

UNIVERSITY OF BIRMINGHAM



Department of Metallurgy & Materials

Irradiation Damage Simulations of Platinum Group Metal Modified Austenitic Stainless Steels for Reactor Core Components

by Ben Palmer

A thesis submitted to the University of Birmingham for the degree of Doctor of Philosophy

Supervised by Dr Brian J Connolly and Dr Mark S D Read

**Irradiation Damage Simulations of Platinum Group Metal Modified Austenitic Stainless Steels
for Reactor Core Components**

by

Ben Palmer

Abstract

This work is split into two investigations. The first focuses on deriving the interatomic potentials required to model irradiated austenitic stainless steels doped with small amounts of Palladium or Ruthenium. The second centers on a Fortran program developed to predict the radioactivity of a sample irradiated by a proton beam.

Austenitic stainless steels are an important material, and it is used to construct core components within nuclear reactors. Sensitisation of these steels, by processes such as welding, depletes Chromium from grain boundaries removing the corrosion resistant Cr_2O_3 passive layer. Previous (experimental) work has shown that these steels, when doped with palladium or ruthenium, retain corrosion resistance at the grain boundary. This work investigates whether or not these PGMs deplete or saturate at the grain boundary while under irradiation using Molecular Dynamics (MD) simulations. Interatomic potentials have been derived for Fe-Pd and Fe-Ru as they were not available, and these are essential for the MD simulations of the irradiated grain boundaries. The effect of irradiation on Fe-Ru and Fe-Pd showed depletion/saturation/no effect.

In addition to simulations, it would also be desireable to investigate whether PGMs are depleted or saturated at the grain boundary by emulating neutron damage with a proton beam. The irradiated targets are expected to be radioactive due to the transmutation of target nucleons. A Fortran program, Activity, was written and developed to calculate how radioactive a target sample would be after proton beam irradiation, and from this it can be determined when it would be safe to handle.

Contents

1	Introduction	1
1.1	Motivation	1
1.1.1	Part 1: Activity Computer Program	2
1.1.2	Part 2: Palladium-Iron Potential	2
1.2	Nuclear Power in the UK	2
1.2.1	Nuclear Fuel	2
1.2.2	Generation I Reactors	4
1.2.3	Generation II Reactors	4
1.2.4	Generation III Reactors	4
1.3	An Approaching Energy Gap for the UK	5
1.3.1	Why choose Nuclear Power?	5
1.3.2	Proposed Generation III+ Nuclear Power Plants	6
1.3.3	Generation IV	7
1.3.4	Generation IV Proposed Designs	8
1.4	Damage to Nuclear Core Components by Neutrons	12
1.5	Radiation Damage: Replacing Neutrons with Protons	13
1.6	Replacing Experiment with Simulations	14
1.6.1	Molecular Dynamics	14
1.6.2	Density Functional Theory	16
1.6.3	Interatomic Potentials for Molecular Dynamics	16
2	Background: Ionizing Radiation	17
2.1	Radiation Types Relevant to This Work	17
2.1.1	Introduction	17
2.1.2	Protons and Ions	17
2.1.3	Neutrons	17
2.1.4	High Energy Photons	18
2.2	Effects of Radiation on Organics	20
2.2.1	Radioactivity, Absorbed Dose, Dose Equivalent and Exposure	20

2.2.2	Effect of Radiation on Organics	22
2.2.3	Damage Types	23
2.2.4	Collision Event	23
2.3	Damage Cascades in Metals	24
3	Background: Austenitic Steels and Nuclear Power	26
3.1	Stainless Steel	26
3.1.1	Introduction	26
3.1.2	Grades of Stainless Steel	26
3.2	Austenitic Steels in Nuclear Reactors	29
3.2.1	The Use of Austenitic Steels in Reactors	29
3.2.2	Atom Damage Cascades	29
3.2.3	Damage Rates	29
3.2.4	Swelling	31
3.2.5	Radiation Induced Segregation	31
3.2.6	Irradiation Hardening	31
3.3	Corrosion Resistance of Austenitic Stainless Steels	31
3.3.1	Passive Film Protection of Chromium	31
3.3.2	Sensitization and Passive Film Removal	32
3.3.3	Addition of Molybdenum	33
3.4	Inter Granular Stress Corrosion Cracking	33
3.4.1	IGSCC in Light Water Reactors	35
3.4.2	IGSCC and Advanced Gas-cooled Reactors	35
3.5	Austenitic Stainless Steels in Gen III+ and Gen IV Reactors	36
4	Background: Proton Activation and Radioactive Decay	37
4.1	Ion Sources	37
4.1.1	Linac	37
4.1.2	Cyclotron	37
4.1.3	Synchrotron	38
4.2	Neutron Sources	38
4.2.1	High-Flux Neutron Reactors	38
4.2.2	Spallation	40
4.3	Source Review	42
4.4	Ion Irradiation to Investigate Neutron Damage	42
4.4.1	Ion Irradiation at the University of Birmingham	43
4.4.2	Transmutation of Nuclei by Neutrons and Protons	43

4.4.3	Nuclear Reaction Cross Sections	44
4.4.4	Radioactive Decay	45
4.4.5	Saturation Activity	46
4.4.6	Decay Constants: Joint Evaluated Fission and Fusion File (JEFF) 3.3 Data File	46
4.4.7	Bateman Equation for Radioactive Decay	46
4.4.8	Cross Section Dependence on Energy and Target Thickness	47
4.5	Simulating Ion Irradiation with SRIM	47
4.5.1	What SRIM Does	47
4.5.2	What SRIM Does Not Do	47
5	Background: Interatomic Potential Fitting	49
5.1	Experiment, Modelling and Theory	49
5.1.1	Introduction	49
5.1.2	Simulating Materials on a Variety of Scales in Time and Space	50
5.2	Properties of Crystals	50
5.2.1	Introduction	50
5.2.2	Bulk Modulus	51
5.2.3	Equation of State	51
5.2.4	Voigt Notation	52
5.2.5	Elastic Constants	53
5.2.6	Calculating Elastic Constants for a Cubic Crystal	54
5.2.7	Calculating Elastic Constants for Orthorhombic Crystal	55
5.2.8	Stability Conditions	57
5.2.9	Computing Values from Elastic Constants	58
5.2.10	Correlation of Melting Temperature in Metals with Elastic Constants	58
5.2.11	Poisson Ratio	59
5.3	Interatomic Potentials	60
5.3.1	Introduction	60
5.3.2	Pair Potentials	60
5.3.3	Many Body Potentials	62
5.3.4	Two Band Embedded Atom Method	64
5.3.5	Many Bands for the Embedded Atom Method	66
5.3.6	Function Types used by EAM and Two-Band EAM	67
5.3.7	Modified Embedded Atom Method	69
5.3.8	Calculating Energy, Force and Stress	69
5.3.9	Choice of Functions and Functionals	74

5.3.10	Sacrificing Physical Elegance	75
5.4	Force Matching	75
5.5	Magnetism	76
5.5.1	Brief History of Magnetism	76
5.5.2	Magnetism: Moving Charges and Spin	76
5.5.3	Ferromagnetism and Antiferromagnetism	77
5.6	Density Functional Theory	77
5.6.1	Brief Overview of DFT	78
5.6.2	Time Independent Schrdinger Equation	78
5.6.3	Many Body TISE	79
5.6.4	Born-Oppenheimer	80
5.6.5	Crystals, Reciprocal Space and Bloch Theorem	81
5.6.6	Hohenberg-Kohn Theorem	84
5.6.7	Kohn-Sham Equations	85
5.6.8	Self-Consistent Solution	86
5.6.9	Exchange-Correlation Energy	87
5.6.10	Pseudopotentials	87
5.6.11	Brillouin Zone, Smearing and K-Points	91
5.6.12	Smearing	91
5.7	DFT: Magnetism	91
5.7.1	Collinear	91
5.7.2	Non-Collinear	91
5.8	Collisions and Interatomic Potentials	92
5.9	Classical Molecular Dynamics	92
5.9.1	Introduction	92
5.9.2	Neighbour List	92
5.9.3	Verlet Timestep	92
5.9.4	Molecular Dynamics Codes	93
5.9.5	Selecting a Time Scale	93
5.10	Optimization	93
5.10.1	Introduction	93
5.10.2	Continuous and Discrete Optimization	94
5.10.3	Constrained and Unconstrained Optimization	94
5.10.4	Global and Local Optimization	95
5.10.5	Global Optimization Algorithms	95
5.10.6	Local Optimization Algorithms	98

5.10.7	Gradient Descent	98
5.10.8	Newton-Gauss	99
5.10.9	Levenberg Marquardt	99
6	Methodology: Proton Activation and Radioactive Decay	100
6.1	Introduction	100
6.2	Activation by Ion Irradiation	100
6.2.1	Laplace Transform	100
6.2.2	Constructing the Differential Equations	101
6.2.3	Numerical Inversion of the Laplace Transform	102
6.2.4	Analytic Solution by Partial Fraction Expansion	102
6.2.5	Preference: Analytic over Numeric	105
6.3	Computational Methods	105
6.4	Ion Trajectory Data	106
6.5	Activation by Neutron Irradiation	107
6.5.1	Introduction	107
6.6	Cyclotron Irradiated Iron	107
6.6.1	Cyclotron Beam Line	107
6.6.2	Measurement of Sample Activity	108
6.6.3	Prediction of Activity	108
7	Methodology: Interatomic Potential Fitting	110
7.1	Interatomic Potential Fitting	110
7.2	Experimental & DFT Bulk Property Data	111
7.3	DFT Calculations	111
7.3.1	Quantum Espresso	111
7.3.2	Pseudopotential Selection	111
7.3.3	Smearing Type	112
7.4	Parameter Convergence	113
7.4.1	Ecutwfc and Ecutoff	113
7.4.2	K-Points and Smearing Value	114
7.4.3	Automating Convergence	114
7.5	Bulk Property Calculations	114
7.5.1	QEEOS Python Program	114
7.5.2	DFT Calculations: Austenitic Iron Properties	114
7.6	Reference Database	115
7.6.1	QEFORFIT Python Program	115

7.6.2	DFT Calculations: Iron and Palladium	115
7.6.3	Atomic Configurations for DFT Calculations	115
7.7	DFT Errors and Convergence Failures	115
7.7.1	Adjustment of DFT Parameters	115
7.7.2	A Lack of Memory	115
7.7.3	Not Straying Too Far From Reality	115
7.8	EAMPA: Potential Analysis and Fitting Code	116
7.9	Iron-Palladium Potentials	116
7.10	DL_POLY Contribution	116
7.10.1	Modifying DL_POLY: 2BEAM	116
7.10.2	Mailshot Extract	116
8	Results: Activity Code	118
8.1	Extended Bateman Equations	118
8.2	Activity V1 Computer Package	118
8.3	Activity V2 Computer Package	119
8.4	Neutron Activation	119
8.5	Proton Irradiated Iron	119
8.5.1	Cobalt-55	119
8.6	Cyclotron Beam Line - Proton Irradiated Iron	120
8.7	SRIM Proton Irradiated Iron	121
8.8	Estimate of Activity Using TENDL Data	122
8.9	Activity V1 - Simulated Proton Irradiated Iron	123
8.10	Activity V2 - Simulated Proton Irradiated Iron	124
8.11	Comparing Co-55 Peaks	126
8.12	Predicted Activity for 100DPA Irradiated Iron	126
9	Results: DFT Reference Database	128
9.1	Introduction	128
9.2	QECONVERGE Python Code	128
9.2.1	Purpose of Code	128
9.2.2	Source Code and Instructions	129
9.3	Convergence of Key DFT Settings and Testing	129
9.3.1	Pseudo-potential	129
9.3.2	Collinear Spin Calculations	130
9.3.3	Convergence Results	131
9.4	Preliminary Calculations	132

9.4.1	Relaxed Crystal Calculations	132
9.4.2	Iron-Palladium Atom Positions	133
9.5	QEEOS Python Code	134
9.5.1	Purpose of Code	134
9.5.2	Source Code and Instructions	134
9.6	Calculated Elastic Properties	134
9.6.1	FCC Aluminium	134
9.6.2	BCC Iron	136
9.6.3	FCC Palladium	138
9.6.4	FCC Iron	139
9.7	QEFORFIT Python Code	141
9.7.1	Purpose of Code	141
9.7.2	Source Code and Instructions	141
9.7.3	DFT Configuration Database	141
10	Results: Interatomic Potential Fitting	142
10.1	EAMPA Python Code	142
10.1.1	Purpose of Code	142
10.1.2	Source Code and Instructions	142
10.1.3	Modular Design	142
10.1.4	Energy, Force and Stress Calculations	143
10.1.5	Bulk Property Calculations	143
10.1.6	RSS - How Well The Potential Fits Experimental & DFT Data	143
10.1.7	Analytic Potential Fitting	143
10.1.8	Knot to Knot Splines	145
10.1.9	Minimising Runtime	152
10.1.10	Interpolation for Non Analytic Potentials	152
10.2	Iron-Palladium Potentials: Data and Initial Potential Parameters	154
10.2.1	Bulk Properties for Fitting	154
10.2.2	Reference Database	155
10.2.3	Potential Functions & Functionals	155
10.3	Iron-Palladium Potentials: First Potential	159
10.4	Iron-Palladium Potentials: Second Potential	161

11 Conclusions	165
11.1 Inter Granular Stress Corrosion Cracking	165
11.2 Activity Code	166
11.3 Modified Equation and Activity Code	166
11.4 Interatomic Potential: Iron-Palladium	167
11.5 Collinear Spin DFT	167
11.6 Bulk Properties of Orthorhombic FCC Iron	167
11.7 Palladium-Iron Configurations as Fitting Data	168
11.8 Iron-Palladium Potential	168
11.9 Two-Band EAM Contribution to DL-Poly	168
12 Future Work	169
12.1 Activity Code	169
12.1.1 Experimental Activity Readings for Ion Irradiated Targets	169
12.1.2 Code Improvements	169
12.2 Potential Fitting and MD Simulations	169
12.3 Improved Potential: Taking Into Account Non-Isotropic Structure of "FCC" Iron	169
12.4 Improved Fitting: Surface Energy, Cohesive Energy and Defects	169
13 Acknowledgements	174
Appendices	175
A Crystal Structures	176
A.1 Common Cubic Structures	176
A.1.1 Simple Cubic	176
A.1.2 Body Centered Cubic	177
A.1.3 Face Centered Cubic	177
A.1.4 Zincblende	178
A.2 FCC Defects	179
A.2.1 Dumbbell <100>	179
A.2.2 Dumbbell <110>	179
A.2.3 Dumbbell <111>	179
A.2.4 Crowdion	180
A.2.5 Octahedral Interstitial	180
A.2.6 Tetrahedral Interstitial	180
B Elastic Constants	181

C Activity V1	183
C.1 Sample Input File	183
C.2 Output Plots	184
C.3 Paper published in Computer Physics Communications	185
C.4 Accompanying Manual	193
D Activity V2	228
D.1 Sample Input File	228
D.2 Sample Output	228
D.3 Output Plots	237
D.4 An Example to Show Saturation In Beam	241
D.5 Accompanying Manual	242
E Activity V1 Full Results for Iron	268
F Activity V2 Full Results for Iron	280
G Neutron Activity Code	286
G.1 Full Source Code	286
G.2 Implementation of the Activity Equations in Python	286
H Iron Activity	289
H.1 Activity vs Beam Energy	289
H.1.1 5MeV Proton Beam - 60 microamp 100DPA	289
H.1.2 10MeV Proton Beam - 60 microamp 100DPA	292
H.1.3 15MeV Proton Beam - 60 microamp 100DPA	293
H.1.4 20MeV Proton Beam - 60 microamp 100DPA	296
H.1.5 25MeV Proton Beam - 60 microamp 100DPA	297
H.1.6 Results Summary	300
I DFT Convergence Plots	301
J DFT Calculated Properties	308
J.1 QE EoS	308
J.2 FCC Aluminium	309
J.2.1 DFT Settings for PWscf	309
J.2.2 Equation of State and Elastic Constants	309
J.2.3 Aluminium FCC Results File	310
J.3 BCC Iron [No Magnetism]	312

J.3.1	Major DFT Settings	312
J.3.2	Equation of State and Elastic Constants	312
J.3.3	Iron BCC Results File	313
J.4	BCC Iron [Ferromagnetic]	316
J.4.1	Major DFT Settings	316
J.4.2	Equation of State and Elastic Constants	316
J.4.3	Iron Ferromagnetic BCC Results File	317
J.5	FCC Iron [Antiferromagnetic]	320
J.5.1	Major DFT Settings	320
J.5.2	Equation of State and Elastic Constants	320
J.5.3	Iron FCC Results File	321
J.6	FCC Palladium	324
J.6.1	Major DFT Settings	324
J.6.2	Equation of State and Elastic Constants	324
J.6.3	Palladium FCC Results File	325

K Potential Functions 328

K.1	Introduction	328
K.2	Functions	328
K.2.1	Ackland-Mendelev Pair	328
K.2.2	Buckingham	329
K.2.3	Cubic Knot Spline	330
K.2.4	Cubic Knot Spline Fixed End	330
K.2.5	Cubic Knot Spline Fixed End Pair	331
K.2.6	Cubic Splines	332
K.2.7	Cubic Spline Plus ZBL	333
K.2.8	Embedding A - Finnis Sinclair	334
K.2.9	Embedding B - Mendelev	335
K.2.10	Embedding C	336
K.2.11	Embedding D - Ackland-Mendelev	336
K.2.12	Embedding E	337
K.2.13	Embedding F	337
K.2.14	Embedding G	338
K.2.15	Lennard-Jones	339
K.2.16	Morse	339
K.2.17	Slater 4S (Squared)	340
K.2.18	ZBL	341
K.2.19	Zero	341

L FePd Fit: Version 1	343
L.1 Potential Files	343
L.2 Potential Plots	345
L.3 Equation of State and Elastic Constant (Distortion) Plots	348
L.4 Cohesive Energy Plots	348
L.5 Surface Energy Plots	349
L.6 Fe-Pd V1 Potential Properties	349
M FePd Fit: Version 2	351
M.1 Potential Files	351
M.2 Potential Plots	353
M.3 Equation of State and Elastic Constant (Distortion) Plots	353
M.4 Cohesive Energy Plots	354
M.5 Surface Energy Plots	354
M.6 Fe-Pd V2 Potential Properties	354

List of Figures

1.1 graph:U-235 and U-238 Fission Cross Sections	3
1.2 Remaining reactor locations in the UK	4
1.3 Electricity in Millions of Tonnes of Oil Equivalent	5
1.4 Inelastic scattering - scattering angle vs energy loss for a range of target atoms	9
1.5 Typical Reactor Flux - Thermal and Fast Spectra	9
1.6 Energy demand to produce hydrogen by electrolysis as temperature changes	10
1.7 Phases of Water: Pressure vs Temperature	11
1.8 Neutron Energy Spectrum from Fission	12
1.9 5KeV Iron PKA Damage Cascade in Iron	15
1.10 10KeV Iron PKA Damage Cascade in Iron	15
1.11 20KeV Iron PKA Damage Cascade in Iron	15
1.12 50KeV Iron PKA Damage Cascade in Iron	15
2.1 Mn 56 Activity	21
2.2 Steel Irradiation Gamma Dose $2.5e7$ Neutrons per cmsquared per second	21
2.3 High Flux Isotope Reactor Neutron Flux	22
2.4 Steel Irradiation Gamma Dose $1.0e14$ Neutrons per cmsquared per second	23
2.5	24
3.1 Iron-Chromium-Nickel Phase Diagram	27
3.2 Steel Cr-Ni Schaeffler Diagram	27
3.3 Expected DPA during component life time and operating temperatures[25]	30
3.4 Energy transfer and neutron scattering angle	30
3.5 Grain boundary concentrations at $200^{\circ}C$ to $500^{\circ}C$	31
3.6 Chromium grain boundary depletion	32
3.7 Energy transfer and neutron scattering angle	32
3.8 Inter Granular Stress Corrosion Cracking in Nickel Alloy[35]	34
3.9 Alloy choices for early LWRs	34
3.10 Nickel, IGSCC and TGSCC	35

4.1	A cross section of the HFIR [48]	39
4.2	Neutron Flux in the HFIR while at 85MW [49]	39
4.3	Neutron spallation source [50]	41
4.4	Neutron energy spectrum [50]	41
4.5	Neutron Spectra from the Fission of U235[52]	42
4.6	Cross section: Fe56 to Ni58	43
4.7	TALYS work flow [54]	44
4.8	TALYS Fe54-Co55 cross section comparison with experiment [55]	45
4.9	An example decay chain from an unstable parent isotope, through unstable daughter isotopes ending with a stable daughter isotope.	47
4.10	One hundred simulated 13MeV proton energy loss curves in Fe simulated with SRIM [57]	48
5.1	Time and Size Scales for Computer Packages [58]	50
5.2	Graph caption	59
5.3	Lennard-Jones Potential	60
5.4	Morse Potential	61
5.5	Buckingham Potential	61
5.6	Graph caption	62
5.7	Polynomial knot to knot spline	68
5.8	Neighbouring atoms with halo	70
5.9	Common approximations used to enable DFT calculations	79
5.10	A useful, albeit incorrect, way of visualising an "infinite" chain of atoms in 1D	82
5.11	A 2D example of a translation by an integer multiple of the unit cell from the origin unit cell	82
5.12	BCC Wigner Seitz Cell and BZ [80]	84
5.13	FCC Wigner Seitz Cell and BZ [80]	84
5.14	Charge density FCC aluminium xy plane at $z = 0.0a_0$	85
5.15	Charge density FCC aluminium xy plane at $z = 0.5a_0$	85
5.16	Replacing the complex potentials with a pseudopotential[84]	88
5.17	Damage cascade: 500kev iron projectiles into an iron target calculated with SRIM	93
6.1	An example of several decay chains including branching factors and possible external source terms for each isotope on each chain.	101
6.2	Decay of Po-218: Analytic and Gaver-Stehfest Calculations [56]	104
6.3	Flow chart of major processes in the Activity code	105
6.4	36MeV Ion Track	106
6.5	36MeV Ion Track	106
6.6	Cyclotron layout - main hall and beam lines	107

6.7	Gamma warning alarm during 0.5 microamp 36MeV proton irradiation of iron	108
6.8	High Purity Germanium Detector	108
8.1	Decay path of Cobalt-55	120
8.2	Cobalt-55 Gamma Intensity Plot	120
8.3	Energy vs target depth - protons through a 0.5mm thick iron target	121
8.4	Ion trajectory - 0.5mm thick iron target	121
8.5	Ion energy lost in 0.5mm thick target	121
8.6	Energy vs target depth - protons in iron target (1m thick)	122
8.7	Ion trajectory - iron target (1m thick)	122
8.8	Ion energy lost in target (1m thick)	122
8.9	Isotopes with the top 5 radioactivity plotted over approximately 3 days	124
8.10	Predicted gamma lines approximately 3 days after irradiation	124
8.11	Total activity over time of the proton irradiated iron target	124
8.12	Saturation curve for Co55 with the current beam settings	124
8.13	Energy vs target depth - protons in iron target thick enough to stop all ions	125
8.14	Ion trajectory - iron target thick enough to stop all ions	125
9.1	Equation of state fit through data points for Chromium BCC with no magnetism, ferromagnetic and antiferromagnetic configurations	130
9.2	Equation of state fit through data points for Iron BCC (no magnetism, ferromagnetic) and Iron FCC (no magnetism and antiferromagnetic)	131
10.1	Parameter Breed Event	148
10.2	DIRAC calculated radial electron density for Iron	156
10.3	DIRAC calculated radial electron density for Palladium	157
10.4	Sheng palladium pair function	158
10.5	Sheng palladium pair function (zoomed in)	158
10.6	Sheng palladium density function	158
10.7	Sheng palladium embedding function	158
10.8	FCC Iron Neighbour Separation	162
10.9	FCC Palladium Neighbour Separation	162
12.1	DFT Aluminium Cohesive Energy	170
12.2	DFT Aluminium Surface Energy	170
C.1	Decay of Po-218: Analytic and Gaver-Stehfest Calculations [56]	227
D.1	Decay of Po-218: Analytic and Gaver-Stehfest Calculations [56]	238
D.2	Decay of Po-218: Analytic and Gaver-Stehfest Calculations [56]	239

D.3	Decay of Po-218: Analytic and Gaver-Stehfest Calculations [56]	240
D.4	Decay of Po-218: Analytic and Gaver-Stehfest Calculations [56]	241
D.5	Gradual saturation of steel with radioactive isotopes under proton irradiation	267
H.1	Percentage of Annual Dose Equivalent	290
H.2	Activity of all isotopes over time	290
H.3	Ion trajectory in the target	291
H.4	SRIM output summary	291
H.5	Percentage of Annual Dose Equivalent	292
H.6	Activity of all isotopes over time	292
H.7	Ion trajectory in the target	293
H.8	SRIM output summary	293
H.9	Percentage of Annual Dose Equivalent	294
H.10	Activity of all isotopes over time	294
H.11	Ion trajectory in the target	295
H.12	SRIM output summary	295
H.13	Percentage of Annual Dose Equivalent	296
H.14	Activity of all isotopes over time	296
H.15	Ion trajectory in the target	297
H.16	SRIM output summary	297
H.17	Percentage of Annual Dose Equivalent	298
H.18	Activity of all isotopes over time	298
H.19	Ion trajectory in the target	299
H.20	SRIM output summary	299
I.1	Ecutwfc and Ecutrho convergence for aluminium (energy and force)	302
I.2	2D density map - force and energy convergence with k-points and smearing - aluminium	303
I.3	Ecutwfc and Ecutrho convergence for iron (energy and force)	304
I.4	2D density map - force and energy convergence with k-points and smearing - iron	305
I.5	Ecutwfc and Ecutrho convergence for palladium (energy and force)	306
I.6	2D density map - force and energy convergence with k-points and smearing - palladium	307
J.1	Aluminium FCC equation of state plot	309
J.2	Aluminium FCC elastic constants plot	309
J.3	Iron BCC equation of state plot	312
J.4	Iron BCC elastic constants plot	312
J.5	Iron BCC (magnetic) equation of state plot	316

