMBS Conference on Biomedical Engineering and Sciences (IECBES)*, IEEE, 2018, pp. 686–690.

[42] P. J. Huber, "Robust Estimation of a Location Parameter," *The Annals of Mathematical Statistics*, vol. 35, no. 1, pp. 73–101, 1964. DOI: 10.1214/aoms/1177703732. [Online]. Available: https://doi.org/10.1214/aoms/1177703732.

[43] D. A. Freund, T. J. Kniesner, and A. T. LoSasso, "Dealing with the common econometric problems of count data with excess zeros, endogenous treatment effects, and attrition bias," *Economics Letters*, vol. 62, no. 1, pp. 7–12, 1999.

[44] W. H. Greene, "Accounting for excess zeros and sample selection in poisson and negative binomial regression models," 1994.

[45] P. A. Lachenbruch, "Analysis of data with excess zeros," *Statistical methods in medical research*, vol. 11, no. 4, pp. 297–302, 2002.

[46] D. Lambert, "Zero-inflated poisson regression, with an application to defects in manufacturing," *Technometrics*, vol. 34, no. 1, pp. 1–14, 1992, ISSN: 00401706. [Online]. Available: http://www.jstor.org/stable/1269547 (visited on 11/15/2022).

[47] J. G. Cragg, "Some statistical models for limited dependent variables with application to the demand for durable goods," *Econometrica*, vol. 39, no. 5, pp. 829–844, 1971, ISSN: 00129682, 14680262. [Online]. Available: http://www.jstor.org/stable/1909582 (visited on 11/15/2022).

[48] vincent d warmerdam, MBrouns, S. Collot, *et al.*, "Koaning/scikit-lego: 0.6.14," Nov. 2022. DOI: 10.5281/zenodo.7273890.

[49] F. Pedregosa, G. Varoquaux, A. Gramfort, *et al.*, "Scikit-learn: Machine learning in Python," *Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011.

[50] C. X. Feng, "A comparison of zero-inflated and hurdle models for modeling zero-inflated count data," *Journal of statistical distributions and applications*, vol. 8, no. 1, pp. 1–19, 2021.

[51] E. Borgonovo and E. Plischke, "Sensitivity analysis: A review of recent advances," *European Journal of Operational Research*, vol. 248, no. 3, pp. 869–887, 2016.

[52] A. Saltelli, M. Ratto, T. Andres, *et al.*, *Global Sensitivity Analysis. The Primer*. John Wiley & Sons, Ltd, 2008. DOI: https://doi.org/10.1002/9780470725184.

[53] F. Pianosi, K. Beven, J. Freer, *et al.*, "Sensitivity analysis of environmental models: A systematic review with practical workflow," *Environmental Modelling & Software*, vol. 79, pp. 214–232, 2016.

[54] M. Frey and A. Adelmann, "Global sensitivity analysis on numerical solver parameters of particle-in-cell models in particle accelerator systems," *Computer Physics Communications*, vol. 258, p. 107 577, 2021, ISSN: 0010-4655. DOI: https://doi.org/10.1016/j.cpc.2020.107577. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0010465520302770.

[55] A. Vaidheeswaran, A. Gel, M. A. Clarke, and W. Rogers, "Sensitivity analysis of particle-in-cell modeling parameters in settling bed, bubbling fluidized bed and circulating fluidized bed," National Energy Technology Laboratory (NETL), Pittsburgh, PA, Morgantown, WV ..., Tech. Rep., 2021.

[56] J. Carlsson, A. Khrabrov, I. Kaganovich, T. Sommerer, and D. Keating, "Validation and benchmarking of two particle-in-cell codes for a glow discharge," *Plasma Sources Science and Technology*, vol. 26, no. 1, p. 014 003, Nov. 2016. DOI: 10.1088/0963-0252/26/1/014003. [Online]. Available: https://doi.org/10.1088/0963-0252/26/1/014003.

[57] D. Sydorenko, "Particle-in-cell simulations of electron dynamics in low pressure discharges with magnetic fields," Ph.D. dissertation, 2006.

[58] R. Clark and T. Hughes, "Lsp user's manual and reference," in *Mission Research Corporation*, 2005.