J.6	Iron BCC (magnetic) elastic constants plot	316
J.7	Iron FCC (magnetic) equation of state plot	320
J.8	Iron FCC (magnetic) elastic constants plot	320
J.9	Palladium FCC equation of state plot	324
J.10	Palladium FCC elastic constants plot	324
K.1	Ackland Mendelev Pair	329
K.2	Buckingham Potential	329
K.3	Cubic Knot Spline	330
K.4	Cubic Knot Spline Fixed End	331
K.5	Cubic Knot Spline Fixed End Pair	332
K.6	Ackland Embedding	333
K.7	Ackland Embedding	334
K.8	Ackland Embedding	334
K.9	Embedding A	335
K.10	Embedding B	335
K.11	Embedding C	336
K.12	Embedding D	337
K.13	Ackland Embedding	337
K.14	Embedding F Function Plot	338
K.15	Embedding G Function Plot	338
K.16	Lennard-Jones	339
K.17	Morse	340
K.18	Quadratic Density	340
K.19	ZBL	341
K.20	Zero	342
L.1	Fe-Fe Pair Function	346
L.2	Fe-Fe Pair Function (Zoomed In)	346
L.3	Fe Density Function	346
L.4	Fe Embedding Function	346
L.5	Pd-Pd Pair Function	346
L.6	Pd-Pd Pair Function (Zoomed In)	346
L.7	Pd Density Function	346
L.8	Pd Embedding Function	346
L.9	Fe-Pd Pair Function	347
L.10	Fe-Pd Pair Function (Zoomed In)	347

L.11 Iron equation of state	348
L.12 Iron elastic constants	348
L.13 Palladium equation of state	348
L.14 Palladium elastic constants	348
L.15 Iron cohesive energy	348
L.16 Iron cohesive energy (zoomed in)	348
L.17 Palladium cohesive energy	349
L.18 Palladium cohesive energy (zoomed in)	349
L.19 Iron surface energy	349
L.20 Palladium surface energy	349
M.1 Fe-Fe Pair Function	353
M.2 Fe-Fe Pair Function (Zoomed In)	353
M.3 Fe Density Function	353
M.4 Fe Embedding Function	353
M.5 Pd-Pd Pair Function	354
M.6 Pd-Pd Pair Function (Zoomed In)	354
M.7 Pd Density Function	354
M.8 Pd Embedding Function	354
M.9 Fe-Pd Pair Function	354
M.10 Fe-Pd Pair Function (Zoomed In)	354
M.11 Iron equation of state	355
M.12 Iron elastic constants	355
M.13 Palladium equation of state	355
M.14 Palladium elastic constants	355
M.15 Iron cohesive energy	355
M.16 Iron cohesive energy (zoomed in)	355
M.17 Palladium cohesive energy	355
M.18 Palladium cohesive energy (zoomed in)	355
M.19 Iron surface energy	356
M.20 Palladium surface energy	356

Chapter 1

Introduction

Several generation III+ nuclear reactors have been proposed for construction in the UK and generation IV nuclear reactors being researched and developed. New materials are required to withstand the extreme conditions in and around the core of these reactors. Austenitic stainless steels have been an important structural material in the industry, and may continue to be so, providing the problem of Inter Granular Stress Corrosion Cracking (IGSCC) can be addressed. Doping these steels with platinum group metals has been seen to reduce IGSCC, but the effects are unknown for these steels in a radiation field.

1.1 Motivation

I have often questioned the motivation behind this work, and at times it has been a challenge to see the wood through the trees. To help focus, I restated to myself, as simply as possible, the motivation for my work.

Mass produced steels are not perfect repeating crystals, but are made up of small grains. Chromium is added to steel to make stainless steel, and this is more resistant to corroding than steel. When steel is in a nuclear reactor, it will have to perform in extreme conditions.

- high temperatures
- strain caused by high pressures
- radiation damage and strains resulting from this damage
- corrosive environments while in the reactor
- corrosive environments out of the reactor (e.g. fuel cladding in storage)

Radiation damages the steel in a number of ways, including directly knocking atoms out of their place within the lattice structure of the steel as well as changing the isotopes that make up the steel into new isotopes. One example is a neutron reacting with an iron atom, transmuting it into a cobalt atom.

In time, the radiation causes the percentage of Chromium at the surface of the grains to drop, and as it falls the steel loses its protection from corrosion at the boundary between the grains it is made of.

This work is divided into two parts.

1.1.1 Part 1: Activity Computer Program

The materials must be tested before being used in a nuclear reactor. One way to do this would be to place samples of the steel into a test reactor. This is expensive and, as a by-product, the steel sample becomes radioactive. It is difficult to create a large number of neutrons, but it is much easier, and cheaper, to create a beam of protons. Protons can be accelerated in a machine such as the Cyclotron at the University of Birmingham.

The damage that protons cause to a sample is not precisely the same as that caused by neutrons, but it is a cheaper alternative and is a trade-off between the cost and results. One side effect that proton irradiation shares with neutron irradiation is the creation of radioactive waste.

The first part of this work investigates exactly how radioactive a sample becomes when irradiated with a proton beam. An existing set of equations, named after Mathematician Harry Bateman, were modified, and a computer program was created to perform the calculation. The user inputs the constituent elements that make up the material, the ratio of these elements, and the irradiation settings. The program then estimates how radioactive the sample will be and the predicted gamma energies.

1.1.2 Part 2: Palladium-Iron Potential

Adding Chromium to make stainless steel is not the only way to make a steel resistant to corrosion. Adding metals such as Molybdenum and Palladium to steel can increase the resistance to corrosion, but Molybdenum is several hundred times the cost of iron ore, and Palladium is thousands of times as expensive.

Simulating radiation damage using a computer is now a feasible and sensible way to investigate how these materials will be affected by radiation damage, and the simulations may reveal insights that experiments are not able to show, either because they happen on too small a timescale or within the material at too small a scale.

Key to the simulations success is being able to accurately calculate how the atoms interact with one another. The second part of this work concentrates on deriving a mathematical description of how Palladium and Iron atoms interact with one another, which would then allow future simulations of steel with small amounts of Palladium added to it.

Radiation causes chrome to be depleted at the grain boundary. If these simulations show that Palladium is not depleted, it would suggest that the corrosion resistance is maintained, despite the decrease in the amount of chromium at the grain boundary due to radiation damage.

1.2 Nuclear Power in the UK

1.2.1 Nuclear Fuel

The primary fuel of commercial nuclear power stations since those first built in the 1950s has been Uranium, and in particular the isotope U-235. U-235 is fissile and fissions with a high probability when thermal neutrons are captured, due to its very high thermal neutron fission cross section.

Natural Uranium contains a small percentage of fissile U-235 and it mostly fissionable U-238. U-238 is fissionable, but its fission cross section is much lower than U-235. Many of the design considerations for reactors in operation revolve around the amount of U-235 there is in the fuel and having a sufficient flux of thermal neutrons passing through the fuel to sustain a reaction.

A number of Generation IV designs use the fast neutron spectrum to fission U-238 and other fuels such as Thorium-232 to generate energy and breed other fissile isotopes. By moving away from U-235, this increases the potential fuel stockpile, but also brings its own set of challenges.

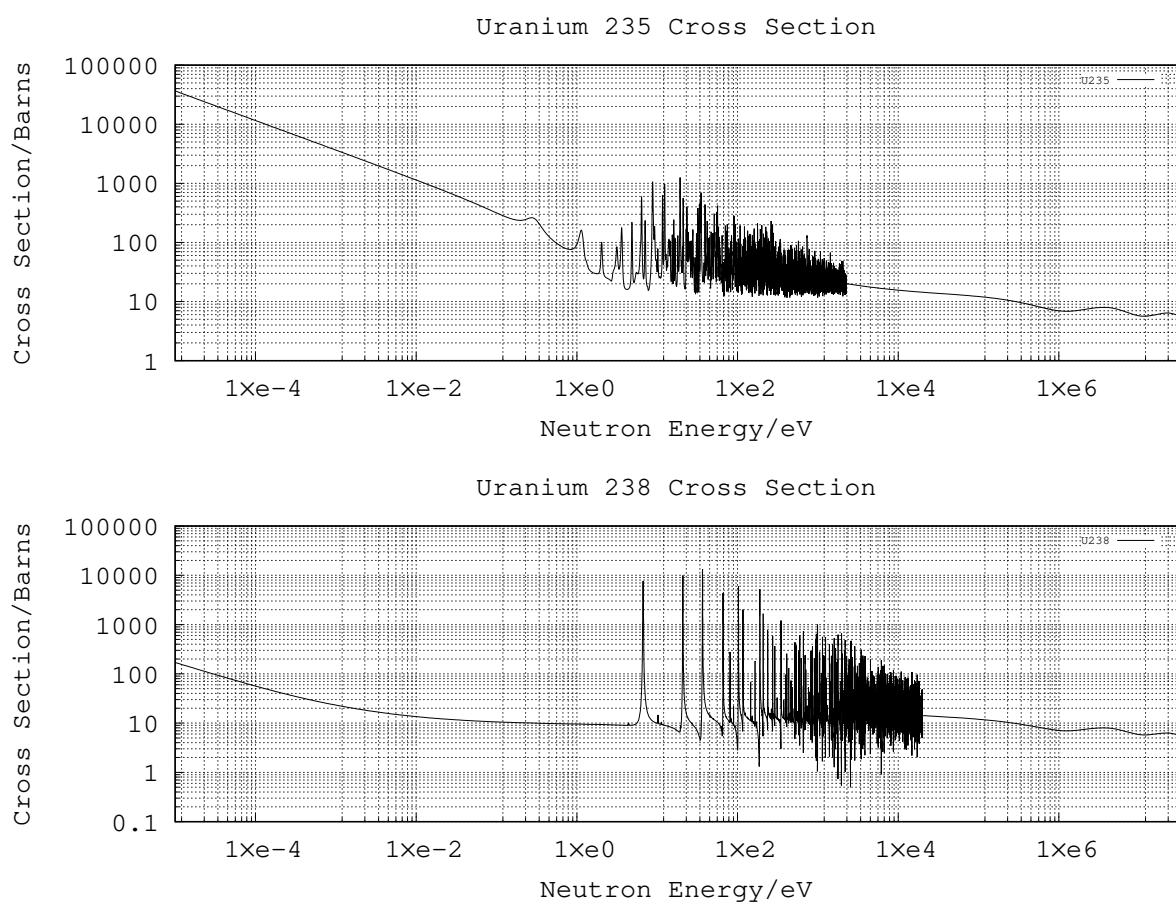


Figure 1.1: graph:U-235 and U-238 Fission Cross Sections

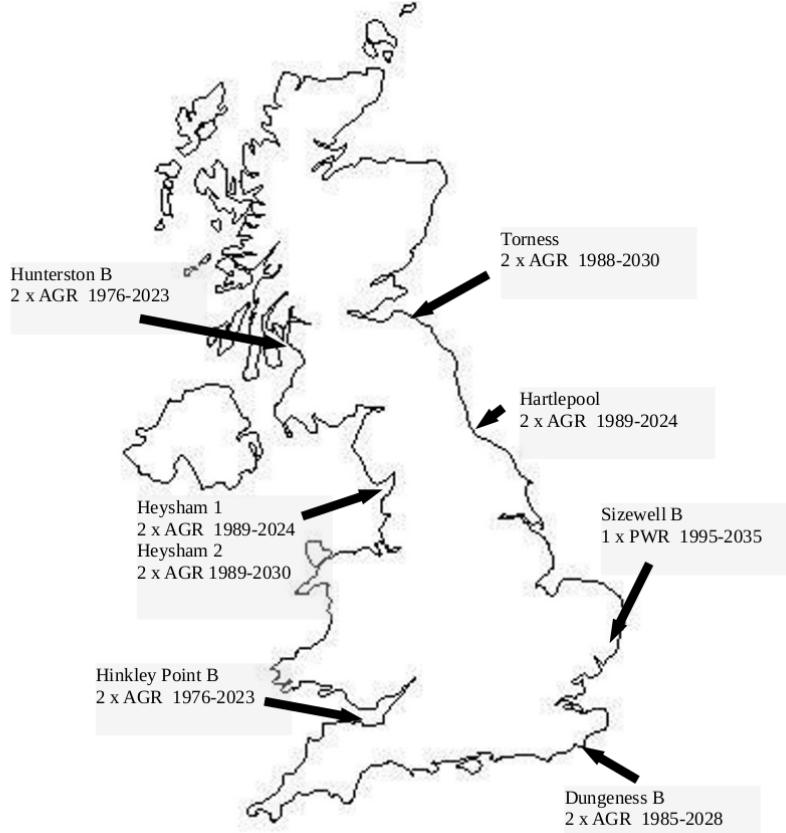


Figure 1.2: Remaining reactor locations in the UK

1.2.2 Generation I Reactors

Magnox type reactors were the first used in the United Kingdom. These reactors used natural Uranium as a fuel and were carefully designed to produce energy despite using an un-enriched fuel. Graphite was used as a moderator, and the low neutron capture cross section of the Magnox cladding allowed a nuclear reaction to occur despite the fuel not being enriched.

In all there were 11 Magnox power stations built in the UK with 26 reactors in total. All have now shut down and the last, Wylfa, closed in 2015.

1.2.3 Generation II Reactors

There are 15 reactors currently operating in the UK, and they are all Generation II reactors. Of these, 14 are Advanced Gas Reactors and 1, Sizewell B, is a Pressurised Water Reactor.

1.2.4 Generation III Reactors

There are no Generation III reactors operating in the UK. The first was built in Japan, and the type of reactor installed was an Advanced Boiling Water Reactor (ABWR) built in 1996 in Japan. Two sites have been proposed for the UK for the Generation III Hualong One type PWR reactor power plant: Sizewell C and Bradwell B, and the expected opening date is between 2030-2035. The final type of operational reactor of this generation is the Russian fast breeder reactor, a single reactor of its type, built in Zarechny, Russia.

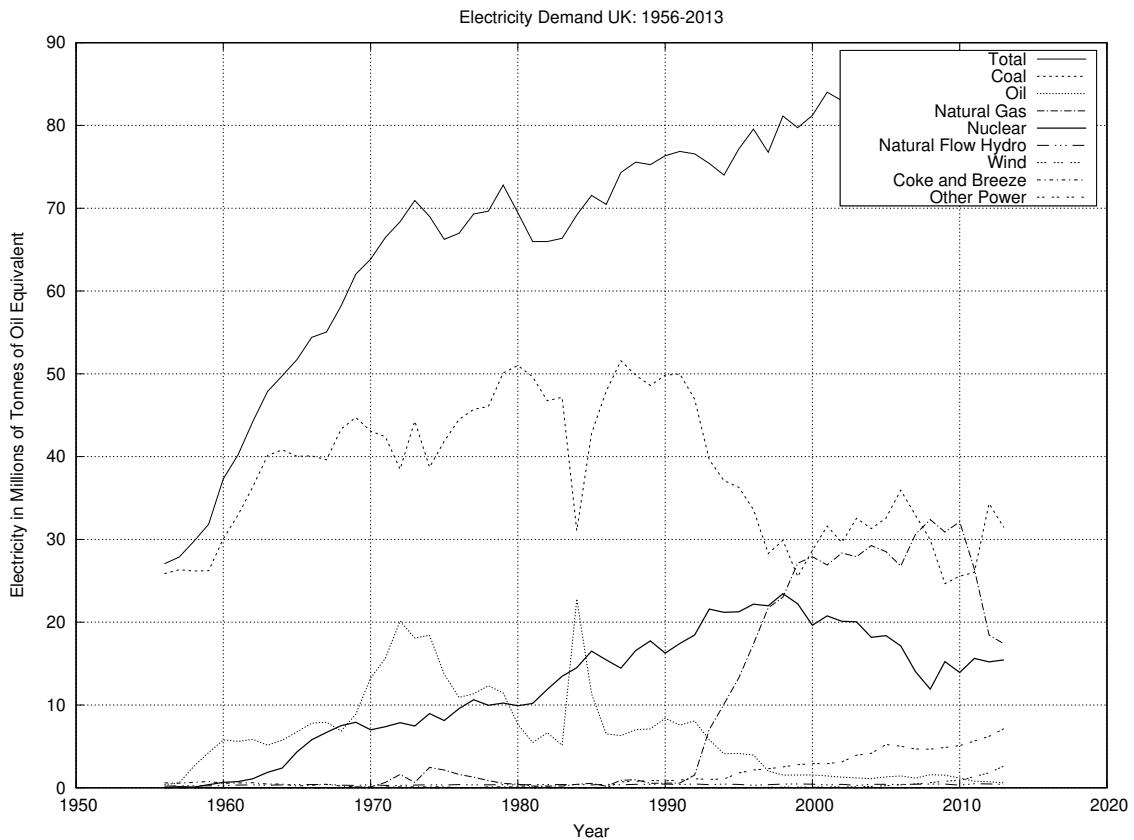


Figure 1.3: Electricity in Millions of Tonnes of Oil Equivalent

1.3 An Approaching Energy Gap for the UK

Since Calder Hall, the first commercial nuclear power plant, opened in 1956, the demand on electrical power generation in the UK has tripled. There is now a reliance on cheap and clean power from nuclear reactors as these provide a quarter of our electricity. There are sixteen reactors operational in the UK: the Magnox reactor at Wylfa and the fourteen AGR reactors are due to be decommissioned by 2023[1], and the remaining PWR reactor, Sizewell B, is expected to remain operational until 2035[1].

There is an obvious concern that within the next ten years the UK will lose a sizeable proportion of its electricity generation capabilities, however, there are proposals to remedy this. EDF have planned to build two new reactors at the Hinkley Point site in Somerset. Hinkley Point C will contain two Areva NP designed Gen III+ European Pressurised Reactor (EPR) reactors. Many advanced materials, including a variety of types of Austenitic Stainless Steel, will be used in the construction of these reactors.

The physical stresses subjected to the Gen IV reactors will go beyond those that are currently being built. There will be higher radiation doses and faster, more damaging neutrons. Coolant temperatures will be higher to either give better thermodynamic efficiency or open the route to creating hydrogen directly as a fuel. Novel coolants such as lead and sodium, each with their own challenges to overcome, are also being considered.

1.3.1 Why choose Nuclear Power?

As a society we need energy, whether it's in the form of electricity or stored chemically, and whilst we are becoming more efficient at using that energy, the demand will remain, unless something drastically changes our society. There is a choice between burning fossil fuels or bio fuels, using energy from the Sun, wind, ground, oceans or rivers and finally using the energy of the nucleus, whether by fission or fusion.

Each has its drawbacks and advantages. Renewable energies are unreliable; wind power only provides energy when there is a wind, and if the wind is too strong, they must be shut down or risk damage. The Sun is only available for a portion of the day, and the duration and intensity change with the seasons, not to mention the impact of clouds on solar power. They aren't very good at responding to demand; there isn't a button to turn up the power of the sun, or increase the velocity of the wind when the national grid demands it. If we were completely reliant on renewables, there would either need to be an efficient way to store energy on a large scale, or many more solar, wind, tidal and hydroelectric power stations than would be needed.

Fossil fuels release carbon dioxide, sulphur dioxide, naturally occurring trace radioactive elements and other pollutants into the atmosphere. These power plants are useful because the fuel is cheap, the waste is put back into the environment without much processing and the energy produced may be varied to meet the demand of the national grid.

Nuclear power has a bad reputation with the public, in part caused by accidents such as the Windscale fire, Three Mile Island, Chernobyl, and Fukushima. There have been examples of poor handling of nuclear waste in the past, such as the legacy storage ponds at Sellafield, and long term storage in geologically stable areas is something that hasn't been achieved to store the existing waste, let alone waste created in the future.

Modern nuclear power plants are very expensive to build, with the proposed 3.2GWe Sizewell C power plant expected to cost 20 billion or more [2]. When compared to 0.5 billion for the 0.884GWe Carrington CCGT Power Station [3], the initial costs are 0.57 billion per GWe for CCGT compared to 6.25 billion per GWe for Nuclear.

There is an effort around the world of countries aiming to reduce the production of carbon dioxide as a result of burning fossil fuels. When a power plant is operational, the carbon dioxide produced by nuclear power is negligible when compared with gas, oil and coal power plants.

As a species, we have used combustion of chemicals to generate heat for thousands of years, culminating in high efficiency CCGT power stations, with efficiencies above 60

Future designs may help to tackle the problematic waste generated in the past, and by using fertile material to produce fissionable elements other than U²³⁵, the amount of fuel available to reactors will increase.

1.3.2 Proposed Generation III+ Nuclear Power Plants

Two Generation III+ reactors are under construction in the UK: Hinkley Point C 1 and C 2. These reactors are EPR, which are PWRs, and the proposed opening years are 2025 and 2026 respectively.

Areva EPR

The 1.6GW Areva EPR design has four primary loops transferring heat by pressurised water from the reactor to heat exchangers. It requires Uranium enriched to 5% 235U in the form of Uranium Oxide Pellets. The inlet temperature is 295.6°C and the outlet temperature is 329.8°C.

There are 241 fuel assemblies, each containing 265 fuel rods giving a total of 63865 rods. The fuel rod cladding is made from 316 stainless steel and has an inner diameter of 7.72mm and an outer diameter of 9.68mm. The materials used to construct the control rod drive mechanisms includes 410 stainless steel and 304 stainless steel.

There are two EPR reactors planned for the proposed Sizewell C power plant.

Westinghouse AP1000

The 1.1GW Westinghouse AP1000 design is a pressurised water reactor with two primary loops transferring heat from the reactors to heat exchangers. An enriched Uranium Dioxide fuel, up to 5% 235U, is clad in Zircaloy.

(a proprietary Zirconium alloy).

Zirlo is an alternative cladding material to 316 stainless steel, and as a cladding material is absorbs fewer neutrons than steel cladding.

	316SS ($FeCr_{20}Ni_8Mo_1$)	Zirlo (Hill Approximation)
Bulk Modulus (GPa)	164.9 [4]	98.5 [5]
Shear Modulus (GPa)	74.6 [4]	33.2 [5]
Young's Modulus (GPa)	194.3 [4]	89.7 [5]
Poisson's Ratio	0.30 [4]	0.35 [5]

Table 1.1: Bulk, Shear, Young's modulus comparison: Zirlo and austenitic stainless steel

The inlet temperature is $279.4^{\circ}C$ and the outlet temperature is $324.7^{\circ}C$ which is close to the temperature range of the EPR. The composition of the control rod absorber material includes 304 Stainless Steel.

Although work has not started, at the time of writing, several sites have been acquired with the aim of building new nuclear power stations. There are five sites and three reactor designs[6]:

- Hinkley Point: two Areva EPRs (EDF Energy)
- Sizewell: two Areva EPRs (EDF Energy)
- Wylfa: 2-3 Hitachi ABWRs (Horizon Nuclear Power)
- Oldbury: 2-3 Hitachi ABWRs (Horizon Nuclear Power)
- Sellafield: 3 Westinghouse AP1000s (NuGeneration) ...

1.3.3 Generation IV

Goals of Generation IV Reactors

The GenIV International Forum has put forward four main goals for this next generation of nuclear power[7]:

- sustainability
- safety and reliability
- economics
- proliferation resistance and physical protection

The selection of known materials, and the development of new materials, will play a key part in all four of these goals.

Carnot's Theorem

Whatever the motivation, whether it's to increase profits or to supply energy at greater amounts and for less cost to the consumer, increasing the efficiency of making energy is critical. Solar panels are continually being improved to edge closer to their theoretical limit, and wind turbines are being constructed larger and with more advanced materials to extract as much energy from the wind as possible.

In the same way, powerplants that use heat are constantly being developed to improve efficiency. In the nineteenth century Carnot showed that the maximum possible efficiency of a heat engine is determined by the difference in temperature between the heat reservoirs.[8]

$$\eta_{max} = 1 - \frac{T_c}{T_h} \quad (1.1)$$

To increase maximum efficiency, the temperature difference should be increased, and this leads to higher temperature reactors. There will be a trade off between the increased temperature, the ability of the materials to withstand the temperature, health and safety considerations, lifetime of components, the effect on the coolant and more.

The first generation of reactors in the UK were the gas cooled Magnox reactors. With core temperatures of approximately $350^{\circ}C$ [9], the thermodynamic efficiency was relatively poor. This was limited by the magnesium oxide cladding, which was in turn selected due to the fuel. Combined cycle gas turbines have much higher temperatures within their turbines, and the temperature of the steam within the secondary steam turbine can reach $580^{\circ}C$ [10], significantly higher than in a Magnox reactor.

The second generation of power plants, particularly in the UK, included the advanced gas reactor that used carbon dioxide as a coolant. This reactor design increased temperatures to over $600^{\circ}C$, and thus the maximum possible thermodynamic efficiency was increased.

Several Generation IV reactors have designed operating temperatures that exceed those of the AGR, and this brings a new set of challenges to overcome.

1.3.4 Generation IV Proposed Designs

Fast Reactors

Examples of thermal neutron reactors include Magnox, AGR, LWR and Candu reactors. These reactors contain moderators, designed to slow neutrons, decreasing their energy to thermal temperatures and leveraging the large neutron fission cross section of Uranium-235. The cross section for fast neutrons and U-235 is in the region of 1 barn, but thermal neutrons and U-235 have a fission cross section of over 1,000 barns.

Natural Uranium contains 0.7% U235 and 99.3% U238[11]. The thermal fission cross section for U238 is small for thermal neutrons, and can be measured in the microbarn to millibarn range, but it has a much higher fission cross section with fast neutrons with a cross section closer to 1 barn for neutrons with an energy of 1MeV or more.

Fast neutron reactors have been tested and used since the 1950s, and there are several Generation IV reactors based on this approach. Given the much larger percentage of U238 in natural Uranium, Fast Neutron Reactors have a larger stockpile of fuel. They also breed fissile isotopes, Plutonium 239 and 241. This has its advantages and disadvantages. The advantage is clear, as it produces fissile fuel as it runs, but the creation of fissile materials is a concern as these isotopes could be used in nuclear weapons.

Lead-cooled Fast Reactors

Light elements such as the hydrogen in water molecules, or carbon in a graphite stack, allow neutrons to efficiently lose energy. When neutrons scatter inelastically with heavier elements, such as lead, they lose a much smaller percentage of their energy.

The LFR uses lead as the coolant. Lead has a low reaction cross section and, because of its mass relative to a neutron, the kinetic energy lost is low, keeping more neutrons in the fast energy group. As the coolant is a liquid, and the boiling point of lead is over $1,700^{\circ}C$, the system will run at atmospheric pressures and any problems associated with void formation due to boiling is removed.

The molten lead itself will act as a gamma shield, and as a coolant it will be chemically less reactive than the coolant in a Sodium-Cooled Fast Reactor (SFR) or Supercritical-Water-Cooled Reactor (SCWR).

There are several disadvantages to using lead as a coolant. The melting point of lead is 327°C , so the reactor would need to be heated for the coolant to become a liquid. This also restricts the lowest temperature and thus the maximum efficiency possible that may be extracted from the system without the coolant forming a solid. The density of lead also poses a problem for the structure needed to support the reactor.

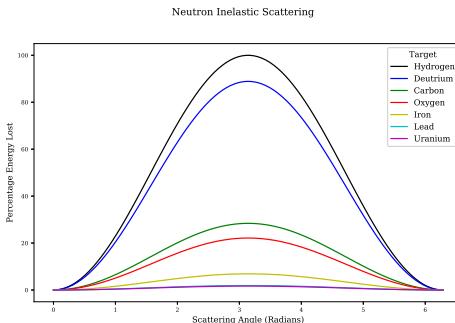


Figure 1.4: Inelastic scattering - scattering angle vs energy loss for a range of target atoms

European Lead Fast Reactor (ELSY) is a 600MWe Lead-Bismuth eutectic cooled fast neutron reactor. It has a lower melting point than lead, but there are concerns for the transmutation of Bismuth to Polonium. The 9,000 tonne molten lead coolant and the high flux of fast neutrons[12] will be challenging conditions to overcome.

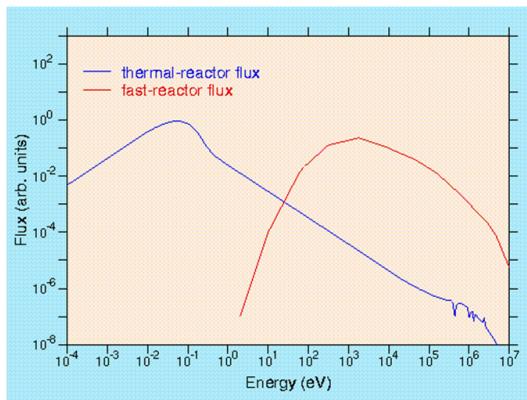


Figure 1.5: Typical Reactor Flux - Thermal and Fast Spectra

VHTR

Traditional nuclear power plants, as well as coal and the secondary cycle of CCGT plants, boil water to drive turbines. Very-High-Temperature reactor designs use thermal neutrons and a fissile fuels. Designs plan to have outlet temperatures of up to $1,000^{\circ}\text{C}$ [13]. With a helium coolant, there are several options.

- directly drive turbines
- heat water to create steam and drive turbines
- use the high temperature to help create hydrogen, and combine with steam driven turbine

Hydrogen may be extracted from high temperature water either by a thermo-chemical, high temperature electrolysis or hybrid process[14], and they all benefit from the very high temperatures provided by the predicted outlet temperatures.

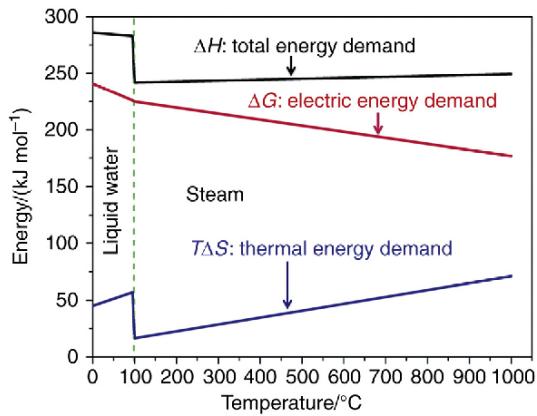


Figure 1.6: Energy demand to produce hydrogen by electrolysis as temperature changes

The Pebble-Bed Reactor (FBR) is a type of VHTR where spherical fuel pellets are held in a hopper shaped reactor. Temperatures within the core may reach $1,500^{\circ}\text{C}$, and this will pose a challenge for materials scientists.

Gas Cooled Fast Reactors

This design is similar to the VHTR, but rather than use thermal neutrons, it will use fast neutrons. The design will use a ceramic core and have outlet temperatures up to 850°C . This will give similar options to the VHTR where high temperatures would be used to assist in the production of hydrogen, or it would be used to power a turbine with a secondary steam powered circuit similar to that of a CCGT.

There are challenges ahead, including the development and testing of advanced materials that can withstand these temperatures while under high energy neutron flux. However, unlike LFRs, SFRs and SCWRs, the coolant is chemically inert and isn't transmuted by neutrons removing the risk of the coolant becoming radioactive.

Sodium Cooled Fast Reactors

Memories of small amounts of sodium, kept under oil in school chemistry labs, being carefully cut and prepared ahead of a violent reaction with water, probably come to mind at the first mention of an SFR. It may not be the first coolant choice that comes to mind, but it does have advantages.

Sodium has a low melting point, of just under 100°C , and a boiling point of 880°C . With an operating temperature of up to 550°C , the reactor would be able to run at atmospheric pressure and without the issues associated with the coolant boiling.

There have been several reactors built and operated using this technology, such as the Russian BN-600 reactor. This particular reactor has a sodium primary loop which heats a secondary sodium loop which in turn heats a third loop for water and steam.

Supercritical-Water-Cooled Reactor

At 647K and a pressure of 22.1MPa (approximately 218 atmospheres), water becomes supercritical which is neither a liquid or a gas[15]. The current generation PWR and BWR operate at approximately $228 - 325^{\circ}\text{C}$ at 15.2MPa [6] and $278 - 287^{\circ}\text{C}$ at 7.1MPa [16] respectively.

Supercritical water exists above 374°C and 22.1MPa , and in this state water has a higher thermodynamic efficiency. The design of the nuclear power plant is also simplified as there is no phase change of the water, so a condenser is not needed. The SCWR is the only GEN IV reactor design that uses water as the coolant[1]. The

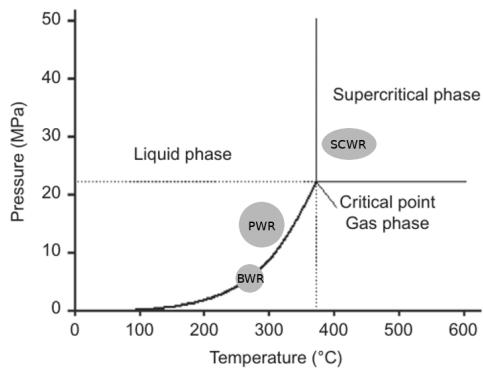


Figure 1.7: Phases of Water: Pressure vs Temperature

economic benefits have already been seen in SCW fossil fuel power stations, and it is incorporated in GenIV water cooled fast and thermal reactors.

There are side effects to using supercritical water. It is an oxidant and is now being considered to "burn" organic waste in future waste processing plants. The combination of supercritical water chemistry and irradiation damage must be considered, as well as higher temperature and pressure, when considering the materials used by a SCWR.

Molten Salt Reactor

MSRs have been operated for over 50 years. Chemically, salts are very stable, and this has obvious safety benefits over reactors like the SFR. The coolants are fluoride salts that have high boiling points which gives the added safety protection of being able to operate at atmospheric pressure, unlike a PWR where a pressure vessel is needed.

The fuel is dissolved into the molten salt. There are no solid fuel rods to place into the core, remove or reprocess. The molten salt is processed on the site to remove poisons, waste and add new fuel.

MSRs may use either thermal or fast neutrons and a range of fuels including Thorium. This is particularly interesting as Thorium is three times as abundant as Uranium and would increase the fuel available to us to use to generate nuclear power.

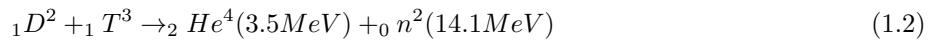
A safety feature that takes advantage of the fuel being dissolved in the molten salt are large drain tanks under the reactor. A plug is designed to melt if it reaches a certain temperature, that would quickly drain the molten salt from the reactor, dumping it into large, cold drainage tanks when reactivity would drop and the fuel would cool reverting to a solid salt.

Salts considered in various designs are chlorides, nitrates and fluorides. Corrosion of the reactor due to the molten salts is a concern. Elements that provide protection to corrosion, such as Chromium, are prone to dissolution into the molten salt. Where the fuel is suspended in the molten salt as Tristructural Isotropic (TRISO) fuel particles, formation of chromium-carbide precipitates may be an issue.

Experimental Fusion Reactors

Nuclear Fusion is a very attractive technology and could be the answer to all of our energy problems. Much work is being invested in developing this technology and the ITER (International Thermonuclear Experimental Reactor) has been designed to output more energy than is required to start the fusion reaction. The process of fusion combines two isotopes of hydrogen and leaves helium and fast neutrons. As neutrons have no charge,

they can penetrate shielding causing damage as they lose energy through nuclear interactions. Any atoms they interact with have a chance to capture the neutron and become unstable.



The fast neutron spectrum for fission reactors ranges from a few eV to a few MeV, whereas the neutrons in a fusion reaction have 2-3 times more energy than the most energetic neutrons from fast fission. Engineers must develop materials to construct components that will be resilient to this damage, while having a low reaction cross section and being able to withstand other extreme conditions within the reactor.

1.4 Damage to Nuclear Core Components by Neutrons

Radiation damage in a reactor is caused by several projectiles. Fission fragments are heavy, contain charged particles and lose energy close to their source. Gamma rays ionise may excite atoms and electrons, but do not have the momentum to knock atoms out of place. Fast neutrons, however, may impart a great deal of momentum to a target atom and, as they are neutral, travel much further into a material than fission fragments.

One fuel source for many of the Gen III+ and Gen IV reactors is Uranium-235, whether as enriched Uranium or otherwise. The neutron/s released by the fission of Uranium-235 atoms have a spectra of energy which may roughly be split into four categories: cold (below 0.025eV), thermal (0.025eV), slow and intermediate (above 0.025eV and below 1MeV) and fast (1MeV and above).

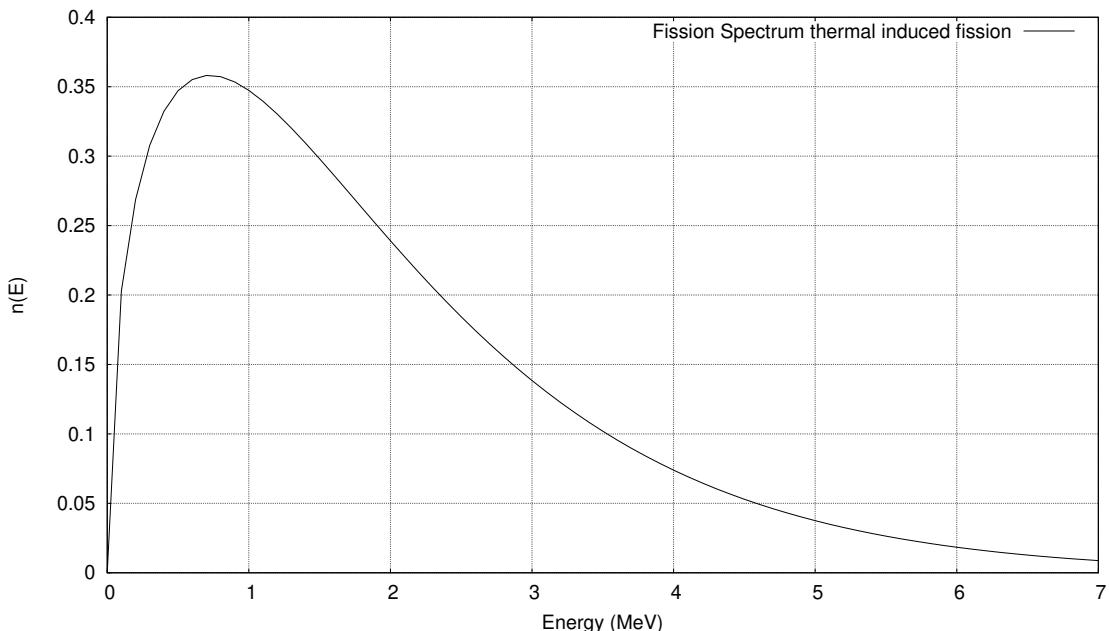


Figure 1.8: Neutron Energy Spectrum from Fission

Thermal and slow/intermediate neutrons cause damage in their own particular way, as they are captured by and transmute the atoms within the target material. Fast neutrons have enough energy to create a great deal of damage, whereby they transfer kinetic energy to an atom in the material that become the Primary Knock-On Atom (PKA) in a cascade of atoms through the material. Neutrons can travel deep into a material, creating many damage cascades through the material until they lose enough energy or pass through the material.

1.5 Radiation Damage: Replacing Neutrons with Protons

It is difficult to generate large fluxes of neutrons. When we pick up a cup, press keys on a keyboard, they react because of the electromagnetic force, for example, pushing the atoms in our hand away from the atoms in the cup. Neutrons have no net charge, so they cannot be controlled in a similar way. Nuclear reactors are most common way to create large fluxes of neutrons.

This an expensive method, and one that make be inconvenient as the sample being tested would need to be placed inside a reactor. Rather than do this, ions may be used to replicate the damage cause by neutrons, but in a more controlled and accessible way.

A major side effect from the process of nuclear fission is the creation of both radioactive fission fragments and radioactive isotopes within the components and stuructural material of the reactor. Low energy Protons are not captured as low energy Neutrons would be, due to the opposing force between the proton and nucleus of the target atom. Once the proton energies exceed a few MeV, they have sufficient energy to transmute target nuclei.

There are a number data files available

Evaluated Nuclear Data Files

Engineers and Scientists working for or researching in the nuclear industry need accurate data for a wide range of behaviours and properties of isotopes. There is no magic formula to return the requested data for a given isotope, and this is why there is a need for ENDF files. Experimental data

PADF

The Proton Activation Data File was released in 2007 and contained nuclear reaction data for 2355 target nuclei, ranging from Magnesium (12) to Radon (86) with proton energies up to 150MeV. The file contains the sum of all individual reactions as well as certain yields, and the data was generated using the TALYS and ALICE/ASH codes, as well as experimental data from Exfor[17].

Listing 1.1: A sample of the Iron-56 PADF data file

```
1      Proton Activation Data File          528 1451 12
2      ****
3      ****
4      Authors: C.H.M.Broeders, U.Fischer, A.Yu.Konobeyev, L.Mercatali 528 1451 15
5          528 1451 16
6          528 1451 17
7      Data for PADF file were obtained using the TALYS code [1] for 528 1451 18
8      target nuclei with half-life more than 10 min and using the 528 1451 19
9      ALICE/ASH code [2] for nuclei with 1 sec < T1/2 < 10 min and 528 1451 20
10     available experimental data          528 1451 21
11     MAT numbers are taken according to JEFF-3.1/RDD      528 1451 22
12                           528 1451 23
13     Evaluation for Fe-56 (stable) : TALYS code          528 1451 24
14     Experimental data used for the correction of calculated 528 1451 25
15     excitation functions are taken from Refs.[3-12]       528 1451 26
16                           528 1451 27
17                           528 1451 28
18     File contains          528 1451 29
19                           528 1451 30
20     MF=3  MT=5          528 1451 31
21     Sum of all individual reaction cross-sections (Total 528 1451 32
22     reaction cross-section)          528 1451 33
```

23			528	1451	34
24	MF=6 MT=5		528	1451	35
25	Yields of nuclei in nuclear reactions including		528	1451	36
26	n, p, d, t, He-3, He-4 and photon production		528	1451	37

TENDL

TENDL is a collection of files, each in the ENDF format, of nuclear reaction data generated by the TALYS code, and this is also evaluated against experimental data.

Listing 1.2: A sample of the Iron-56 TENDL-P data file

1	TENDL-2019	DIST-	REV1-	2631	1451	6		
2	----TENDL-2019	Material	2631	REVISION	1	2631	1451	7
3	-----Incident proton	data				2631	1451	8
4	-----ENDF-6	Format				2631	1451	9
5						2631	1451	10
6	TENDL-2019	(TALYS Evaluated Nuclear	Data Library)			2631	1451	11
7						2631	1451	12
8	Fe056	proton general purpose library				2631	1451	13
9						2631	1451	14
10	Author:	A.J. Koning, IAEA				2631	1451	15
11						2631	1451	16
12	Ref.:	A.J. Koning, D. Rochman, J.-Ch. Sublet, N. Dzysiuk,				2631	1451	17
13		M. Fleming, and S. van der Marck, TENDL: Complete nuclear	2631	1451	18			
14		data library for innovative nuclear science and				2631	1451	19
15		technology, Nuclear Data Sheets 155, 1 (2019).				2631	1451	20

In terms of bytes, the TENDL-P database is twenty times the size of the PADF. This is partly due to individual reactions being stored and for a much larger range of reactions. One negative points is that the range of proton energies is not as large as it is for the PADF.

TALYS

TALYS is a computer code, written for Linux and Unix systems, that is used to predict and analyse nuclear reactions. It is also used as a tool to generate nuclear data. It was used to create the two previously mentioned data files (TENDL and PADF).

1.6 Replacing Experiment with Simulations

Irradiating and testing materials irradiated by both neutrons and protons have a number of drawbacks, including expensive facilities and the creation of radioactive waste. Simulations may be used to help understand the processes on a mesoscopic scale and make predictions on how the material in question may behave inside a reactor under certain specified conditions.

Part of this work focuses on the damage of stainless steel by neutrons. The damage process occurs on time scales and sizes too small to capture experimentally and play back to study. The theory is not sufficiently understood, or is perhaps unsolvable with current techniques. Modelling helps bridge the gap between theory and experiment.

1.6.1 Molecular Dynamics

Molecular dynamics models have been used to simulate radiation damage. Damage cascades caused by Primary Knock-On Atom (PKA)s with energies of 10, 20 and 50KeV have been modelled in BCC Tungsten[18], whilst

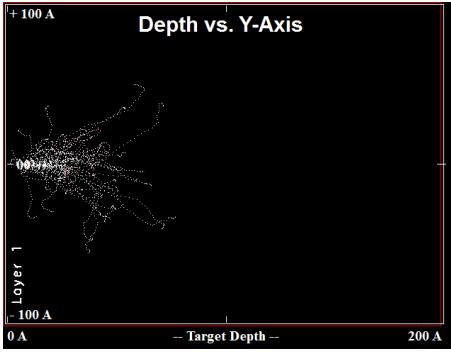


Figure 1.9: 5KeV Iron PKA Damage Cascade in Iron

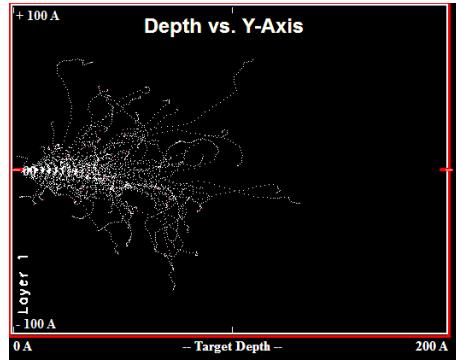


Figure 1.10: 10KeV Iron PKA Damage Cascade in Iron

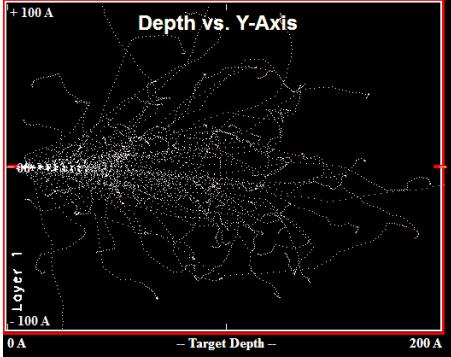


Figure 1.11: 20KeV Iron PKA Damage Cascade in Iron

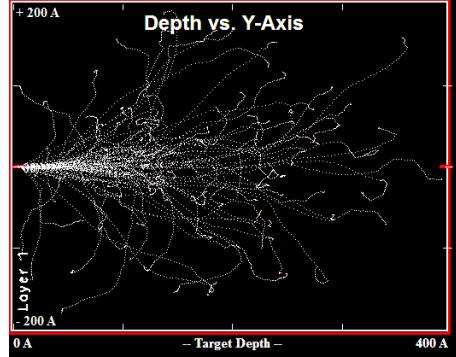


Figure 1.12: 50KeV Iron PKA Damage Cascade in Iron

cascades caused by PKAs of 5, 10 and 20KeV have been simulated in BCC Iron[19].

Neutrons have a long mean path before interacting with a target when compared to ions. It would be impossible with current and foreseeable technology to simulate a neutron travelling through a material because the volume of the material, and the number of atoms and interactions between atoms, would be so large. Rather, focusing on individual damage cascades caused by PKAs would conserve resources.

Molecular Dynamics simulations require interatomic to govern how the atoms in the simulation interact with one another. Typical simulation sizes range from thousands of atoms to millions of atoms.

Iron has a density of approximately 8.5×10^{-2} atoms per cubic angstrom, and this leads to simulation sizes between almost one million atoms and fifty million atoms for cascades with PKAs ranging from 5KeV to 50KeV.

PKA Energy (KeV)	Volume (Ang^3)	Atoms
5	1.0×10^6	9.0×10^5
10	6.0×10^6	5.4×10^6
20	8.0×10^6	6.9×10^6
50	6.4×10^7	5.5×10^7

Table 1.2: Approximate simulation sizes by PKA energy

The interatomic potentials used to simulate a damage cascade must be able to model the very close separations during the collisions event. For very close separations, an exponential potential such as the Ziegler-Biersack-Littmark (ZBL) potential may be used, and for other separations a standard potential such as the Embedded-Atom method (EAM) may be used, in particular for metals.

There have been attempts to create potentials based on Physics, but the nuances are still poorly understood and there aren't any models that can be applied accurately to any material type under any set of circumstances. A number of recent interatomic potentials are fit to match the experimental data instead.

1.6.2 Density Functional Theory

In quantum mechanics, the Schrodinger equation and the wavefunction of a system describes all that can be known about that quantum system. Unfortunately, it is very difficult to compute when there are just several electrons in the system. Many atoms in a system makes this an impossible task with current technology.

Density Functional Theory (DFT) replaces the many electron wavefunction with a single electron wavefunction. Rather than use the positions of all the electrons, which would result in 3^n parameters, the charge density, which varies in space with just 3 parameters, is used.

DFT allows for the calculation of the forces between atoms and the energy of that collection of atoms.

1.6.3 Interatomic Potentials for Molecular Dynamics

Interatomic potentials help to describe the interaction between atoms. Originally they were used for pairs of atoms and a collection of atoms on a pair by pair basis, but in the 1980s they complexity increased to include the effects of the background electron density that the atoms are embedded in.

Whilst DFT is a purer form of computational materials science, more closely linked to physics, it is computationally intensive and suitable for small systems of hundreds to thousands of atoms. By using interatomic potentials to recreate the forces and energies that would be computed by DFT, the systems may be increased in size to thousands to millions of atoms, and these calculations may be run over a time frame with hundreds or thousands of time steps being a possibility.

Chapter 2

Background: Ionizing Radiation

There are a wide range of radiation sources and types of ionising radiation. They interact differently with matter, depending on the type of radiation and its energy. Damage may be caused directly, and target material may be changed chemically, for example water radiolysis, or transmuted, for example the creation of Cobalt-60 in steel in the nuclear industry.