[59] K. L. Cartwright and G. A. Radtke, "Numerical uncertainty estimation for stochastic particle-in-cell simulations applied to verification and validation.," Sandia National Lab.(SNL-NM), Albuquerque, NM (United States), Tech. Rep., 2015.

[60] G. A. Radtke, K. L. Cartwright, and L. C. Musson, "Stochastic richardson extrapolation based numerical error estimation for kinetic plasma simulations," Sandia National Lab.(SNL-NM), Albuquerque, NM (United States); Sandia ..., Tech. Rep., 2015.

[61] S. W. Chung, S. D. Bond, E. C. Cyr, and J. B. Freund, "Regular sensitivity computation avoiding chaotic effects in particle-in-cell plasma methods," *Journal of Computational Physics*, vol. 400, p. 108 969, 2020, ISSN: 0021-9991. DOI: https://doi.org/10.1016/j.jcp.2019.108969. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0021999119306746.

[62] S. Joe and F. Y. Kuo, "Remark on algorithm 659: Implementing sobol's quasirandom sequence generator," vol. 29, no. 1, pp. 49–57, Mar. 2003, ISSN: 0098-3500. DOI: 10.1145/641876.641879.

[63] T. Iwanaga, W. Usher, and J. Herman, "Toward SALib 2.0: Advancing the accessibility and interpretability of global sensitivity analyses," *Socio-Environmental Systems Modelling*, vol. 4, p. 18 155, May 2022. DOI: 10.18174/sesmo.18155. [Online]. Available: https://sesmo.org/article/view/18155.

[64] J. Herman and W. Usher, "SALib: An open-source python library for sensitivity analysis," *The Journal of Open Source Software*, vol. 2, no. 9, Jan. 2017. DOI: 10.21105/joss.00097. [Online]. Available: https://doi.org/10.21105/joss.00097.

[65] P. Moreau, A. Le-Luyer, P. Spuig, *et al.*, "The new magnetic diagnostics in the west tokamak," *Review of Scientific Instruments*, vol. 89, no. 10, 10J109, 2018.

[66] M. I. Radaideh and M. I. Radaideh, "Application of stochastic and deterministic techniques for uncertainty quantification and sensitivity analysis of energy systems," *arXiv preprint arXiv:1901.05566*, 2019.

[67] S. D. Baalrud, "Influence of ion streaming instabilities on transport near plasma boundaries," *Plasma Sources Science and Technology*, vol. 25, no. 2, p. 025 008, Feb. 2016. DOI: 10.1088/0963-0252/25/2/025008. [Online]. Available: https://dx.doi.org/10.1088/0963-0252/25/2/025008.

[68] P. Seleson, M. Mustafa, D. Curreli, C. D. Hauck, M. Stoyanov, and D. E. Bernholdt, "Data-driven surrogate modeling of hpic ion energy-angle distributions for high-dimensional sensitivity analysis of plasma parameters' uncertainty," *Computer Physics Communications*, vol. 279, p. 108 436, 2022, ISSN: 0010-4655. DOI: https://doi.org/10.1016/j.cpc.2022.108436. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0010465522001552.

[69] European Organization For Nuclear Research and OpenAIRE, *Zenodo*, en, 2013. DOI: 10.25495/7GXK-RD71. [Online]. Available: https://www.zenodo.org/.

Appendix A

IEAD's Surrogate Model Misc.