2.1 Radiation Types Relevant to This Work

2.1.1 Introduction

There are three types of radiation that are useful to discuss in this work, and two of these are of particular interest: Neutrons, Ions and Gammas.

2.1.2 Protons and Ions

Charged particles interact with matter through the Coulomb interaction. As a charged particle passes through matter, it may interact with both the nucleus and electrons of an atom. A sufficiently energetic ion may lose kinetic energy to electrons by either raising the electrons to a higher energy levels in the atoms, or by removing electrons from atoms altogether, ionizing atoms.

Ions may also lose kinetic energy to the nucleus of an atom through elastic scattering and, where the atom is in a crystal structure, through knocking atoms in the material out of their lattice positions.

Knock on atoms and electrons with enough kinetic energy that have been removed from atoms (delta rays), continue the irradiation of the material while they have the kinetic energy available to do so.

A large proportion of the energy of a charged projectile is lost to the electrons of the target material. There is a chance, depending on the energy and type of charged projectile, and the cross section of the target nucleus, that the charged particle will overcome the coulomb potential and be captured by the nucleus.

This may result in a stable nucleus or an unstable nucleus. If it is unstable, there is a probability that it will decay releasing energy in the form of photons, nucleons or electrons. The initial ion irradiation creates sources of futher irradiation within the target material.

2.1.3 Neutrons

Neutrons interact with matter differently to that of protons, ions and other atoms, as the Neutron has no overall charge. Neutrons do have a magnetic moment and experience a weak interaction with electrons, but

the dominant interaction is between Neutrons and the Nucleus. There are different methods in which Neutrons interact and these are determined by the kinetic energy, velocity and wavelength of the Neutron.

Name	Energy Range	Velocity/ms ⁻¹	Wavelength Ang
Cold	0-0.025 eV	$0.0 - 2.2 \times 10^3$	> 1.8
Thermal	0.025 eV	2.2×10^3	1.8
Epithermal	0.025-0.4 eV	$2.2 \times 10^3 - 8.8 \times 10^3$	0.5-1.8
Cadmium	0.4-0.6 eV	$8.8 \times 10^3 - 1.1 \times 10^4$	0.4-0.5
Epicadmium	0.6-1.0 eV	$1.1 \times 10^4 - 1.4 \times 10^4$	0.3-0.4
Slow	1-10 eV	$1.4 \times 10^4 - 4.4 \times 10^4$	0.09-0.3
Resonance	10-300 eV	$4.4 \times 10^4 - 2.4 \times 10^5$	0.02-0.09
Intermediate	300 eV - 1 MeV	2.4×10^5	$2.9 \times 10^{-4} - 0.02$
Fast	1-20 MeV	$1.4 \times 10^7 - 6.1 \times 10^7$	$6.5 \times 10^{-5} - 2.9 \times 10^{-4}$
Relativistic	> 20 MeV	$> 6.1 \times 10^7$	$< 6.5 \times 10^{-5}$

Table 2.1: Neutron Categories by Energy Range [20]

2.1.4 High Energy Photons

The electromagnetic spectrum classifies photons based on their energy and/or source, but visible light, x-rays, gamma rays and so on are all the same elementary 'particle'. During the early years of Quantum Mechanics, the relationship between the energy and wavelength of a photon was discovered: the Planck-Einstein relation.

$$E = hf \quad (2.1)$$

Pair Production

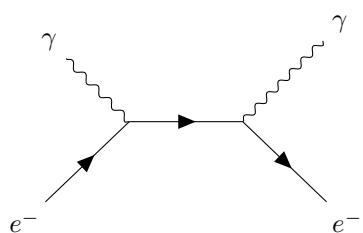
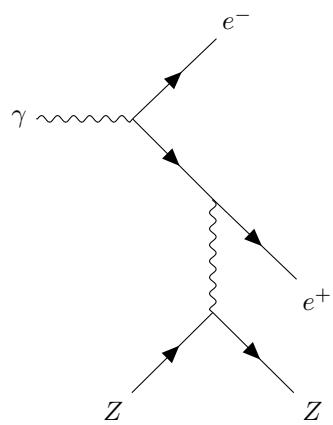
The energy of photons in the database used in this work ranges from 1keV up to almost 10MeV. There are several ways high energy photons will interact with the atoms of a target material. The rest mass of an electron is 511keV. If the photon energy is greater than 1.02MeV, i.e. there is at least enough energy to create an electron and positron, there is a chance that the photon will create an electron-proton pair.

$$hf = (m_e + m_p)c^2 + T_e + T_p \quad (2.2)$$

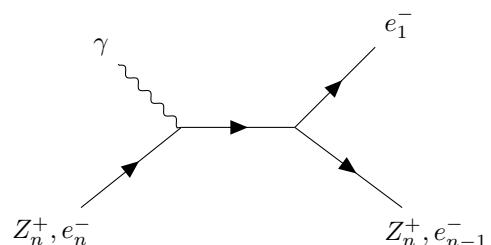
The creation conserves energy and mass, with excess energy carried away as the kintetic energy of the particle pair. The charge before the creation is zero, as the photon is neutral, and the charge after is also zero, with the -1 of the electron and +1 of the positron cancelling out. Angular momentum is also conserved; the photon is a spin 1 Boson and, as electrons and positrons are Leptons they have half integer spin, adding up to 1. Finally, momentum is not conserved in a vacuum, and this is why pair production occurs in the coulomb field of a nucleus. The nucleus carries away excess momentum, fulfilling this conservation law.

Compton Scattering

An incident photon, with enough energy, may interact with the electron of an atom, an it transfers enough kinetic energy to eject the electron from the atom. A lower energy photon is also created that carries away the remainder of the energy, but also linear momentum, as both energy and momentum must be conserved.

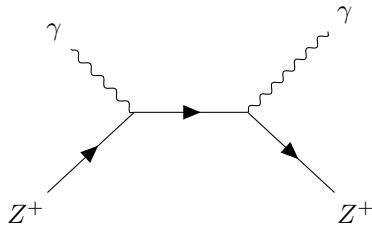


Photoelectric Effect



Coherent Scattering

Lower energy photons



2.2 Effects of Radiation on Organics

2.2.1 Radioactivity, Absorbed Dose, Dose Equivalent and Exposure

There is a distinction between how radioactive a source is and how much radiation is absorbed by an organic due to that source. There are different measures of radioactivity and it's effects, and they may be summarised very briefly as follows:

- Radioactivity of a source - decay events per second (Bq) (also measured in Curie, 3.7×10^{10} decays per second)
- Absorbed dose of any target - energy absorbed per unit mass (Gy) (equivalent of Joules per Kilo)
- Equivalent dose absorbed by tissue/organ - type of radiation is important (Si)
- Effective dose on tissue/organ - how the equivalent dose affects the specific tissue/organ (Si)

One joule of ionising radiation absorbed by a person may not seem a lot of energy, but the average annual doses are so small they are measured in millisieverts per year. A dose of just 5 Sieverts is enough to kill 50% of people within 1 month.

Exposure	Dose (mSv)
Dental X-ray	0.005
Chest X-ray	0.02
UK average annual dose	2.7
Whole body CT scan	10
Nuclear industry employee annual exposure	20
Acute radiation effects	1,000
50% lethal dose within 1 month	5,000

Table 2.2: DFT Settings

In an MCNP5 simulated neutron flux environment, with a thermal flux of $2.5 \times 10^7 n/cm^2 s$ a stainless steel cylinder, 20mm diameter and 70mm irradiated for 10 hours[21]. After 10 minutes cooling, it was calculated to have an activity of over 1.0×10^7 Bq, predominantly due to the creation of Mn^{56} .

The activity in this work seems slightly. The maximum cross section of $Mn^{55} (n, \gamma) Mn^{56}$ is 2.0 barns and the estimated reaction rate at this flux is just over 2.0×10^6 . As the half life of Mn^{55} is short compared to the irradiation time (2.5 hours) the maximum number of Mn^{55} atoms are created, with the creation rate equalling the decay rate.

A simple neutron activation code was developed, which will be discussed in more detail in this work, and the predicted Mn^{55} activity vs time was calculated.

Activity vs Time Mn 56

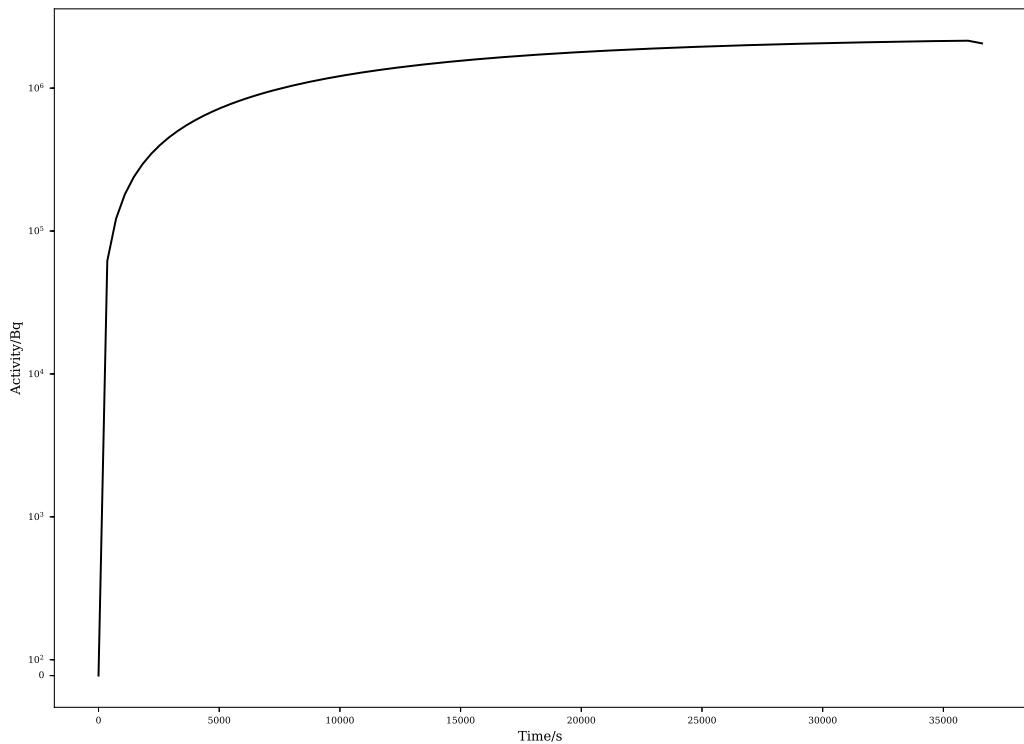


Figure 2.1: Mn 56 Activity

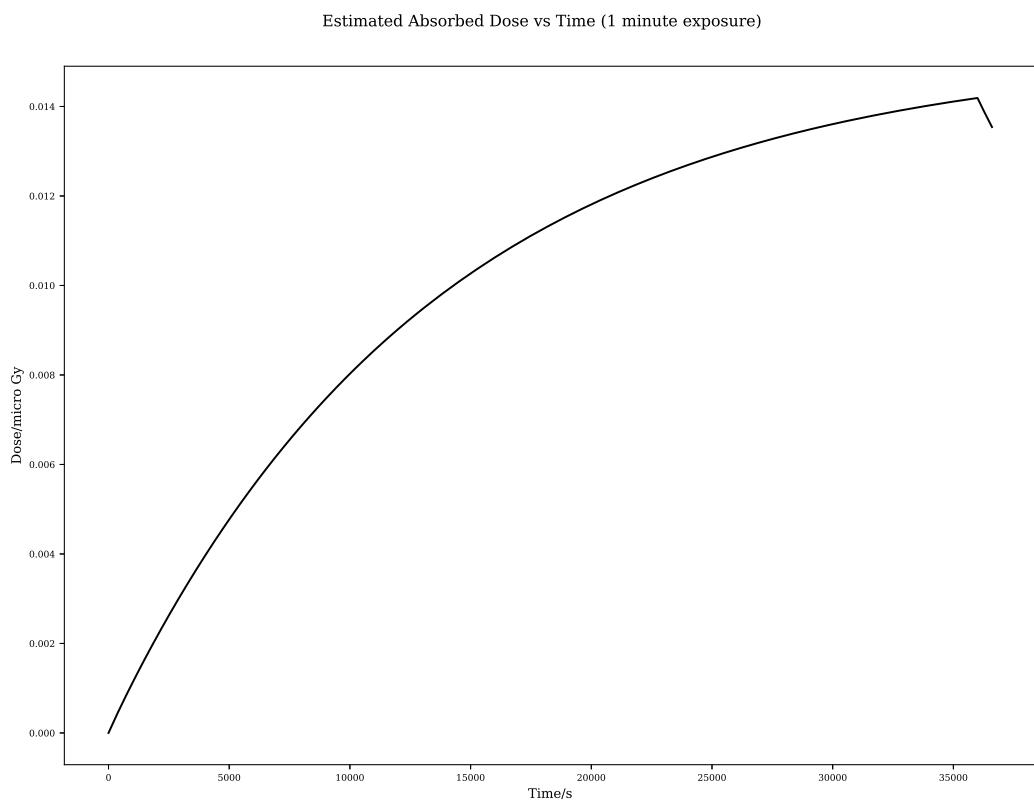


Figure 2.2: Steel Irradiation Gamma Dose $2.5e7$ Neutrons per cmsquared per second

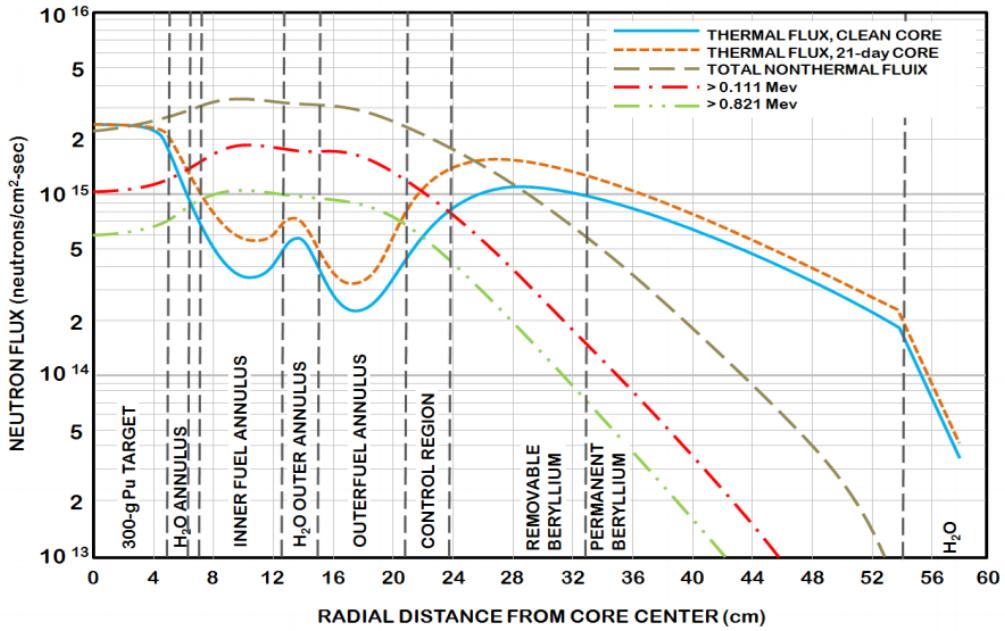


Figure 9: Neutron Flux at the MidPlane of HFIR (at 85MW)

Figure 2.3: High Flux Isotope Reactor Neutron Flux

The overall gamma dose, for 1 minute of exposure to an 80kg person standing 2m from the source was also calculated.

The flux in the within the flux trap of the HFIR is much higher, with high energy neutrons having a flux of $1.0e14$ neutrons per cmsquared per second.

A second calculation was performed for this increased flux, also irradiated over 10 hours. The overall dose per minute was much higher, between 40 and 50 milligrey per minute over the first hour of cooling.

2.2.2 Effect of Radiation on Organics

Life evolved on Earth from single cell to multicelled organisms such as humans. Prokaryotic cells are single celled organisms that do not have a nucleus, an example being bacteria. Our cells are eukaryotic cells, and these have a nucleus that contains the genetic information. Millions of cells die in our body every second, and our body replaces these by replicating living cells. During the replication stage, the DNA within the nucleus is replicated.

DNA is a polymer that is carefully constructed from six components. Stretched out, it is approximately 2m in length and approximately 25 angstrom wide, and it is neatly contained within the cell nucleus which is on average just 6 microns across. The structure of DNA is the well known double helix. The sides of the helix are made from alternating sugars and phosphates, while the "rungs" of the ladder like structure are pairs of either thymine and adenine or cytosine and guanine. These pairs are covalently bonded to the sugar-phosphate sides, and by hydrogen bonds to each other.

Mitosis is the part of the cell cycle where the DNA within the nucleus is replicated and the cell splits into two. There are repair mechanisms, but if the DNA beyond repair, the cell may die through apoptosis, or it may replicate in an uncontrolled way leading to cancer.

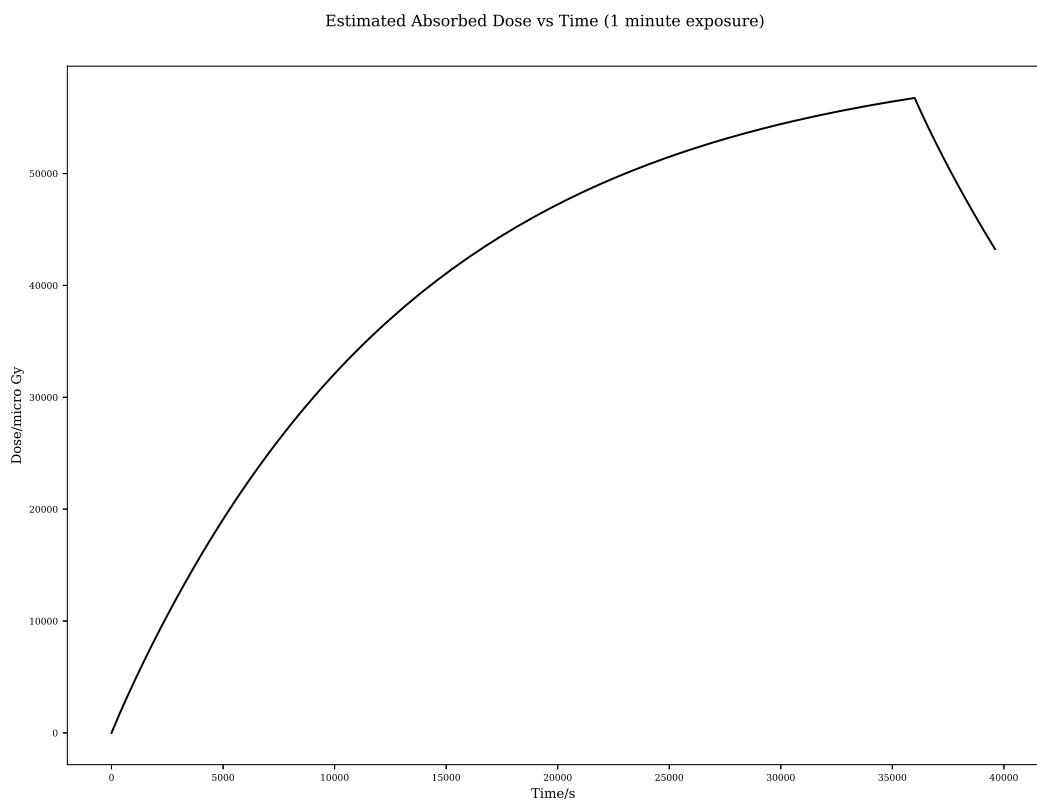


Figure 2.4: Steel Irradiation Gamma Dose $1.0e14$ Neutrons per cmsquared per second

2.2.3 Damage Types

Direct damage is caused when ionising radiation alters or destroys sections of DNA through a direct collision, for example a neutron colliding with and knocking out an atom in the DNA strand.

Our cells are predominantly water, so there is a higher probability of the radiation colliding with water molecules than DNA directly. The chemistry of the contents of the cell is changed, for example by the creation of free radicals, and these lead to damage of DNA as an indirect result of radiation passing through the cell.

2.2.4 Collision Event

The damage event in a material such as Steel takes place in stages, and this is similar to the way the event is handled for biological structures.

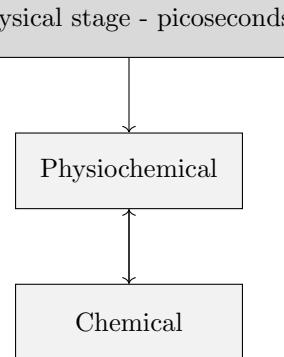


Figure 2.5

2.3 Damage Cascades in Metals

When a metal is damaged by radiation, there are microscopic defects created and removed in a very short space of time. The result of many months or years of damage is complex and will be discussed in more detail in the next chapter.

The cohesive energy of atoms are under 10eV, whereas the energy of incoming neutrons or fission fragments are measured in MeV.

Element	Cohesive Energy/eV
Aluminium	-3.36
Iron	-4.32
Palladium	-3.91
Platinum	-5.77
Zirconium	-6.32

Table 2.3: Cohesive Energies [22]

Radiation in the form of electrons, protons and heavy ions will lose energy to electrons in the target as they pass through. They may also lose energy to atoms within the lattice, knocking that atom out of place and creating a PKA. Due to it's neutral charge and magnetic moment, neutrons have a very weak interaction with charged particles. They primarily lose energy directly with the nucleus of atoms within the target, also creating a PKA.

The PKAs create a damage cascade, and their mean recoil energy is dependent upon the energy and type of radiation.

Projectile	Mean Recoil Energy
1 MeV Electrons	60eV
1 MeV protons	200eV
1 MeV heavy ions	5keV
1 MeV neutrons	35keV

Table 2.4: Average recoil energy of Nickel PKAs[23]

After each cascade, there will be a period of time where the material quenches, with a recombination of some interstitials and vacancies. The PKA creation and cascade thermal spike cover a time range of 10^{-18} s to 10^{-13} s,

with the quenching phase lasting in the region of 10^{-11} s. However, the cumulative effect of radiation damage on a component within a reactor core may only become apparent after months or years.

Chapter 3

Background: Austenitic Steels and Nuclear Power

Austenitic stainless steels have played an important role in the nuclear industry. They have a good resistance to corrosion and behave well at moderately high temperatures, having a good resistance to creep. They have been used within the primary circuit and secondary circuit of power stations, as cladding for fuel pellets, in the steam dryers, pre-heater tubing, core structurals, primary piping and more. Ferritic steel is magnetic and has a BCC structure, whereas Austenitic steel is non-magnetic with a FCC structure, with its high Nickel content stabilising this phase. An area of concern is the susceptibility of austenitic stainless steels to IGSCC.

3.1 Stainless Steel

3.1.1 Introduction

Stainless steel is a relatively new material, having first been developed and refined from the 1800s to the early 1900s, then being defined as a steel with at least 10.5% chromium in 1911[[stainlesssteelhistory](#)]. The addition of Chromium to this level causes the formation of a passive protective layer of Chromium Oxide.

Iron is alloyed with Chromium, Nickel and Carbon in varying quantities to make Stainless Steel. Depending on the application, other elements, such as Molybdenum, may be added to enhance the properties of the steel.

3.1.2 Grades of Stainless Steel

While the criteria that qualifies an alloy as Stainless Steel is containing Iron, Carbon and at least 10.5% Chromium, there are many grades that have variety of properties and atomic structures. The composition influences the structure, resistance to corrosion, yield strength, whether it is magnetic or not, and more.

Both Iron and Chromium are BCC at standard temperatures and pressures, but adding austenite stabilisers such as Nickel, Manganese or Carbon, steel can alter the structure from BCC to FCC. Balancing the proportion of these elements changes the phase of the steel, and this may be represented as a Schaeffler diagram or a ternary phase diagram.

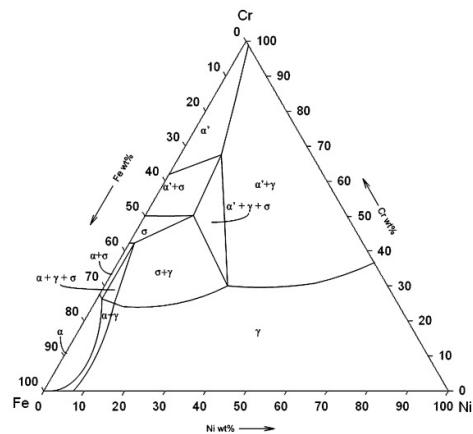


Figure 3.1: Iron-Chromium-Nickel Phase Diagram

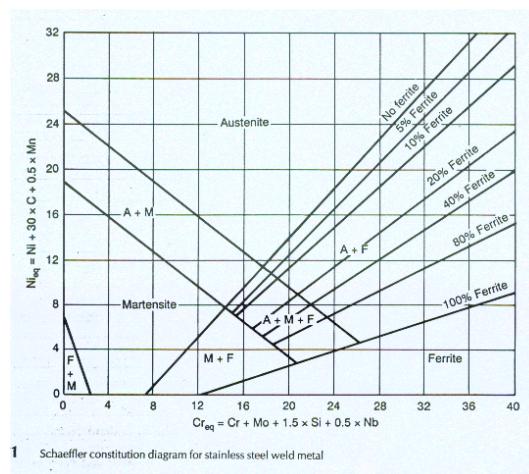
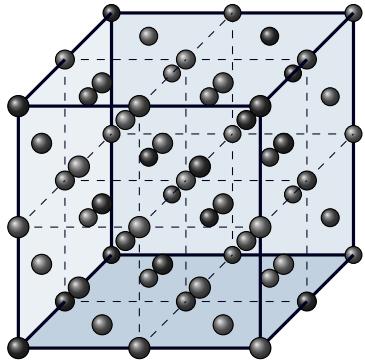


Figure 3.2: Steel Cr-Ni Schaeffler Diagram

Ferritic Stainless Steel

At room temperatures and pressures, Iron exists in the alpha phase, which is body centered cubic crystal. It is energetically favourable for the magnetic fields of the iron atoms to align

Ferritic stainless steels have the natural alpha phase crystal structure of pure Iron at room temperature which is bcc. These steels are magnetic and may be hardened by cold working. They are less corrosion resistant than austenitic stainless steels. Two common examples of this grade of steel are ASME (American Society of Mechanical Engineers) codes 405 and 430.

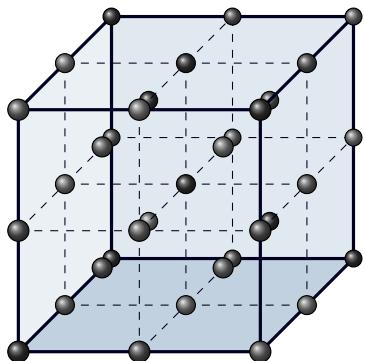


Austenitic Stainless Steel

Austenite is a FCC allotrope of Iron, and austenitic stainless steels are useful in many applications, including a structural material for nuclear plant components, due to their resistance to corrosion. In addition to 10.5 wt% or more chromium they require an austenite stabilising element such as carbon and/or nickel to be added.

Two examples of such steels are ASME codes 304 and 316. Both have a high Chromium content, in the region of 18-20%, which is in excess of the minimum passive film requirement of around 10-11%. The natural structure of such an Fe Cr alloy would be BCC, however 304 and 316 Steels contain Nickel (approximately 8% and 10% respectively) which is an Austenite stabiliser, and this give the FCC structure of the steel. The 316 grade contains a minimum of 2% Molybdenum to improve its resistance to corrosion.

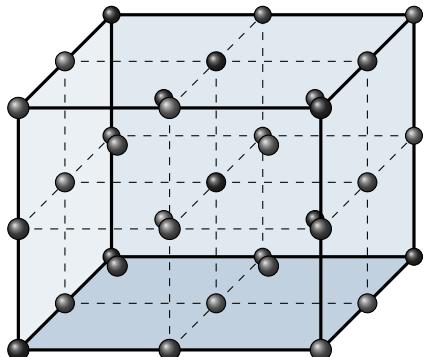
Ferritic and martensitic have a better resistance to thermal or swelling shock than austenitic [24], but the corrosion resistance of austenitic steels is a deciding factor when choosing a steel for an application where corrosion resistance is important.



Martensitic Stainless Steel

Martensitic stainless steels have a FCC structure at high temperature, but when heat treated take on a BCC structure. This allows Martensitic, unlike Austenitic and Ferritic, to be hardened by heat treatment. These steels are magnetic, they contain more than 10.5% Chromium but have a much lower Nickel than Austenitic

grades, if any. They are corrosion resistant, but the corrosion resistance of austenitic steel is typically better. Two examples of such steels are ASME codes 410 and 431.



Grade	Code	Hardness	Proof Strength	Tensile Strength (MPa)	Density (kgm^3)	Modulus of Elasticity
Austenitic	304	80	230	540-750	7900	200
Austenitic	316	79	240	530-680	8000	200
Ferritic	405	75	250	400-600	7700	220
Ferritic	430	85	280	450-600	7700	220
Duplex	255	32	550	750-1000	7800	200
Martensitic	410	90	205	600	7700	215
Martensitic	420	95	345	700	7700	215

3.2 Austenitic Steels in Nuclear Reactors

3.2.1 The Use of Austenitic Steels in Reactors

brief history, uses, problems etc

3.2.2 Atom Damage Cascades

Fission fragments carry away the majority of the energy released during fission, with an approximate energy of 170MeV per ^{235}U fission event. The fragments are large and have a highly charged positive nucleus that stops rapidly due to the Coulomb interaction with the nuclei of surrounding atoms. Electrons released by beta decay travel a greater distance; they lose energy by interacting with electrons, but may also knock atoms out of place. Neutrons travel further still and lose energy by interacting directly with the nuclei of the surrounding material.

When an incoming projectile hits an atom, the primary knock-on atom (PKA), this is knocked out of its place and loses energy by colliding with other atoms and to the electrons in the metal. Depending on the energy of the PKA, this causes a damage cascade. Atoms knocked out of their regular position in the crystal lattice leave vacancies behind. The displaced atoms either recombine with vacancies, become interstitial atoms or diffuse to defect sinks.

A number of complex problems arise as a result of this damage.

3.2.3 Damage Rates

The safety of a Nuclear reactor is the primary concern; it must generate power, but it must be safe. It must also be cost efficient. Generation II PWR fuel assemblies were expected to withstand several DPA[25]. In BWRs the materials are designed to be irradiated by $10^{22}n/cm^2/s$, experiencing approximately 7 DPA over the lifetime of

the components[26]. Components in PWRs are irradiated by up to $10^{22} n/cm^2/s$, and are expected to operate up to an irradiation dose of 70 dpa[26]. To remain cost efficient, components in Gen IV reactors will be expected to operate safely up to even higher doses or irradiation damage. Components in sodium-cooled fast reactors will be expected to withstand up to 200 DPA over their lifetime to meet the requirements of cost effectiveness and durability[25].

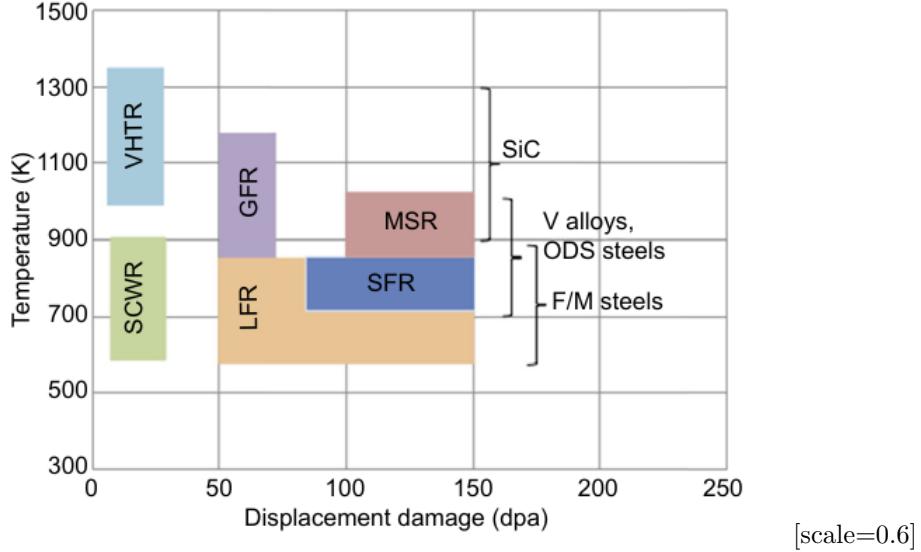


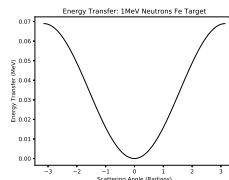
Figure 3.3: Expected DPA during component life time and operating temperatures[25]

Neutrons given out during ^{235}U fission range from 0 to 14 MeV, and the energy transferred to a target atom depends on the scattering angle of the neutron and the mass of the target nucleus.

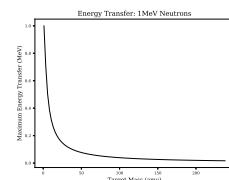
$$\alpha = \left(\frac{m_N - 1}{m_N + 1}\right)^2$$

$$E_N = E_n \left(1 - \frac{(1 + \alpha) + (1 - \alpha)\cos\theta_C}{2}\right) \quad (3.1)$$

The



(a) Scattering Angle



(b) Energy Transfer

Figure 3.4: Energy transfer and neutron scattering angle

3.2.4 Swelling

3.2.5 Radiation Induced Segregation

The diffusion of atoms within an alloy has an impact on the characteristics of the metal [27]. When radiation causes point defects, these interstitials also diffuse parallel to thermal solute diffusion. At low temperatures, the atoms are unable to diffuse at an appreciable rate; the mobility of vacancies are low[23] and there are an excess of vacancies due to radiation damage, and this leads to recombination of defects. At high temperatures, there is a higher concentration of thermal defects[26]. This increases the defect recombination rate and reduces migration of defects to sinks, such as grain boundaries.

The melting temperature range of 304 and 316 stainless steel are approximately 1695-1722K and 1644-1672K respectively. Generation II reactors, such as Sizewell B, have an operating temperature of several hundred degrees centigrade. The inlet temperature for the Sizewell B PWR reactor is 566K[28] and the outlet temperature is 597K[28]. Operating at approximately 35

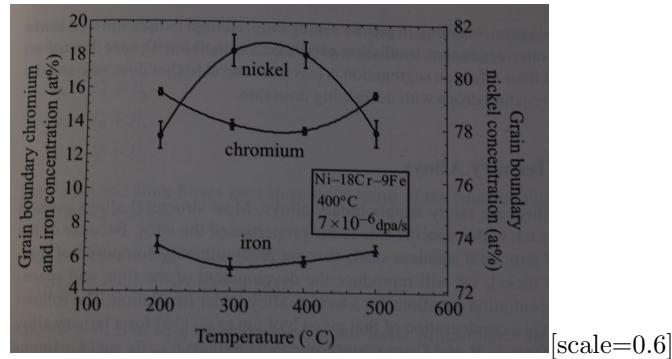


Figure 3.5: Grain boundary concentrations at 200°C to 500°C

The Kirkendall effect concerns the diffusion of one material into another and vice versa at an interface. In the original experiment performed by Kirkendall and Smigelskas, brass (Cu and Zn) was sandwiched between copper and left at a temperature of over 1050K for almost two months. Using Molybdenum as a marker, it was discovered that zinc diffused out of the brass faster than the copper diffused into the brass.

The inverse Kirkendall effect is driven by an external force, such as irradiation.

3.2.6 Irradiation Hardening

3.3 Corrosion Resistance of Austenitic Stainless Steels

3.3.1 Passive Film Protection of Chromium

The addition of Chromium improves the resistance of stainless steel to corrosion by orders of magnitude. A very thin passive oxide layer, 20-30 angstroms in width, forms when the surface is exposed to an environment containing oxygen. The protective layer is self repairing and, if the surface is damaged, the oxide layer reforms in the presence of oxygen.

When the steel is in an environment containing oxygen, electrons in the metal tunnel through the surface. As an oxidizing agent, the nearby oxygen atoms readily accept the electrons. A strong electric field forms between positive ions within the metal and the negatively charged oxygen atoms [29].

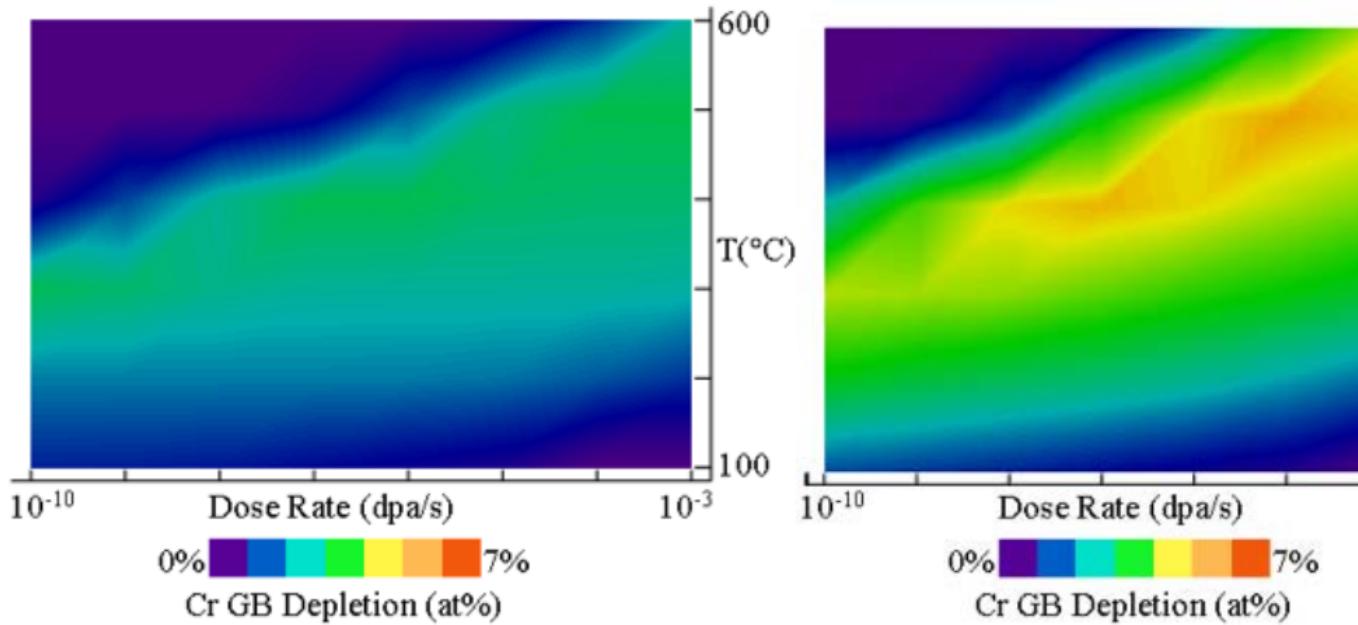


Figure 3.6: Chromium grain boundary depletion

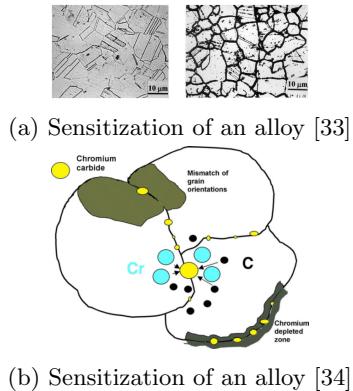


Figure 3.7: Energy transfer and neutron scattering angle

Iron at the surface of the forming oxide layer is preferentially dissolved away over chromium [30] and chromium ions within the oxide layer have a lower mobility than iron ions [31]. Nickel remains in place within the alloy, and this leads to an enrichment of Chromium oxide in the passive layer. Once the layer is thick enough to reduce the electric field across it, the formation of the layer stops. As mentioned previously, this is at a layer thickness of 2-3nm for stainless steel.

The Cr_2O_3 layer may be formed by heating the steel to 500°C [32].

3.3.2 Sensitization and Passive Film Removal

Steel by definition is iron alloyed with varying small percentages of carbon. Carbon changes the property of the alloy, and one example of this is an increase in hardness over pure iron. At elevated temperatures, 400°C to 750°C , the added chromium within the steel forms precipitates of iron-chromium carbides $(\text{FeCr})_{23}\text{C}_6$ at the grain boundaries, reducing the percentage of chromium at the grain boundary and removing the layer of passive protection.

3.3.3 Addition of Molybdenum

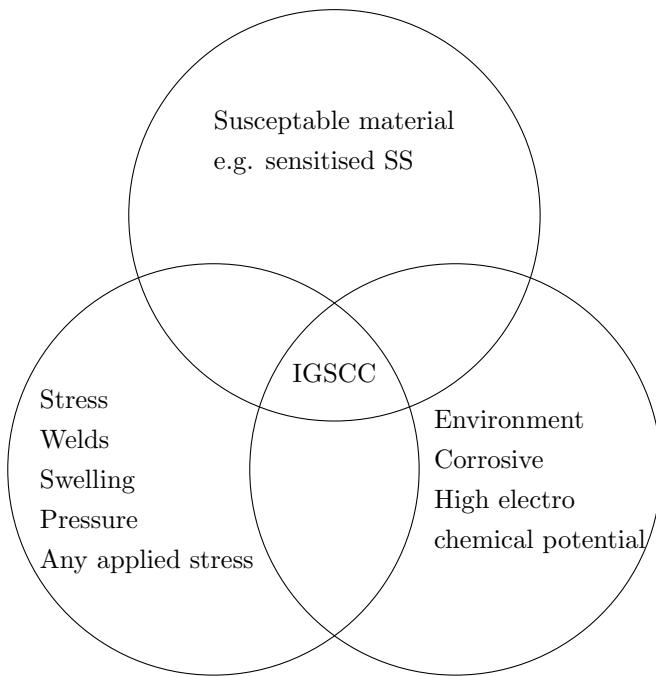
A common austenitic stainless steel is 304 and this relies heavily on the passive film due to a high content of Chromium, ranging from 17.5% to 20% for grades 304, 304L and 304H. A similar Stainless Steel, 316, has slightly less Chromium, slightly more Nickel and 2-3% Molybdenum, an element not present in 304 stainless steel.

Molybdenum is a ferrite former, but high proportions of austenite former such as Nickel and Nitrogen force 316 stainless steel to keep it's FCC structure despite the addition of Mo.

3.4 Inter Granular Stress Corrosion Cracking

Stress corrosion cracking may be trans granular (through the grains) or inter granular, at the grain boundary. IGSCC is a particularly prominent failure mechanism for Austenitic stainless steels. For IGSCC to occur, there must be a material that is susceptible to this form of cracking, an environment that is corrosive to the material as well as stress.

Stress may come from a high pressure environment, residual stress due to welding. In a reactor, the atomic structure may be under stress due to damage caused by neutrons passing through the steel, or due to the swelling of the steel on the macroscopic scale. If chromium is depleted at the grain boundary, protection to corrosion due to the passive layer is lost. This, coupled with the knowledge that austenitic stainless steels are susceptible to IGSCC, completes the three requirements and, over time, components of this material in these conditions will eventually fail to IGSCC.



IGSCC has not only been a defect in stainless steel. High Nickel content alloys, such as Alloy 600 (Inconel: Ni-72, Cr-17, Fe-10), have been known to suffer from IGSCC since the very early days of nuclear energy, in particular with the prototype S1W reactor, prototype for the first nuclear powered submarine, the USS Nautilus.

After the war, there was a drive to develop a nuclear powered navy for the US. There are obvious benefits in replacing conventional power in vessels with nuclear, in particular for submarines. This was pushed by Admiral Rickover and the route to building the USS Nautilus began.

A number of available Fe-Cr-Ni alloys were considered for the construction of the reactor 3.10.

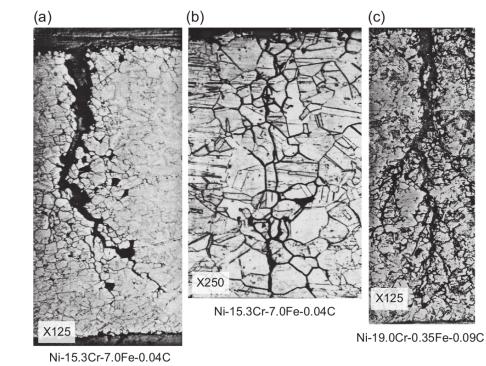


Figure 3.8: Inter Granular Stress Corrosion Cracking in Nickel Alloy[35]

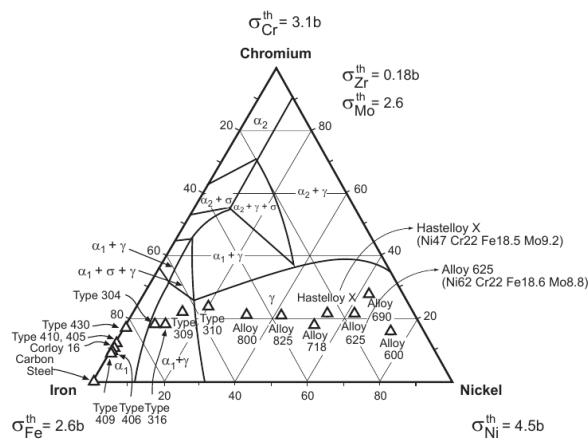


Figure 3.9: Alloy choices for early LWRs

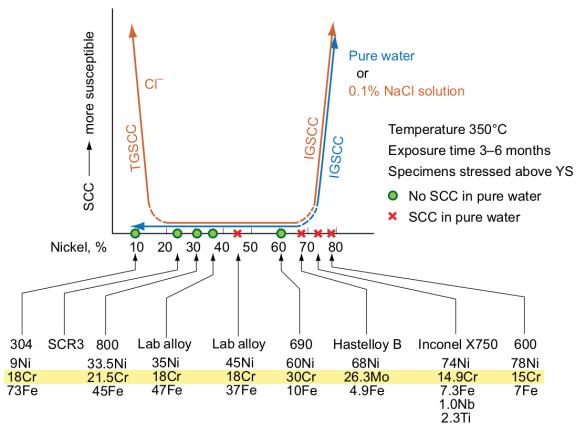


Figure 3.10: Nickel, IGSCC and TGSCC

Prototypes developed for the USS Nautilus experienced stress corrosion cracking in the Inconel Alloy 600 components. H. Coriou replicated this damage to Inconel in the laboratory, holding a sample at 350°C in deoxygenated water for 3 months[35]. Further studies showed that IGSCC became a particular issue to alloys with a high Nickel content in pure water, commencing with a content of approximately 70% Nickel. In a weak salt water solution (0.1% NaCl) IGSCC occurred as with the same alloy in pure water, but TGSCC occurred in similar alloys with a low Nickel content (less than 15%).

3.4.1 IGSCC in Light Water Reactors

In LWRs a major factor in corrosion of components is the water chemistry and Electro Chemical Potential that they operate in. Higher levels of oxygen dissolved in the water leads to an increase in the corrosion potential[36]. Other factors within LWRs include depletion of Chromium due to either heat treatment, welding or irradiation.

The primary circuit of a BWR includes components such as a reactor pressure vessel, piping that leads out of the containment building, fuel and fuel assembly. 300 series steels, including 304 and 316, are used within the primary circuit, including the piping and control rod absorbers. In BWRs the purity of the water has been addressed, and the addition of hydrogen to the water has reduced cracking of components[35].

Higher nickel content steels such as the 600 series alloys are used in PWRs, and components that are made from these alloys include steam generators and pressure vessels. In the AP-1000 design, the control rod drive mechanism at the reactor coolant pressure boundary are made from 304, 304L, 304LN, 316, 316L and 316LN steel. Higher Nickel content 690 is used for penetration into the pressure vessel[37]. As illustrated by Coriou's work in the 1950s and 1960s, higher Nickel content steels are susceptible to Stress Corrosion Cracking.

3.4.2 IGSCC and Advanced Gas-cooled Reactors

The first generation of nuclear reactors in the UK were Magnox type reactors. They used unenriched Uranium 0.7% U235 as fuel which was contained within magnesium oxide cladding due to its low absorption cross section. A major drawback was the relatively low operating temperature of 360°C.

Carnot's theorem shows that the maximum amount of energy from an engine is dependent on the difference between the hot and cold reservoirs. The ambient temperature of the powerstation will vary a small amount with the seasons. The practical way to improve the maximum possible efficiency is to increase the engine (reactor) temperature.

$$\eta_{max} = 1 - \frac{T_c}{T_h} \text{ where the temperature is measured in absolute units} \quad (3.2)$$

AGR's were designed to run at much higher temperatures of 650°C . By increasing the temperature, the maximum possible thermal efficiency was increased from just over 50 percent to almost 70 percent. To withstand higher temperatures, the magnesium oxide cladding used in the earlier Magnox reactors was replaced with stainless steel. To counteract the higher neutron absorption cross section of the cladding, the fuel was enriched up to 3.5% U²³⁵.

The cladding holds the fuel elements together and at the required location inside the reactor as it is being used up. The cladding also performs several functions once the fuel has been spent. It holds the fuel elements together and acts as a primary containment for the spent fuel[38].

During its lifetime as a fuel element within the reactor, the cladding has been heated and irradiated. By the process of RIS the chromium at the grain boundary can drop to 10% concentration[39]. The carbon rich environment within the reactor, due to the CO₂ coolant, provides yet another mechanism to deplete Chromium at the grain boundary by the formation of iron-chromium carbides (FeCr)₂₃C₆.

3.5 Austenitic Stainless Steels in Gen III+ and Gen IV Reactors

The AP1000 is an advanced PWR. As a Westinghouse reactor, the fuel cladding will be their trademarked Zirlo alloy. Inconel will be used for the steam generator and heat exchanger with carbon steel used for the construction of the pressure vessel. The control rod absorbers, however, will be constructed from 304SS[40].

Areva designed the EPR and this will use 316SS as the fuel cladding material. It will also use the slightly less corrosion resistant 304SS, forged, in parts of the control rod drive mechanisms[41].

The perforated upper core plate, between the reactor core and the inlet/outlet/control rod mechanisms, is also made from austenitic stainless steel. The pressuriser is constructed from ferritic steel, but its internals are clad with austenitic stainless steel to protect the structure from the coolant.

Generation IV designs also include the use of austenitic stainless steels. Experimental fast reactors in many countries, including the US, UK, Japan, France, India and China have used either 304SS, 316SS or both in their construction.