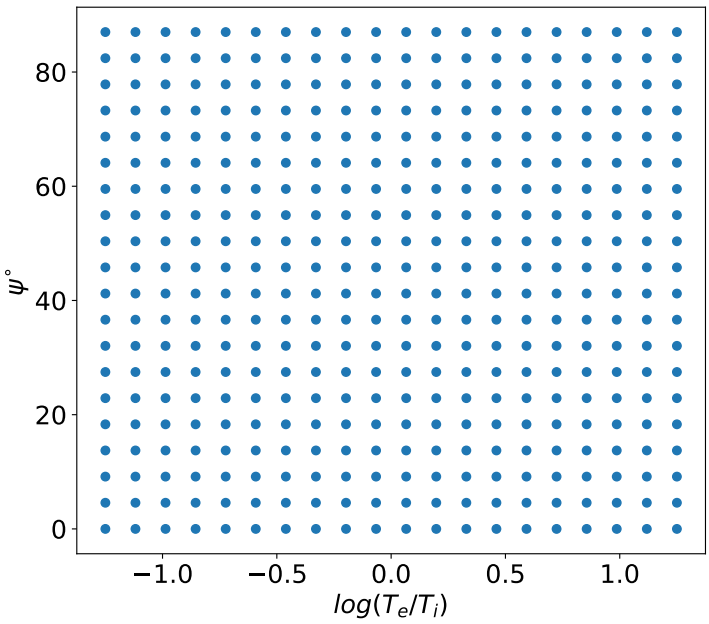


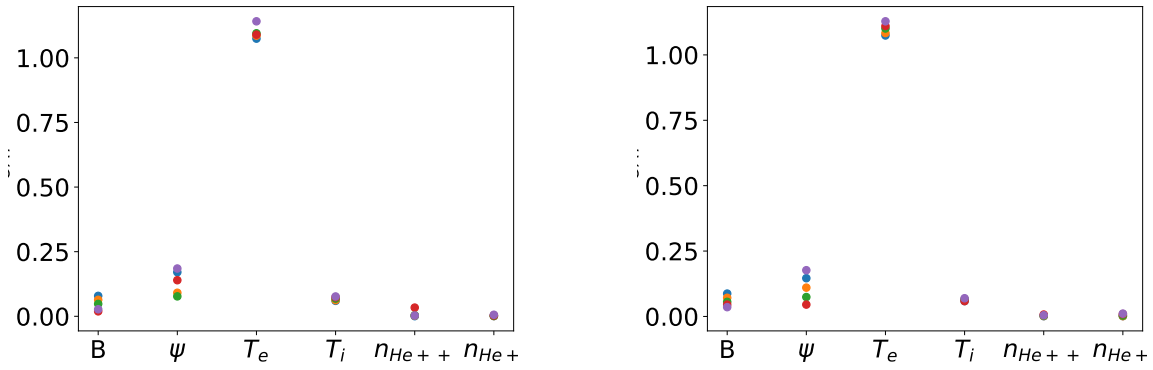
Figure A.1: 20 × 20 testing grid for the 2D **IEAD** surrogate model.

Appendix B

Additional Details for WEST Local Sensitivity Analysis.

Table B.1: WEST points unperturbed input parameters.

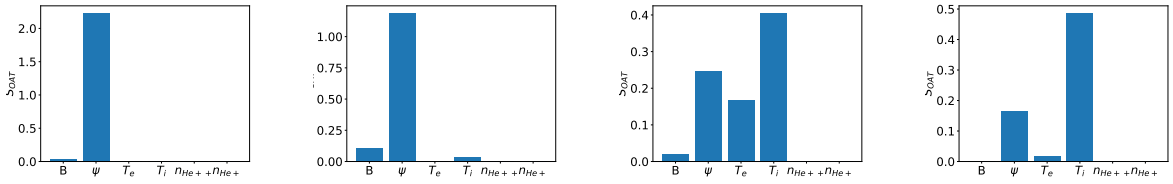
Point	$B\,[T]$	$\psi\,[deg]$	$T_e\,[eV]$	$T_i\,[eV]$	$n_{He++}\,[m^{-3}]$	$n_{He+}\,[m^{-3}]$
1	3.5931	80.9050	16.2170	68.7540	1.23E+18	1.30E+17
2	3.5929	80.9990	7.6975	40.8300	5.86E+17	1.30E+17
3	3.5928	81.0860	3.1478	21.8470	2.40E+17	1.30E+17
4	3.5926	81.1960	0.9555	9.4902	7.28E+16	1.30E+17
5	3.5925	81.2620	0.2900	4.1225	2.21E+16	1.30E+17