Austenitic steels are more corrosion resistant than ferritic or martensitic steels. They do expand more as a result of being irradiated, but they are stronger at higher temperatures. Many of the GenIV reactors operate at much higher temperatures than existing reactors and the components will be expected to resist higher doses of radiation damage through their lifetime.

GFRs are expected to use austenitic steels. Due to their chemical compatibility with sodium they are also expected to be used in SFRs. As well as being used in the reactor core, these steels will also be used outside the core. For example, SFRs in France will use austenitic steels in the secondary circuit, primary pump, internal heat exchanger and more[42].

Chapter 4

Background: Proton Activation and Radioactive Decay

Where proton or deuteron radiation is used to emulate neutron damage, there is still an associated risk of transmuting atoms in the target material, creating radioactive isotopes. An equation was derived to calculate the radioactivity of each element in the decay chain. It takes into account a source term for each isotope in the decay chain, as well as any branching factors.

Once derived, the equation was used in a computer code that, coupled with cross section data from the TENDL library and ion transport data from the SRIM ion transport code, predicts the radioactivity of a sample.

4.1 Ion Sources

4.1.1 Linac

Since the development of the first linacs (linear accelerators) in the 1940s, their modern day versions have become some of the most powerful accelerators in the world. The longest of linac, Stanford Linear Accelerator Center, is 3.2km in length and it accelerates electrons and positrons at energies of up to 50GeV. Several linacs for protons include the 800MeV linac component of the ISIS neutron source in Oxfordshire, and the 800MeV linac used by the Spallation Neutron Source at Oak Ridge National Laboratory.

The accelerator is constructed of several tubes, connected alternately to opposite terminals of a high frequency alternating current supply. As protons enter the first tube, a negative voltage is applied. As the protons reach the gap between the first and second tube, the polarity is reversed. The positive charge that is now applied to the first tube pushes the protons forward as the negative charge on the second tube pulls the protons forward. This process is repeated along the length of the accelerator, with the sections increasing in length due to the increase in velocity of the protons.

4.1.2 Cyclotron

Cyclotrons are reasonably compact and cost effective. The largest current cyclotron, TRI-University Meson Facility, is located in Canada and is able to output protons with energies over 500MeV. It is relatively large, weighing 170 tons. There are approximately 350 cyclotrons[43] around the world today.

The University of Birmingham cyclotron is more compact and the protons it accelerates are in the 8-40MeV range.

Projectile	Energy (MeV)	Maximum Current (micro A)
proton	8-40	60
deuteron	8-40	30
He^{2+}	8-53	30

Table 4.1: University of Birmingham Cyclotron Ion Beams

The moderate size, energy range, availability, cost of these Cyclotrons make them ideal candidates for damaging targets with light ions, rather than neutrons from a neutron source.

The Scanditronix MC40 is an isochronous cyclotron. As the ions are accelerated to higher and higher velocities, the effects of special relativity become appreciable.

$$v = c \sqrt{1 - \frac{E_{rest}^2}{E_V^2}} \quad (4.1)$$

The cyclotron has two Dees and these are D shaped hollow electrodes. Ions enter at the center of the cyclotron and are held in the Dees as they are accelerated. The magnetic field is varied to bend the path of the ions from a circle into a round cornered triangle.

4.1.3 Synchrotron

Two of the most well known accelerators are Synchrotrons: the Large Hadron Collider at CERN, and the now retired Tevatron at Fermilab. These are typically large machines and there are approximately 70 around the world[44], approximately a fifth the number of Cyclotrons.

Synchrotrons are used for storing high energy particles in a continuous loop, either to generate light through magnetobremssstrahlung, or for high energy collisions. Cyclotrons are more commonly used for lower energy (10s of MeV) projectiles, in materials science for damaging materials and for medical use creating radioactive isotopes.

4.2 Neutron Sources

4.2.1 High-Flux Neutron Reactors

There are 250 or so research reactors in 55 countries [45], and a number of these are used for materials research. In the UK, there is only one remaining research reactor, and this is the Neptune pool type reactor at Rolls Royce[46]. In Cadarache, France, the Jules Horowitz reactor is under construction and this is being built specifically as a materials testing reactor. It will be crucial in researching new materials for use in upcomming Gen IV nuclear power stations [45].

High Flux Isotope Reactor, Oak Ridge

The High Flux Isotope Reactor, at the Oak Ridge National Laboratory in America, is an 85MW research reactor that provides testing space within the reactor as well as a number of neutron beam lines. The reactor uses highly enriched Uranium as its fuel source and is scheduled to operate at 100% capacity for 161 days per year, in cycles of approximately 23 days. A 30cm surround of Beryllium is used to reflect neutrons, and in the reactor core there is a high flux of thermal neutrons at a rate of 2.3×10^{15} neutrons $cm^{-2}s^{-1}$ [47].

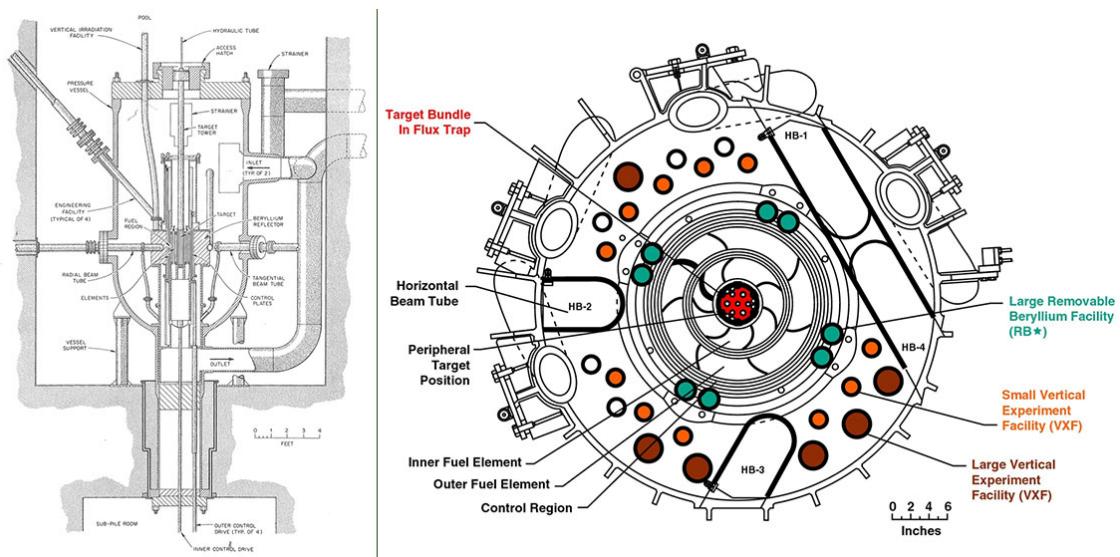


Figure 4.1: A cross section of the HFIR [48]

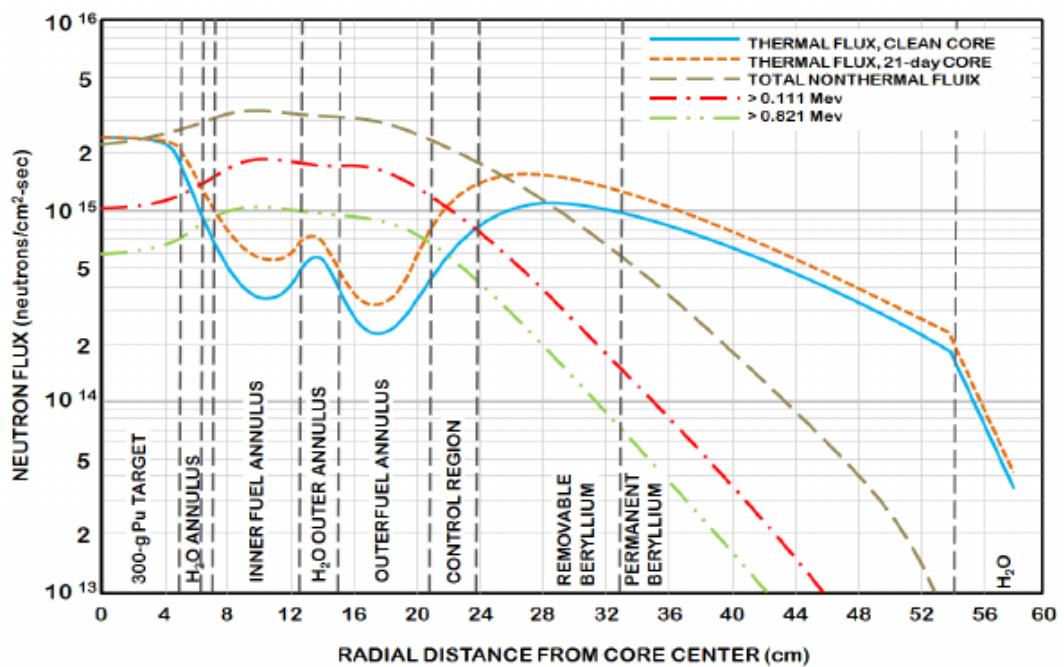


Figure 4.2: Neutron Flux in the HFIR while at 85MW [49]

NIST Center for Neutron Research

The NBS (National Bureau of Standards) Reactor was designed as a research reactor with a power output of 40MW and a neutron flux of approximately 1.0×10^{15} neutrons $cm^{-2}s^{-1}$. In 2006 it was decided to extend the facility and add beam lines

4.2.2 Spallation

Neutrons cause fission in nuclear reactors, but spallation sources require protons and high mass targets. The energy of the protons required is a magnitude greater than that of the Scanditronix MC-40 Cyclotron at the University of Birmingham, with spallation source accelerators having a range from 500MeV to over 1GeV.

ISIS is a neutron spallation source located in Harwell, Oxford.

Oak Ridge National Laboratory also has a neutron spallation source, the SNS. A LINAC accelerates a negatively charged proton (a hydrogen atom with two electrons) up to just below 0.9c. It is stripped of the electrons and enters an accumulation ring as a proton. More protons are accumulated until the beam in the accululator is diverted at a mercury target. The collision between the mercury target and the protons releases approximately 20 neutrons per collision, and these neutrons pass through beam lines to the experiments.

The source creates a pulse of neutrons from the 50 ton mercury target 60 times per second. The protons are accelerated by the LINAC to between 2.5MeV and 1.0GeV and this results in neutrons with energies almost as high as the proton projectiles. A spectrum of neutron energies is produced with each pulse, although the neutrons may be moderated at the end of the beam lines to slow the neutrons down.

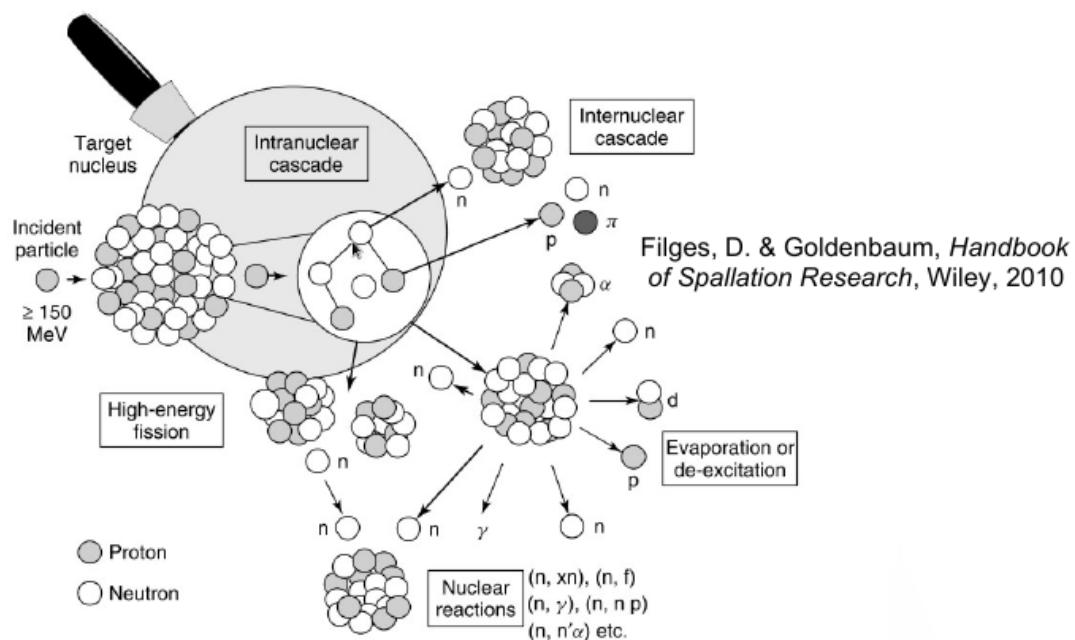


Figure 4.3: Neutron spallation source [50]

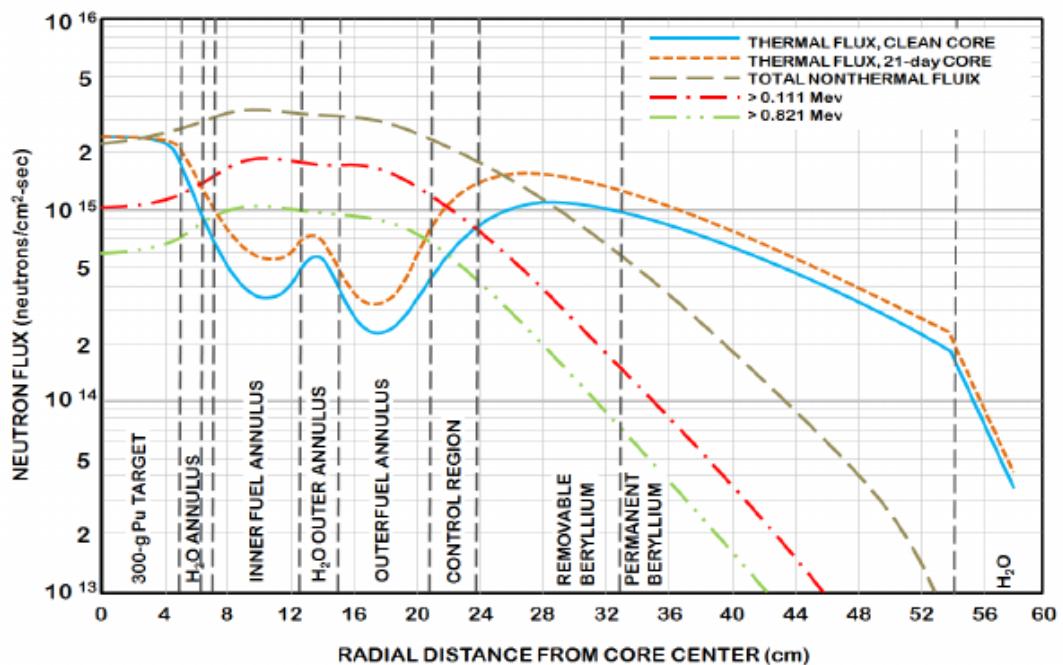


Figure 4.4: Neutron energy spectrum [50]

4.3 Source Review

Source	Cost	Projectile	Flux/Current	Energy
Scanditronix MC-40	1 million	Proton	3.7×10^{14}	8-40MeV
HFIR[49]		Neutron	3.0×10^{15}	Full Spectrum
HFIR[49]		Thermal Neutron	Over 2.0×10^{15}	Thermal
HFIR[49]		Fast Neutron	2.0×10^{15}	$\sim 0.111\text{MeV}$
HFIR[49]		Fast Neutron	1.0×10^{15}	$\sim 0.821\text{MeV}$
SNS[51]	\$1.4 billion	Neutron	Average 1.2×10^{13}	Full Spectrum
ISIS spallation source[51]		Neutron	Average 4.0×10^{13}	Full Spectrum

Table 4.2: Neutron sources

The compact proton source that is the cyclotron is compared to neutron sources. It is much smaller and cheaper in comparison and the energy of the projectile may be controlled, whereas a spectrum of energies of neutrons is produced by reactors and spallation sources. Faster neutrons will cause more damage to materials in GenIV reactors giving the cyclotron an advantage, being able to focus in just one energy range. The three mentioned isotope sources are national facilities that cost much more than a cyclotron. Access is shared between many researchers whereas a cyclotron, such as the Scanditronix MC-40 at the University of Birmingham may have a beam dedicated to a particular task.

4.4 Ion Irradiation to Investigate Neutron Damage

Neutrons emitted during the fission of Uranium-235 have an energy spectra in the Intermediate to Fast range, with a peak at 1MeV, but a sizeable amount in the 1MeV to 10MeV range^{4.5}. The higher energy neutrons are more of a concern to this work as higher energy neutrons, on colliding with atoms within the target material, cause large damage cascades.

As discussed earlier in this chapter, there are a number of methods available to create both high energy neutrons and ions, but each has its own set of advantages and disadvantages. At the University of Birmingham high energy ions are created using a cyclotron.

Figure 4.5: Neutron Spectra from the Fission of U235[52]

4.4.1 Ion Irradiation at the University of Birmingham

The Scanditronix MC-40 Cyclotron is used at the University of Birmingham to create a beam of protons or other light ions. The energies of these ions are typically between 10 MeV and 60 MeV with beam currents ranging up to 50 microamps (3.1×10^{14} protons per second). Target materials are irradiated by this cyclotron for a number of reasons, including purposely creating radioactive isotopes for the nearby Queen Elizabeth Hospital, investigating ion irradiation damage and emulating neutron irradiation.

The Cyclotron is usually used to create radioactive isotopes for medical use, but an additional beam line has been devoted to material science investigations into radiation damage. While the creation of radioactive isotopes is desired in some cases, material being tested for radiation damage should preferably have low levels of radioactivity.

It is expensive to arrange the irradiation of target materials by high energy neutrons sources, whereas it is relatively inexpensive to irradiate using an ion beam on the MC-40 Cyclotron. The energies can be controlled, and a set dose at a single energy, or a range of energies, can be precisely deposited into the target material.

The Activity code discussed here was developed to calculate the activity of a target material irradiated by a proton beam. It has been developed in Fortran and uses data from the TENDL-2013 proton cross section database, SRIM ion transport code and Nuclear Data Services radioactive decay database.

4.4.2 Transmutation of Nuclei by Neutrons and Protons

Neutron Activation

The fission of Uranium-235 atoms results in neutrons with a varied spectrum of energies. The neutrons will bounce around inside the reactor losing energy quickly to light atoms within moderators and coolants, such as water. At lower energies the neutrons may be captured by the nuclei of atoms they interact with. This creates a new isotope which may or may not be stable.

Proton Activation

Considering a simplified nuclear potential well, energetic protons approaching a nucleus may overcome the Coulomb potential barrier. They are captured by the nucleus and held within the potential well by the strong nuclear force. This process may leave the nucleus in an excited and unstable state, depending on the input energy of the proton and configuration of nucleons. The process is probabilistic, and the average chance of a reaction (the microscopic cross section) may be measured as a function of the projectile, projectile energy and target, either experimentally or by optical model potential calculations. The reaction rate is calculated from the microscopic cross section using the following equation:

$$R = \frac{J}{Q} \cdot n_p \cdot \sigma \cdot 10^{-28} \delta t \quad (4.2)$$

- R Reaction Rate (reactions per second)
- J Beam current (A)
- n_p Number density of target (atoms per cubic metre)

Figure 4.6: Cross section: Fe56 to Ni58

Figure 4.7: TALYS work flow [54]

- σ microscopic reaction cross section (barns)
- Q projectile charge e.g $1.602177 \times 10^{-19} C$ for a proton
- δt target thickness (m)

4.4.3 Nuclear Reaction Cross Sections

Reaction Cross Sections

The type of reaction for an individual reaction cannot be determined, but the probability of that reaction happening may be measured and future events predicted. This data may be gathered experimentally, or it may be calculated using various models.

TALYS and the Optical Model Potential

In the standard model, protons and neutrons are composed of quarks, held together by the strong nuclear force. The nucleus of an atom is also held together by the strong nuclear force that on such small separations overwhelms the electromagnetic force of the protons with one another.

This is a complicated system, not to mention the excited states that nuclei may occupy. The interaction of a projectile (proton, neutron or a heavier ion) with the nucleus is a challenge to model.

Initially, reaction data for protons and neutrons was gathered through experimentation. In the 1950s, Feshbach et al put forward a simple model for nuclear reactions between the nucleus and neutrons, and this model was restricted to 0-20MeV neutrons. The form of the potential used was a complex function[53].

$$\begin{aligned} V &= V_0(1 + i\zeta)r < R \\ V &= 0 \quad r > R \end{aligned} \tag{4.3}$$

The potential in equation [eq:FeshbachPotential] has the parameters $R = 1.4 \times A^{\frac{1}{3}}$, $V_0 = 42\text{MeV}$ and $\zeta = 0.03$ where A is the atomic mass of the target atom.

Considering the complexity of the system being modelled, this simple model was very successful. Over the years since, the models used have become more complex and parameters used have been fit to an increasing amount of experimental data.

The Talys code uses a range of models (figure 4.7). These include the optical model which is solved using the Equations Couples Iterations Squentielle (ECIS-06) code of Jacques Raynal, implemented as a subroutine within Talys, and this is accurate up to 180MeV[54]. Whilst the Talys code has been extended up to 1GeV, it is experimental. Fortunately, this work only requires nuclear reaction cross section data up to approximately 100MeV as the current ion source under consideration produced ions with energies up to 60MeV.

The Talys code has potentials for protons and neutrons, but it also has potentials for deuterons, tritons, helium-3 and alpha particles. The potentials are discussed in detail in the Talys manual[54]. This extension to heavier ions may be useful for this work as the University of Birmingham cyclotron is capable of accelerating deuterons and He^{2+} ions.

Figure 4.8: TALYS Fe54-Co55 cross section comparison with experiment [55]

Comparing experimental data to the Talys data for $Fe54(p, \gamma)Co55$ shows good agreement between 1MeV and 10MeV. There was insufficient experimental data from the this source, but the Talys generated data ranges from micro eV up to 100MeV+.

TENDL 2019 Data Files

The cross section data for protons and neutrons is available to download in ENDF format files. The data files used by this work are TENDL data files, and these are created by a combination of different nuclear models and experimental data. The TENDL nuclear reaction simulation code provides the calculated data.

The nuclear reaction files are rather large, and they all follow a standard format.

Listing 4.1: Sample TENDL File

1	2.605600+4	5.545443+1	0	0	0	02631	3	2	1
2	0.000000+0	0.000000+0	0	0	1	462631	3	2	2
3	46	2				2631	3	2	3
4	1.000000+3-9.920042-7	1.000000+6-9.920042-7	2.000000+6-2.167241-42631	3	2	4			
5	3.000000+6-7.609736-3	4.000000+6-3.232345-2	5.000000+6-5.975554-22631	3	2	5			
6	6.000000+6-5.149510-2	7.000000+6-3.685484-2	8.000000+6-5.542173-22631	3	2	6			
7	9.000000+6-9.978765-2	1.000000+7-1.599116-1	1.100000+7-2.206871-12631	3	2	7			
8	1.200000+7-2.743084-1	1.300000+7-3.165007-1	1.400000+7-3.471231-12631	3	2	8			
9	1.500000+7-3.674350-1	1.600000+7-3.788282-1	1.700000+7-3.825634-12631	3	2	9			
10	1.800000+7-3.799978-1	1.900000+7-3.725944-1	2.000000+7-3.617662-12631	3	2	10			
11	2.200000+7-3.342054-1	2.400000+7-3.031296-1	2.600000+7-2.709398-12631	3	2	11			
12	2.800000+7-2.388019-1	3.000000+7-2.077513-1	3.500000+7-1.398614-12631	3	2	12			
13	4.000000+7-8.801994-2	4.500000+7-5.013441-2	5.000000+7-2.349373-22631	3	2	13			
14	5.500000+7-5.480233-3	6.000000+7 6.189980-3	6.500000+7 1.332313-22631	3	2	14			
15	7.000000+7 1.727774-2	7.500000+7 1.906833-2	8.000000+7 1.943711-22631	3	2	15			
16	9.000000+7 1.783081-2	1.000000+8 1.499871-2	1.100000+8 1.211197-22631	3	2	16			
17	1.200000+8 9.641120-3	1.300000+8 7.717424-3	1.400000+8 6.318436-32631	3	2	17			
18	1.500000+8 5.367555-3	1.600000+8 4.771939-3	1.800000+8 4.312985-32631	3	2	18			
19	2.000000+8 4.445323-3			2631	3	2	19		

4.4.4 Radioactive Decay

Radioactive decay is the random change in nucleons or energy state of an unstable nucleus. It is impossible to predict when a single nucleus will decay, but the decay of a collection of nuclei is statistical in nature. The radioactivity and number of unstable nuclei at time t can be predicted using the decay constant, λ , for the radioactive isotope. This constant is defined as follows:

$$\lambda = -\frac{N'(t)}{N(t)} \quad (4.4)$$

The number of radioactive nuclei $N(t)$ at time t is given by the following equation, where $N(0)$ is the starting number of nuclei:

$$N(t) = N(0) \exp(-t\lambda) \quad (4.5)$$

The activity $A(t)$ of the radioactive nuclei is predicted at time t by using the following equations, where $N'(t)$ is the change in amount of nuclei with respect to time:

$$A(t) = -N'(t) = \lambda N(t) \quad (4.6)$$

$$A(t) = \lambda N(0) \exp(-t\lambda) \quad (4.7)$$

4.4.5 Saturation Activity

As a radioactive isotope is created by reactions between target atoms and projectiles (protons, neutrons, deuterons) it will begin to decay. The amount of the isotope will continue to increase until the decay rate equals the reaction rate for the creation of the isotope.

For a single isotope:

$$\frac{dN}{dt} = \frac{J}{Q} \cdot n_\rho \cdot \sigma \cdot 10^{-28} \delta t - \lambda N \quad (4.8)$$

$$N(t) = \frac{\frac{J}{Q} \cdot n_\rho \cdot \sigma \cdot 10^{-28} \delta t}{\lambda} (1 - \exp(-\lambda t)) \quad (4.9)$$

4.4.6 Decay Constants: Joint Evaluated Fission and Fusion File (JEFF) 3.3 Data File

JEFF 3.3 is an evaluated data file[56], meaning it has been evaluated by a relevant authority. The quality of the data may affect the health of the public and industry workers, so it must be evaluated. This particular file is managed by and available through the Nuclear Energy Agency (NEA).

It is a collection of many data files. Released in 2017, it also contains several files for incident gammas, protons, deuterons, tritons, helium-3 and alphas from the TENDL 2017 data file.

The files that will be important for this work are the ENDF-6 radioactive decay data files only. The nuclear reaction cross section data will come from Talys and various iterations of the TENDL libraries. The decay data held in the JEFF 3.3 file includes isotope data, masses, half lives, branching factors, continuous and discrete gamma data.

4.4.7 Bateman Equation for Radioactive Decay

The English mathematician Harry Bateman derived an equation (4.10) to calculate the amount of each isotope in a decay chain, illustrated in Figure 4.9, at time t.

$$N_n(t) = \sum_{i=1}^{i=n} \left(\left(\prod_{j=i}^{j=n-1} \lambda_{(ij+1)} \right) \sum_{j=i}^{j=n} \left(\frac{N_{i0} \exp(-\lambda_j t)}{\prod_{p=i, p \neq j}^{p=n} (\lambda_p - \lambda_j)} \right) \right) \quad (4.10)$$

When a radioactive isotope decays, there may be more than one mode of decay, and this leads to branching factors. Pb-214 only decays via beta decay to Bi-214, giving a branching factor of 1.0, whereas Bi-214 has a 99.979% chance of decaying to Po-214 by beta decay and a 0.021% of emitting an alpha particle and decaying to Tl-210 (branching factors of 0.99979 and 0.00021 respectively) [56].

When a target material is irradiated, there is a source term for transmuted nuclei due to the irradiation. The daughter isotopes of these transmuted isotopes will also be affected by the irradiation and will transmute further, giving a source term for each daughter isotope as a result of the irradiation. Sources for each isotope in the decay chain, and branching factors between a parent isotope and its daughter isotope/s must be accounted for.

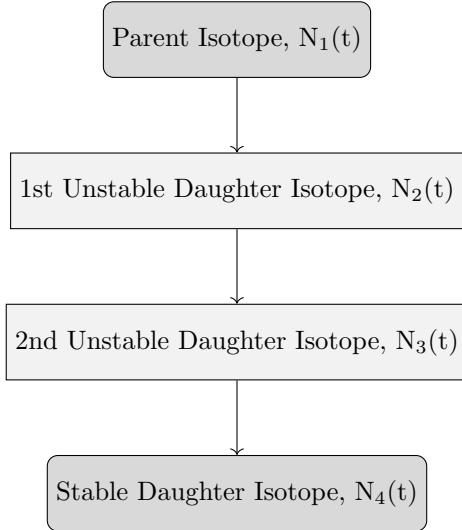


Figure 4.9: An example decay chain from an unstable parent isotope, through unstable daughter isotopes ending with a stable daughter isotope.

4.4.8 Cross Section Dependence on Energy and Target Thickness

4.5 Simulating Ion Irradiation with SRIM

4.5.1 What SRIM Does

The SRIM computer code

4.5.2 What SRIM Does Not Do

A package of ion transport codes, SRIM, is freely available to download and use to investigate the transport of ions through matter. SRIM uses the Binary Collision Approximation to simulate the passage of ions in a material. It is an approximate method, and one key restriction is that it does not take into account the structure of the material, and this approximation is therefore also imposed on the Activity code.

One file that SRIM creates is of importance to the Activity code, and that is the trajectory file that contains the energy and x,y,z co-ordinate data points for simulated ions moving through matter. Figure 4.10 shows the trajectory of one hundred 13MeV protons entering and passing through an Iron target, and it is this set of data points (together with the cross section database) that the Activity code uses to calculate the reaction rates for the transmutation of nuclei in the target. At higher energies, the ions slow as they lose energy due to electronic stopping, but as the ion energy drops the mechanism of loss through nuclear collisions becomes important. The spreading of ion depths at lower energies is a result of the higher momentum transfer during nuclear collisions, as can be seen in Figure 4.10.

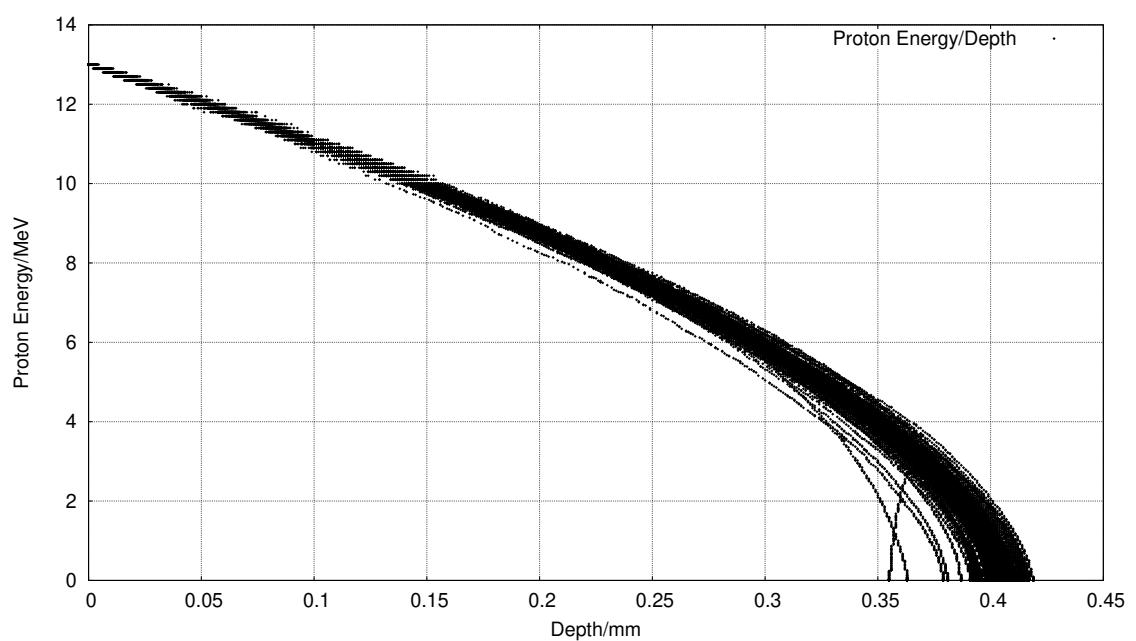


Figure 4.10: One hundred simulated 13MeV proton energy loss curves in Fe simulated with SRIM [57]

Chapter 5

Background: Interatomic Potential Fitting

In order to model Iron and Palladium with Molecular Dynamics codes, an interatomic potential is needed. This will require experimental data and data generated by first principles calculations. The simplified model will consist of just Iron and Palladium, but pure Iron does not take the FCC structure that it does when alloyed with Nickel in Austenitic Stainless Steel. The properties of theoretical pure FCC Iron are estimated using Density Functional Theory that solves the many body Schrodinger Equation. While it is impossible to solve this equation exactly, with our current technology and knowledge, it is possible to solve approximately by making several approximations and by using the Hohenberg-Kohn theorem.

5.1 Experiment, Modelling and Theory

5.1.1 Introduction

Experiment: direct answers from physical reality Limited by technology of the time, also limits due to Heisenberg uncertainty principle Theory: state of the art theories have been replaced numerous times in the past Quantum mechanics is a very accurate theory under certain circumstances Some problems just too hard to solve with these theories Modelling: bridges the gap Experiment and theory have their flaws, so does modelling Helps show what experiment cant, and where theory is too hard to solve Aim: take very accurate DFT calculations based on quantum theory, extrapolate to a larger scale by fitting EAM potentials, open a path to simulations of Pd + Fe and Ru + Fe

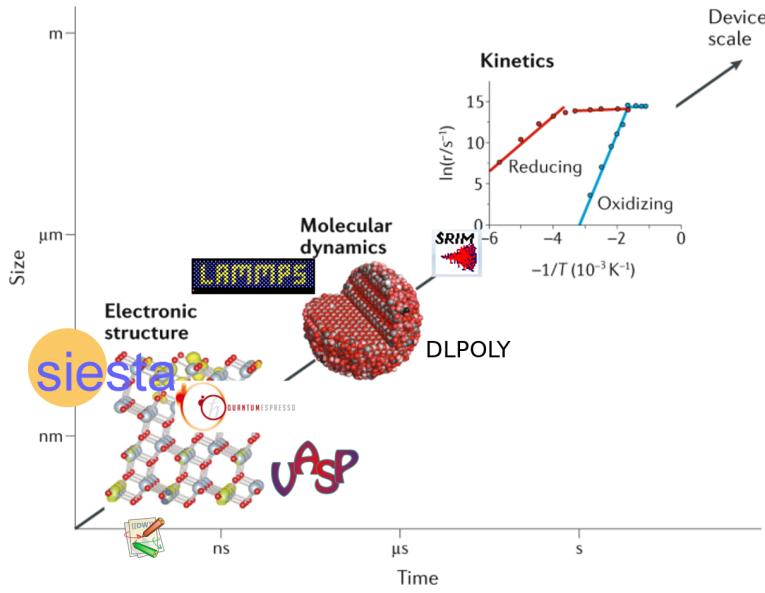


Figure 5.1: Time and Size Scales for Computer Packages [58]

5.1.2 Simulating Materials on a Variety of Scales in Time and Space

Stainless steel grain size less than 100 microns. A 1 micron grain would contain tens of billions of atoms. Given this number of atoms for a very small grain, it is very difficult to simulate just one grain over a short time period.

Certain properties derived ab initio assume the entire material is a single crystal, rather than made from grains of crystals.

5.2 Properties of Crystals

5.2.1 Introduction

There are seven crystal classes, although this work is only concerned with cubic and orthorhombic crystals.

Table 5.1: Seven Crystal Classes

Class	Lengths	Angles
Cubic	$a = b = c$	$\alpha = \beta = \gamma = 90^\circ$
Hexagonal	$a = b, c$	$\alpha = \beta, \gamma = 120^\circ$
Rhombohedral	$a = b = c$	$\alpha = \beta = \gamma \neq 90^\circ$
Tetragonal	$a = b, c$	$\alpha = \beta = \gamma = 90^\circ$
Orthorhombic	a, b, c	$\alpha = \beta = \gamma = 90^\circ$
Monoclinic	a, b, c	$\alpha = \beta = 90^\circ, \gamma \neq 90^\circ$
Triclinic	a, b, c	$\alpha, \beta, \gamma,$

The force matching method uses DFT calculated forces as well as other properties, calculated either by DFT or measured by experiment, in order to fit potential functions. These additional constraints include:

- Lattice Parameter a_0
- Cohesive Energy E_{coh}

- Bulk Modulus B_0
- Equation of State (E_0, V_0, B_0, B'_0)
- Stress Tensor
- Elastic Constants
- Shear Modulus, Young's Modulus, Poisson Ratio
- Melting Temperature
- Surface Energy

5.2.2 Bulk Modulus

The Bulk Modulus of a material is defined as the bulk stress of a sample divided by the bulk strain on that sample. It is also the inverse of the compressibility of that material, which means that materials with a higher bulk modulus are less compressible than those with a lower value.

$$B_0 = -V \frac{\partial P}{\partial V} \quad (5.1)$$

$$B_0 = V \frac{\partial^2 E}{\partial V^2} \quad (5.2)$$

Table 5.2: Useful Conversion Factors

Material	Bulk Modulus/GPa
Aluminium	70
Iron (BCC)	110
Stainless Steel 18-8	163

5.2.3 Equation of State

The equation of state of a material relates either the pressure on that material as a function of the volume, or the energy of a sample of a material to the volume. This not only allows one to predict the energy or pressure at a certain volume, but also the minimum energy, relaxed volume and the bulk modulus.

Murnaghan Equation of State

Hooke's law implies a linear relationship between stress and strain. In practice, where a pressure is applied to a material, the application of Hooke's law is limited[59]. Muraghan derived a new equation to improve upon formulae developed in the 1930's, using compression data from high pressure experiments.

$$P(V) = \frac{B_0}{B'_0} \left(\left(\frac{V_0}{V} \right)^{B'_0} - 1 \right) \quad (5.3)$$

As pressure is the negative derivative of the internal energy of the system with respect to change in volume, $p = -(\partial E/dV)$, and the equation can be integrated and written in terms of the energy, volume, bulk modulus and its derivative[60].

$$E(V) = E_0 + \frac{B_0 V}{B'_0} \left[\left(\frac{V_0}{V} \right)^{B'_0} \frac{1}{B'_0 - 1} + 1 \right] - \frac{B_0 V_0}{B'_0 - 1} \quad (5.4)$$

Birch-Murnaghan Equation of State

Several years after Murnaghan's equation, Birch developed the equation of state further upon the experimental data provided by work from Bridgman in high pressure physics. For cubic symmetry, the description of free energy now includes third order terms in the strain components[61].

$$P(V) = \frac{3B_0}{2} \left[\left(\frac{V_0}{V} \right)^{\frac{7}{3}} - \left(\frac{V_0}{V} \right)^{\frac{5}{3}} \right] \left[1 + \frac{3}{4}(B'_0 - 4) \left(\left(\frac{V_0}{V} \right)^{\frac{2}{3}} - 1 \right) \right] \quad (5.5)$$

The energy-volume relationship may again be constructed[60].

$$E(V) = E_0 + \frac{9V_0 B_0}{16} \left[\left[\left(\frac{V_0}{V} \right)^{\frac{2}{3}} - 1 \right]^3 B'_0 + \left[\left(\frac{V_0}{V} \right)^{\frac{2}{3}} - 1 \right]^2 \left[6 - 4 \left(\frac{V_0}{V} \right)^{\frac{2}{3}} \right] \right] \quad (5.6)$$

In order to fit the Birch-Murnaghan, the first step is to fit a second order polynomial to the energy-volume data. This may be achieved using least-squares fitting with a vandermode matrix. The coefficients from this polynomial may then be used to calculate reasonable value for E_0 , V_0 and B_0 ; sane starting values of B'_0 are between 1 and 10, and the code takes a starting value of 2[62].

$$\begin{aligned} E(V) &= c_0 + c_1 V + c_2 V^2 \\ V_0 &= -\frac{c_1}{2c_2} \\ E_0 &= c_2 * V_0^2 + c_1 V_0 + c_0 \\ B_0 &= 2c_2 V_0 \\ B'_0 &= 2 \end{aligned} \quad (5.7)$$

Newton Gauss is then used to minimise E_0 , V_0 and B_0 while $B'_0 \in 1.0, 1.5, 2.0, 2.5, 3.0, 3.5, 4.0, 4.5, 5.0, 5.5, 6.0$

$$[J^T J] P = J^T R \quad (5.8)$$

The parameters with the lowest residual square sum are returned.

5.2.4 Voigt Notation

Where a tensor is symmetric, Voigt notation is used to simplify how the tensor is written. It reduces a second order tensor such as stress or strain, represented by a 3x3 matrix, to a 6 row matrix. It also reduces a fourth

order tensor, such as the compliance or stiffness tensor (represented by a 3x3x3x3 matrix), to a 6x6 matrix.

$$\vec{A} = \begin{bmatrix} A_{11} & A_{12} & A_{13} \\ A_{21} & A_{22} & A_{23} \\ A_{31} & A_{32} & A_{33} \end{bmatrix} = \begin{bmatrix} A_{11} \\ A_{22} \\ A_{33} \\ A_{23} \\ A_{13} \\ A_{12} \end{bmatrix} \text{ if } \vec{A} \text{ is symmetric} \quad (5.9)$$

5.2.5 Elastic Constants

The Generalized Hooke's law relates the second order stress and strain tensors using a fourth order stiffness tensor.

Stress definition . Stress is measured in Pa, although in this work it may also be measured in either $Ry/Bohr^3$ or eV/Ang^3 .

Strain is a measure of the deformation of a material and by definition has no units. In one dimension $\epsilon = \frac{\delta l}{l_0}$ where l_0 is the unstrained length and δl is the change in length.

In three dimensions, a symmetric tensor

$$\vec{A} = \begin{bmatrix} \epsilon_{11} & \epsilon_{12} & \epsilon_{13} \\ \epsilon_{21} & \epsilon_{22} & \epsilon_{23} \\ \epsilon_{31} & \epsilon_{32} & \epsilon_{33} \end{bmatrix} = \begin{bmatrix} \epsilon_{11} \\ \epsilon_{22} \\ \epsilon_{33} \\ \epsilon_{23} \\ \epsilon_{13} \\ \epsilon_{12} \end{bmatrix} \text{ where } \epsilon_{ij} \equiv \frac{1}{2} \left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} \right) \quad (5.10)$$

$$\sigma_{ij}^{\vec{j}} = C_{ijkl}^{\vec{j}} \epsilon_{kl}^{\vec{j}} \quad (5.11)$$

$$\epsilon_{ij} = \begin{bmatrix} \epsilon_1 = \epsilon_{11} = \epsilon_{11} \\ \epsilon_2 = \epsilon_{22} = \epsilon_{22} \\ \epsilon_3 = \epsilon_{33} = \epsilon_{33} \\ \epsilon_4 = \epsilon_{23} = \epsilon_{32} \\ \epsilon_5 = \epsilon_{13} = \epsilon_{31} \\ \epsilon_6 = \epsilon_{12} = \epsilon_{21} \end{bmatrix} \quad (5.12)$$

$$\begin{aligned}
& \begin{bmatrix} \sigma_1 \\ \sigma_2 \\ \sigma_3 \\ \sigma_4 \\ \sigma_5 \\ \sigma_6 \end{bmatrix} = \\
& \begin{bmatrix} C_{11} & C_{12} & C_{13} & C_{14} & C_{15} & C_{16} \\ C_{21} & C_{22} & C_{23} & C_{24} & C_{25} & C_{26} \\ C_{31} & C_{32} & C_{33} & C_{34} & C_{35} & C_{36} \\ C_{41} & C_{42} & C_{43} & C_{44} & C_{45} & C_{46} \\ C_{51} & C_{52} & C_{53} & C_{54} & C_{55} & C_{56} \\ C_{61} & C_{62} & C_{63} & C_{64} & C_{65} & C_{66} \end{bmatrix} \\
& \begin{bmatrix} \epsilon_1 \\ \epsilon_2 \\ \epsilon_3 \\ \epsilon_4 \\ \epsilon_5 \\ \epsilon_6 \end{bmatrix} \tag{5.13}
\end{aligned}$$

5.2.6 Calculating Elastic Constants for a Cubic Crystal

Cubic crystals are the most simple class with primitive unit cells having three orthonormal basis vectors. Four of the most common variants of the cubic crystal are the simple cubic, body centered cubic, face centered cubic and zincblende. Due to symmetry, a cubic crystal has only three elastic independent constants; in Voigt notation these are C_{11} , C_{12} and C_{44} .

Applying two strains to a cubic crystal [63] coupled with the calculation of the Bulk Modulus, as already discussed, allows the three independent elastic constants to be calculated.

First, the bulk modulus may be calculated either by finding the second derivative of the energy wrt volume at the relaxed volume, $B(V) = -VP'(V) = VE''(V)$ [63], or by fitting the Birch-Murnaghan equation of state.

$$\epsilon_{(C_{11}-C_{22})} = \begin{bmatrix} \delta & 0 & 0 \\ 0 & -\delta & 0 \\ 0 & 0 & \delta^2/(1-\delta^2) \end{bmatrix} \tag{5.14}$$

Second, a volume conserving orthorhombic strain (eq. 5.14) may be applied to the crystal (eq. 5.14). The relation between the relaxed energy and that under strain is symmetric about $E(0)$ and is given by $E(\delta) = E(-\delta) = E(0) + (C_{11} - C_{12})V\delta^2 + O[\delta^4]$ [63]. V us the volume of the relaxed unit cell and $E(0)$ is the minimum energy. By fitting a polynomial to the energy-strain data, the coefficient for the quadratic term is equal to $(C_{11} - C_{12})V$.

$$\epsilon_{(C44)} = \begin{bmatrix} 0 & \frac{\delta}{2} & 0 \\ \frac{\delta}{2} & 0 & 0 \\ 0 & 0 & \delta^2/(4 - \delta^2) \end{bmatrix} \quad (5.15)$$

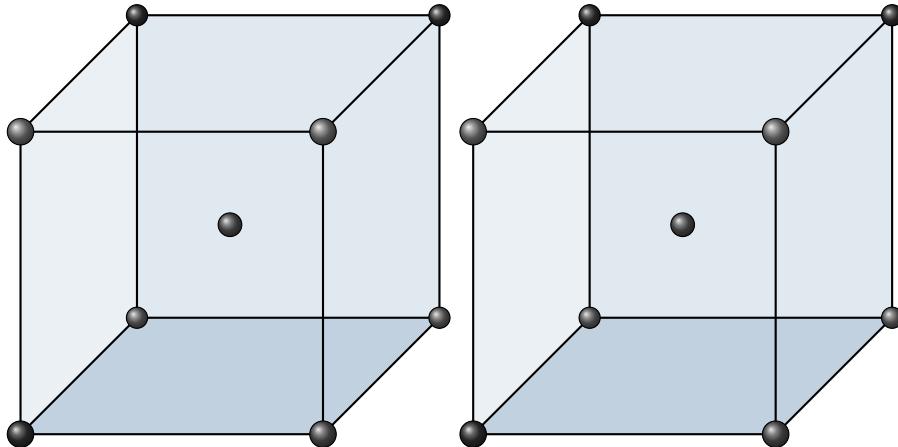
Third, a volume conserving monoclinic strain (eq. 5.15) is applied and the relation between the relaxed energy, strained energy and the elastic constant C_{44} is $E(\delta) = E(-\delta) = E(0) + \frac{1}{2}C_{44}V\delta^2 + O[\delta^4]$. Similarly, fitting a polynomial to the strain-energy data points will calculate the value of C_{44} .