a) Forward Difference

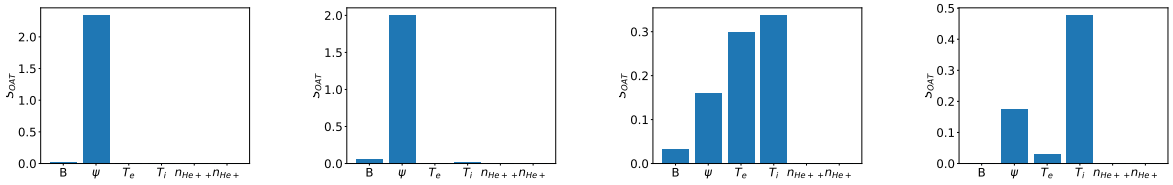
b) Center Difference

Figure B.1: Detailed **OAT** indices, ranking the influence of **WEST** plasma parameters near divertor on the *total potential drop* ϕ, showing indices for each point. Figure 5.6 a) shows indices computed using the forward difference method given by equation 5.19, and 5.6 b) shows center difference method shown in equation 5.20.



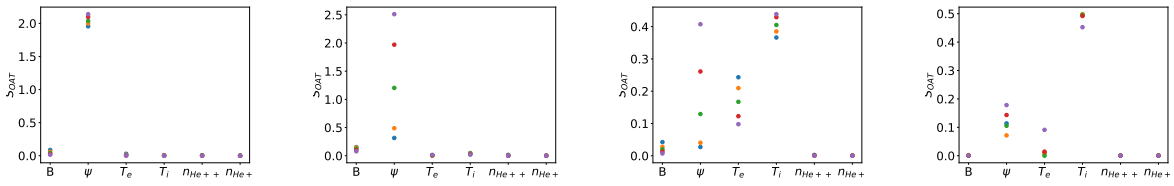
a) $He+$, $\bar{\theta}$ b) $He+$, σ_θ c) $He+$, $\bar{E}$ d) $He+$, σ_E

Figure B.2: Average **OAT** indices, ranking the influence of **WEST** plasma parameters near divertor on $He+$ **IEAD** moments. Indices were calculated using forward finite difference method given by equation 5.19.



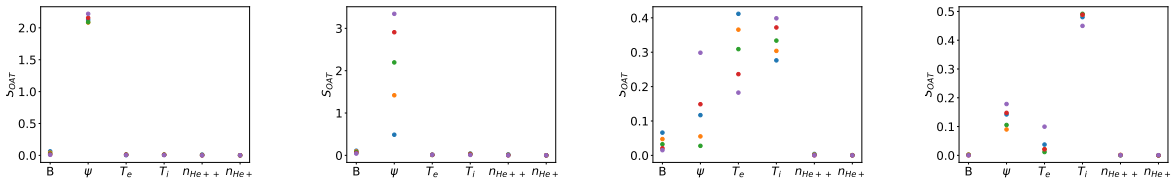
a) $He++$, $\bar{\theta}$ b) $He++$, σ_θ c) $He++$, $\bar{E}$ d) $He++$, σ_E

Figure B.3: Average **OAT** indices, ranking the influence of **WEST** plasma parameters near divertor on $He++$ **IEAD** moments. Indices were calculated using forward finite difference method given by equation 5.19.



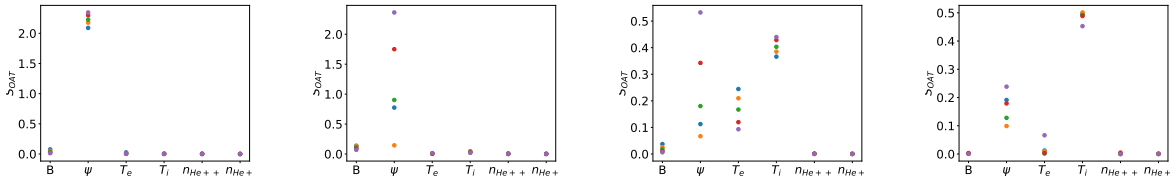
a) $He+$, $\bar{\theta}$ b) $He+$, σ_θ c) $He+$, $\bar{E}$ d) $He+$, σ_E

Figure B.4: Detailed **OAT** indices, ranking the influence of **WEST** plasma parameters near divertor on $He+$ **IEAD** moments. showing indices values for each point. Indices were calculated using center finite difference method given by equation 5.20.



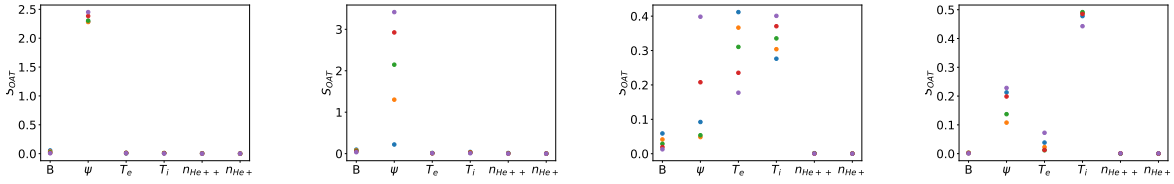
a) $He++$, $\bar{\theta}$ b) $He++$, σ_θ c) $He++$, $\bar{E}$ d) $He++$, σ_E

Figure B.5: Detailed **OAT** indices, ranking the influence of **WEST** plasma parameters near divertor on $He++$ **IEAD** moments. showing indices values for each point. Indices were calculated using center finite difference method given by equation 5.20.



a) $He+$, $\bar{\theta}$ b) $He+$, σ_θ c) $He+$, $\bar{E}$ d) $He+$, σ_E

Figure B.6: Detailed **OAT** indices, ranking the influence of **WEST** plasma parameters near divertor on $He+$ **IEAD** moments. showing indices values for each point. Indices were calculated using forward finite difference method given by equation 5.19.



a) $He++$, $\bar{\theta}$ b) $He++$, σ_θ c) $He++$, $\bar{E}$ d) $He++$, σ_E

Figure B.7: Detailed **OAT** indices, ranking the influence of **WEST** plasma parameters near divertor on $He++$ **IEAD** moments. showing indices values for each point. Indices were calculated using forward finite difference method given by equation 5.19.

Appendix C

Reproducibility

Ensuring that research findings can be reproduced is a crucial aspect of scientific inquiry. The ability to replicate the results of a study not only verifies its validity, but also promotes transparency and facilitates further research. In this section, we detail the steps taken to ensure the reproducibility of our study. We provide a detailed description of the data used and the codes developed to analyze the data, and we make these available for others to access. By doing so, we hope to promote transparency and facilitate future research in this area.

To facilitate reproducibility, we have made the data and codes used in this study openly available. The codes used in the analysis have been uploaded to a publicly accessible repository, along with a detailed README file that explains how to run the codes and reproduce the results. We believe that making the data and codes available to others is critical for advancing scientific research and promoting transparency. By doing so, we hope to encourage other researchers to build upon our work and to replicate our findings.

The following list shows the databases and repositories available for data reproducibility:

- hPIC/hPIC2 were used for training and testing data generation. The code is available on the public repository: https://github.com/lcpp-org/hpic2.git
- Scripts for generating hPIC/hPIC2 input files for sparse grid are available on the public repository: https://github.com/mmustafa5/hpic_hpic2_inputs_gen.git
- Training and testing data necessary for **IEAD** surrogate model construction can be freely accessed online through Constellation, DOI: https://doi.org/10.13139/ORNLNCCS/1846780 [68]
- MATLAB scripts for constructing the **IEAD** surrogate model and **GSA** on **IEAD**s' moments are available on the public repository: https://github.com/pabloseleson/hPIC-surrogate [68]

- Python script for emulating **IEAD** using trained model is available on the public repository: https://github.com/lcpp-org/machine-learning-iead.git
- Data required for the potential surrogate model construction and the **LSA** can be freely accessed online through Zenodo [69], DOI: 10.5281/zenodo.7850117
- Scripts for potentials surrogate model construction, **LSA**, and the M.S. thesis script are available on the repository: https://github.com/mmustafa5/MS_Thesis.git
- Other libraries required for the reproducibility:
  - TASMANIAN [38]: https://github.com/ORNL/TASMANIAN.git
  - SALib [63]: https://github.com/SALib/SALib.git
  - Keras [40]: https://github.com/keras-team/keras.git
  - scikit-learn [49]: https://github.com/scikit-learn/scikit-learn.git
  - scikit-lego [48]: https://github.com/koaning/scikit-lego.git