Finally, the relation between the bulk modulus, C_{11} and C_{12} , $B_0 = (C_{11} + 2C_{12})/3$ allows the computation of the individual constants eq. 5.16 (this relationship is only for cubic crystals).

$$\begin{aligned} C_{ortho} &= C_{11} - C_{12} \\ C_{11} &= \frac{3B_0 + 2C_{ortho}}{3} \\ C_{12} &= \frac{3B_0 - C_{ortho}}{3} \end{aligned} \quad (5.16)$$

5.2.7 Calculating Elastic Constants for Orthorhombic Crystal

The DFT work here includes Palladium and Iron. The natural arrangement of Pd atoms in a pure sample are FCC within a cubic crystal. Pure iron at room temperature is BCC, but this work is interested in austenitic stainless steel where the structure of atoms in the alloy are FCC. When modelling FCC iron using DFT with a non-polarized calculation, the crystal favours a cubic crystal with the atoms fixed in the FCC positions. When a spin-polarized calculation is computed, with magnetization along the x-axis, the crystal becomes tetragonal (once again, the atoms are fixed in FCC positions).



$$C_{ij} = \begin{bmatrix} C_{11} & C_{12} & C_{12} & 0 & 0 & 0 \\ C_{12} & C_{11} & C_{12} & 0 & 0 & 0 \\ C_{12} & C_{12} & C_{11} & 0 & 0 & 0 \\ 0 & 0 & 0 & C_{44} & 0 & 0 \\ 0 & 0 & 0 & 0 & C_{44} & 0 \\ 0 & 0 & 0 & 0 & 0 & C_{44} \end{bmatrix} \quad (5.17)$$

(3 independent values)

$$C_{ij} = \begin{bmatrix} C_{11} & C_{12} & C_{13} & 0 & 0 & 0 \\ C_{12} & C_{22} & C_{23} & 0 & 0 & 0 \\ C_{13} & C_{23} & C_{33} & 0 & 0 & 0 \\ 0 & 0 & 0 & C_{44} & 0 & 0 \\ 0 & 0 & 0 & 0 & C_{55} & 0 \\ 0 & 0 & 0 & 0 & 0 & C_{66} \end{bmatrix} \quad (9 \text{ independent values}) \quad (5.18)$$

Once the optimised parameters have been determined for the orthorhombic crystal, nine strains are applied to the crystal [64] in order to calculate the nine independent elastic constants.

$$E(V, \sigma) = E(V_0, 0) + V_0 \left(\sum_i \tau_i \epsilon_i \sigma_i + \frac{1}{2} \sum_{ij} c_{ij} \sigma_i \epsilon i \sigma_j \epsilon j \right) + O(\sigma^3) \quad (5.19)$$

The first three strains applied to the orthorhombic crystal

$$D_1 = \begin{bmatrix} 1 + \delta & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (5.20)$$

$$E(V, \sigma) = E(V_0, 0) + V_0 \left(\tau_1 \sigma + \frac{c_{11}}{2} \sigma^2 \right) \quad (5.21)$$

$$D_2 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 + \delta & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (5.22)$$

$$E(V, \sigma) = E(V_0, 0) + V_0 \left(\tau_2 \sigma + \frac{c_{22}}{2} \sigma^2 \right) \quad (5.23)$$

$$D_3 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 + \delta \end{bmatrix} \quad (5.24)$$

$$E(V, \sigma) = E(V_0, 0) + V_0 \left(\tau_3 \sigma + \frac{c_{33}}{2} \sigma^2 \right) \quad (5.25)$$

Volume conserving monoclinic distortions are then applied to the crystal to calculate the C_{11} , C_{22} and C_{33} elastic constants.

$$D_4 = \begin{bmatrix} \frac{1}{(1-\sigma^2)^{\frac{1}{3}}} & 0 & 0 \\ 0 & \frac{1}{(1-\sigma^2)^{\frac{1}{3}}} & \frac{\sigma}{(1-\sigma^2)^{\frac{1}{3}}} \\ 0 & \frac{\sigma}{(1-\sigma^2)^{\frac{1}{3}}} & \frac{1}{(1-\sigma^2)^{\frac{1}{3}}} \end{bmatrix} \quad (5.26)$$

$$E(V, \sigma) = E(V_0, 0) + V_0 \left(2\tau_4 \sigma + 2 \frac{c_{44}}{2} \sigma^2 \right) \quad (5.27)$$

$$D_5 = \begin{bmatrix} \frac{1}{(1-\sigma^2)^{\frac{1}{3}}} & 0 & \frac{\sigma}{(1-\sigma^2)^{\frac{1}{3}}} \\ 0 & \frac{1}{(1-\sigma^2)^{\frac{1}{3}}} & 0 \\ \frac{\sigma}{(1-\sigma^2)^{\frac{1}{3}}} & 0 & \frac{1}{(1-\sigma^2)^{\frac{1}{3}}} \end{bmatrix} \quad (5.28)$$

$$E(V, \sigma) = E(V_0, 0) + V_0 \left(2\tau_5 \sigma + 2 \frac{c_{55}}{2} \sigma^2 \right) \quad (5.29)$$

$$D_6 = \begin{bmatrix} \frac{1}{(1-\sigma^2)^{\frac{1}{3}}} & \frac{\sigma}{(1-\sigma^2)^{\frac{1}{3}}} & 0 \\ \frac{\sigma}{(1-\sigma^2)^{\frac{1}{3}}} & \frac{1}{(1-\sigma^2)^{\frac{1}{3}}} & 0 \\ 0 & 0 & \frac{1}{(1-\sigma^2)^{\frac{1}{3}}} \end{bmatrix} \quad (5.30)$$

$$E(V, \sigma) = E(V_0, 0) + V_0 \left(2\tau_6 \sigma + 2 \frac{c_{66}}{2} \sigma^2 \right) \quad (5.31)$$

$$D_7 = \begin{bmatrix} \frac{1+\sigma}{(1-\sigma^2)^{\frac{1}{3}}} & 0 & 0 \\ 0 & \frac{1-\sigma}{(1-\sigma^2)^{\frac{1}{3}}} & 0 \\ 0 & 0 & \frac{1}{(1-\sigma^2)^{\frac{1}{3}}} \end{bmatrix} \quad (5.32)$$

$$E(V, \sigma) = E(V_0, 0) + V_0 \left((\tau_1 - \tau_2 \sigma + \frac{1}{2}(c_{11} + c_{22} - 2c_{12})\sigma^2 \right) \quad (5.33)$$

$$D_8 = \begin{bmatrix} \frac{1+\sigma}{(1-\sigma^2)^{\frac{1}{3}}} & 0 & 0 \\ 0 & \frac{1}{(1-\sigma^2)^{\frac{1}{3}}} & 0 \\ 0 & 0 & \frac{1-\sigma}{(1-\sigma^2)^{\frac{1}{3}}} \end{bmatrix} \quad (5.34)$$

$$E(V, \sigma) = E(V_0, 0) + V_0 \left((\tau_1 - \tau_3 \sigma + \frac{1}{2}(c_{11} + c_{33} - 2c_{13})\sigma^2 \right) \quad (5.35)$$

$$D_9 = \begin{bmatrix} \frac{1}{(1-\sigma^2)^{\frac{1}{3}}} & 0 & 0 \\ 0 & \frac{1+\sigma}{(1-\sigma^2)^{\frac{1}{3}}} & 0 \\ 0 & 0 & \frac{1-\sigma}{(1-\sigma^2)^{\frac{1}{3}}} \end{bmatrix} \quad (5.36)$$

$$E(V, \sigma) = E(V_0, 0) + V_0 \left((\tau_2 - \tau_3 \sigma + \frac{1}{2}(c_{22} + c_{33} - 2c_{23})\sigma^2 \right) \quad (5.37)$$

5.2.8 Stability Conditions

Whilst the elastic constants may be computed using first-principles calculations, there are a series of assumptions and approximations made in order to solve first-principles calculations in a reasonable amount of time. There

must be a sanity check, as it is no use to compute the elastic constants from DFT to find the crystals are unstable.

For a cubic crystal with three independent elastic constants and zero stress, the Born elastic stability criteria apply[65]. The bulk modulus should be positive $(C_{11} + 2C_{12})/3 > 0$, the tetragonal shear should be positive $C_{11} - C_{12} > 0$ and the shear modulus should be positive $c_{44} > 0$.

For an orthorhombic crystal, there are additional conditions given there are nine independent elastic constants^{5.38}[66].

$$\begin{aligned} C_{11} &> 0, C_{22} > 0, C_{33} > 0, C_{44} > 0, C_{55} > 0, C_{66} > 0 \\ (C_{11} + C_{22} - 2C_{12}) &> 0, (C_{11} + C_{33} - 2C_{13}) > 0, (C_{22} + C_{33} - 2C_{23}) > 0 \\ (C_{11} + C_{22} + C_{33} + 2C_{12} + 2C_{13} + 2C_{23}) &> 0 \end{aligned} \quad (5.38)$$

5.2.9 Computing Values from Elastic Constants

The bulk modulus, as noted earlier, is a measure of the effect of strain on stress. While it may be calculated by taking the second derivative of energy with respect to volume, or by fitting an equation of state, it may also be calculated using the elastic constants of a material or the compliance constants.

For cubic crystals the elastic constants may be used to calculate the bulk modulus, tetragonal shear, shear modulus and cauchy pressure^{5.39}.

$$\begin{aligned} B_0 &= \frac{C_{11} + 2C_{12}}{3} \text{ Bulk Modulus} \\ C' &= \frac{C_{11} - C_{12}}{2} \text{ Tetragonal Shear} \\ &\qquad\qquad\qquad C_{44} \text{ Shear Modulus} \\ p_c &= C_{12} - C_{44} \text{ Cauchy Pressure} \end{aligned} \quad (5.39)$$

For an orthorhombic crystal, the Voight bulk modulus and Reuss bulk modulus may be calculated from the stiffness and compliance matrix.

$$B_V = \frac{1}{9} (C_{11} + C_{22} + C_{33} + 2(C_{12} + C_{13} + C_{23})) \quad (5.40)$$

$$B_R = (S_{11} + S_{22} + S_{33} + 2(S_{12} + S_{13} + S_{23}))^{-1} \quad (5.41)$$

Calculation by this method is particularly useful for this work, where the FCC iron crystal is orthorhombic, and not cubic.

5.2.10 Correlation of Melting Temperature in Metals with Elastic Constants

There is a correlation between the melting temperature of metals and their elastic constants [67].

$$T \approx 598.0 + 6.66 \times (C_{11} + C_{22} + C_{33}) - 0.003 * (C_{11} + C_{22} + C_{33}) \quad (5.42)$$

Where the temperature is in K, and the elastic constants are in GPA

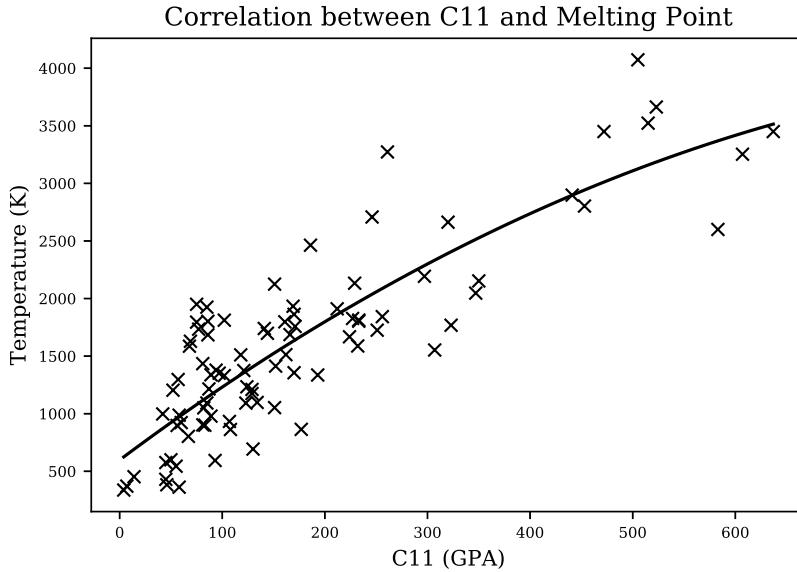


Figure 5.2: Graph caption

The correlation values between the temperature and C_{11} are 0.848 and 0.780 for the Pearsons and Spearmans correlation respectively. An accurate temperature will not be possible to predict, but it will act as a sanity check within the computer codes developed in the results section of this work.

5.2.11 Poisson Ratio

The Poisson ratio is a measure of how much a material changes in the direction perpendicular to a strain applied transversely.

$$\nu = -\frac{d\epsilon_{perp}}{d\epsilon_{transverse}} \quad (5.43)$$

The bulk modulus may be calculated from the equation of state or elastic constants, and the shear modulus may be calculated from the elastic constants. In turn, the Poisson ratio is calculated from these.

$$\nu = \frac{3B_0 - 2G}{2(3B + G)} \quad (5.44)$$

The ratio may be used as another data point in order to fit a potential, or at the very least as a check to compare to the known value once a potential has been derived.

5.3 Interatomic Potentials

5.3.1 Introduction

An interatomic potentials, as used in this work and Molecular Dynamics computer codes, is a function or a set of functions that describe the energy and force between atoms. Simpler functions represent the potential energy between pairs of atoms only, but more complicated functions have been used in molecular dynamics since the 1980s that also attempt to represent the many body nature of materials, which applies in particular to metals.

5.3.2 Pair Potentials

A pair potential only considers pairs of nearby atoms, one pair at a time, and does not consider the effect of any other nearby atoms. Where an alloy is being modelled, there will be a pair potential function for each element and element combination; 1 for a single element, 3 for a two element alloy, 6 for a three element alloy and so on.

Lennard-Jones

The Lennard-Jones potential was proposed in the 1920s and has both an attractive term and a repulsive term; the $(r_m/r)^{12}$ becomes much larger than the $(r_m/r)^6$ term at close distances, and this mimics the coulomb repulsion as two atoms are pushed closer together. At larger seperations, the attractive term dominates.

$$V(r) = e \left(\left(\frac{r_m}{r} \right)^{12} - 2 \left(\frac{r_m}{r} \right)^6 \right) \quad (5.45)$$

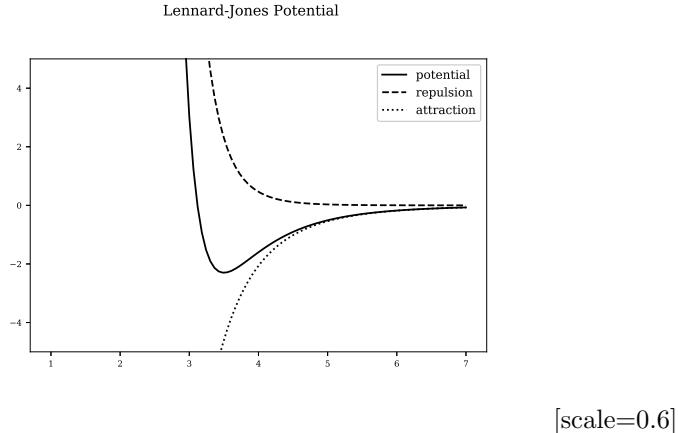


Figure 5.3: Lennard-Jones Potential

Morse Potential

The Morse potential was proposed five years after the Lennard-Jones potential. It also has an attractive and repulsive term, but it uses the exponential function rather than 6th and 12th powers.

$$V(r) = \exp(-2a(r - re)) - 2\exp(-a * (r - re)) \quad (5.46)$$

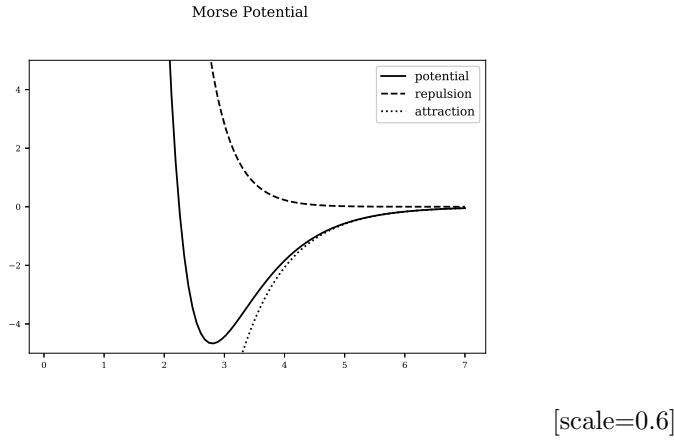


Figure 5.4: Morse Potential

Buckingham Potential

The Buckingham Potential consists of a repulsive and attractive part. As the separation decreases, the attractive term dominates and the overall function tends towards negative infinity. This shouldn't be problematic when gasses or solids are being modelled alone, but when collisions and damage cascades are being modelled, the separation between atoms may be much smaller than that in a typical simulation, ending with these atoms being pulled together.

$$V(r) = A * \exp(-B * r) - \frac{C}{r * * 6} \quad (5.47)$$

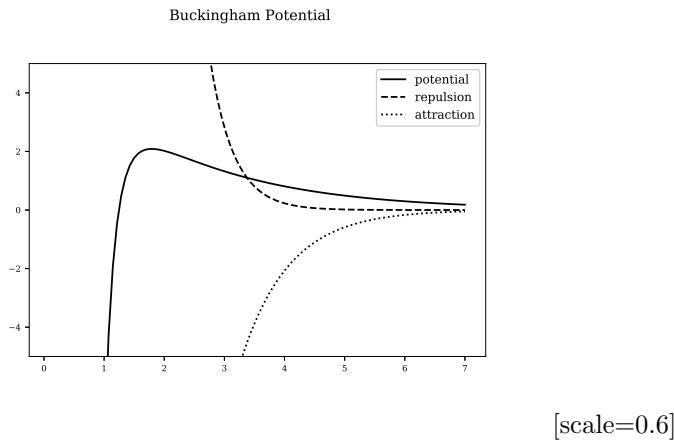


Figure 5.5: Buckingham Potential

ZBL Universal Potential Function

It became clear, while experimenting with a number of existing potentials and molecular dynamics programs, that at the very least modifications to those potentials would need to be made for small atom separations. A simulation to model a projectile failed early on due to the projectile's proximity to target atoms, resulting in the MD code returning an error as the separation was out of the range of the potential.

The Ziegler Biersack Littmark (ZBL) potential, between an atom of charge Z_i and Z_j is set out in the SRIM manual[68].

$$V_{zbl}(r_{ij}, Z_i, Z_j) = \frac{1}{4\pi\epsilon_0} \frac{Z_i Z_j}{r_{ij}} \phi\left(\frac{r_{ij}}{a_{ij}}\right) \quad (5.48)$$

where $\epsilon_0 = 8.85419 \times 10^{-12}$

The parameters of the Universal screening potential function are as follows:

$$\phi(x) = 0.181e^{-3.2x} + 0.5099e^{-0.9423x} + 0.2802e^{-0.4029x} + 0.02817e^{-0.2016x} \quad (5.49)$$

where $a_{ij} = \frac{0.8854a_0}{Z_i^{0.23} + Z_j^{0.23}}$
and $a_0 = 0.529\text{angstrom}$

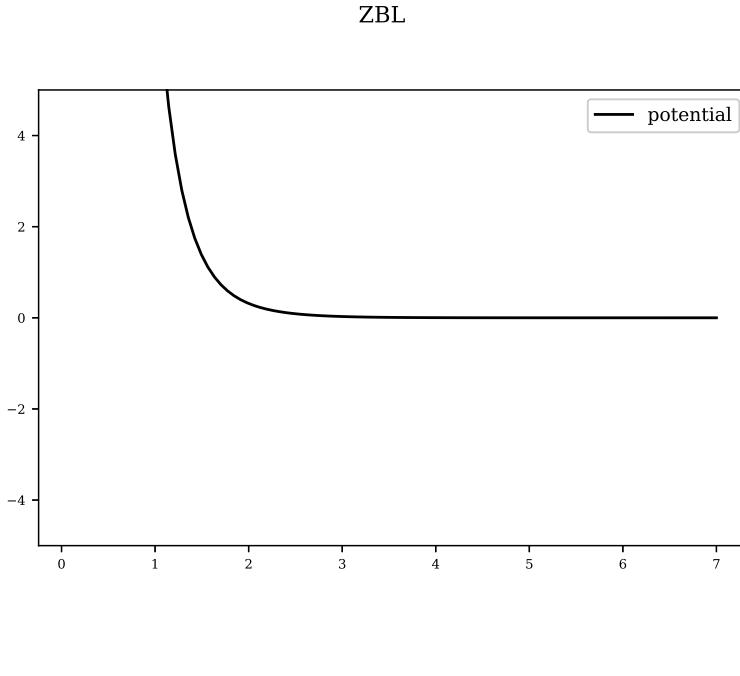


Figure 5.6: Graph caption

5.3.3 Many Body Potentials

This work is focused on metals, and two popular, and closely related potentials, will be discussed in this section.

Finnis-Sinclair

The Finnis-Sinclair potential was published in 1984[69] and it introduced both a pair potential and an embedded term to take into account the cohesive energy dependent on the local electron density. The pair term represents the repulsion between the atoms whereas the embedding functional glues the atoms together in the solid.

The tight-binding that this potential is based on is represented by the functional: square root of the density. The potential for a single element model consists of a pair function, a density function and a tight-binding functional.

$$\begin{aligned}
u^{A,A} &= \sum_{i \neq j}^N (r) v^{A,A}(r_{ij}) \\
rho_i^A(r) &= \sum_{j,j \neq i}^N \phi(r_{ij}) \\
F^A &= \sum_i^N
\end{aligned} \tag{5.50}$$

Embedded Atom Method

The Embedded atom method was also developed in the 1980s. It was developed with metals in mind, and in many ways is a more flexible variant of the Finnis-Sinclair potential. It too has a pair and density function, but the functional of the EAM potential is not restricted to a square root.

$$\begin{aligned}
U_{EAM} &= \frac{1}{2} \sum_{i=1}^N \sum_{j \neq i}^N V_{ij}(r_{ij}) + \sum_{i=1}^N F[\rho_i] \\
\text{where } \rho_i &= \sum_{j=i, j \neq i}^N \rho_{ij}(r_{ij})
\end{aligned} \tag{5.51}$$

Elastic properties were not reliably calculated from pair functions alone[70], but the advent of Finnis-Sinclair and EAM type potentials changed this.

$$A_{ij} + F'(\vec{\rho})V_{ij} = 0$$

where

$$\begin{aligned}
A_{ij} &= \frac{1}{2} \sum_m \phi'_m a_i^m a_j^m / a^m \\
V_{ij} &= \frac{1}{2} \sum_m \rho'_m a_i^m a_j^m / a^m
\end{aligned} \tag{5.52}$$

$$C_{11} = (B_1 1 + F'(\vec{\rho})W_{11} + F''(\vec{\rho})(V_{11})^2) / \Omega_0$$

$$C_{12} = (B_1 2 + F'(\vec{\rho})W_{12} + F''(\vec{\rho})(V_{11})^2) / \Omega_0$$

$$C_{44} = (B_1 2 + F'(\vec{\rho})W_{12}) / \Omega_0$$

where

$$\begin{aligned}
B_{ijkl} &= \frac{1}{2} \sum_m (\phi''_m - \phi'_m / a^m) a_i^m a_j^m a_k^m a_l^m / (a^m)^2 \\
W_{ijkl} &= \frac{1}{2} \sum_m (\rho''_m - \rho'_m / a^m) a_i^m a_j^m a_k^m a_l^m / (a^m)^2
\end{aligned} \tag{5.53}$$

where

$$\phi''_m = (d^2 \phi(r) / dr^2)_{r=a^m},$$

$$\rho''_m = (d^2 \rho(r) / dr^2)_{r=a^m}.$$

For homonuclear cubic crystals, the lattice constant may be calculated by the equilibrium condition in eq 5.52.

The three independent elastic constants, C_{11}, C_{12}, C_{44} , may also be calculated at equilibrium from eq 5.53. By removing the embedding functional, $A_{ij} = 0$, and this leads to the

5.3.4 Two Band Embedded Atom Method

There are several variations of the EAM potential, and one of particular interest to us is the two-band model EAM (2BMEAM) that has two electron density and embedding energy terms. This formalism was originally developed to model Caesium[71], and the transition of electrons between S and D bands under pressure, but it has been modified to apply to alloys.

Caesium changes its structure as pressure is applied. The first change is from BCC to a more compact FCC structure, but it then compresses further as electrons move from the S-band to the more compact D-band, reducing the size of the atom.

The bond energy may be described with eq 5.54 where N_1 and N_2 are the capacities of each band, n_{i1} and n_{i2} are the electrons in each band of the i^{th} atom. E_{prom} is the energy required to promote an electron from band 1 to band 2.

$$U_{\text{bond}} = \sum_i \frac{W_{i1}}{2N_1} n_{i1}(n_{i1} - N_1) + \sum_i \frac{W_{i2}}{2N_2} n_{i2}(n_{i2} - N_2) + E_{\text{prom}} \quad (5.54)$$

A Finnis-Sinclair type EAM potential was used by Ackland and Reed in this work for both bands.

$$\begin{aligned} & \text{Pair} \\ V_s(r_{ij}) &= \sum_i \frac{A_s}{r_{ij}^{12}} \\ V_d(r_{ij}) &= \sum_i \frac{A_d}{r_{ij}^{12}} \end{aligned} \quad (5.55)$$

$$\rho_s(r_{ij}) = \begin{cases} C_s(d_s - r_{ij})^3 & r_{ij} < d_s \\ 0 & r_{ij} > d_s \end{cases} \quad \rho_d(r_{ij}) = \begin{cases} C_d(d_d - r_{ij})^3 & r_{ij} < d_d \\ 0 & r_{ij} > d_d \end{cases} \quad (5.56)$$

$$\text{Embedding} \quad (5.57)$$

C_s	$0.05617 eV^2 \text{angs}^{-3}$
C_d	$0.1681 eV^2 \text{angs}^{-3}$
d_s	9.5097angs
d_d	6.9189angs
A_s	$2.417 \times 10^7 \text{angs}^{12}$
A_d	$3.7668 \times 10^6 \text{angs}^{12}$

Table 5.3: Caesium 2BEAM parameters

The parameters fitted in Ackland and Reed's work are listed in table ??.

The implementation of this two-band potential predicted the transformation of Caesium. The I phase, at ambient pressure, was BCC Caesium with a primitive cell volume of 115.9 cubic angstrom per atom. As more

pressure was added, the optimum structure changed from BCC to FCC, and this resulted in a smaller volume per atom of 67.5 cubic angstrom. Finally Caesium undergoes an isostructural transformation, and the potential predicts a volume of 48.7 cubic angstrom per atom.

There is a transition pressure of 4.3 GPa between the phases II and III. The potentials were "in good agreement with experiment" [71].

An alloy version of the two-band model was developed by Olsson et al. to investigate the -prime phase formation in Fe-Cr[72]. Fe-Cr alloys form ferromagnetic alloys with concentrations of up to 10% chromium at 750°C. As the concentration of chromium in the alloy increases, the alloy begins to decompose into iron rich and chromium rich precipitates, and this decomposition is accelerated under irradiation.

The two-band method for Caesium was extended to an Fe-Cr alloy in order to describe the heat of mixing in the alloy.

$$U_{EAM} = \frac{1}{2} \sum_{i=1}^N \sum_{j \neq i}^N V_{ij}(r_{ij}) + \sum_{i=1}^N F_D[\rho_{d,i}] + \sum_{i=1}^N F_S[\rho_{s,i}] \quad (5.58)$$

where $\rho_{d,i} = \sum_{j=i, j \neq i}^N \rho_{d,ij}(r_{ij})$ and $\rho_{s,i} = \sum_{j=i, j \neq i}^N \rho_{s,ij}(r_{ij})$

The embedding functional (eq. 5.62) was in the form of several functionals used in papers by Ackland and Mendelev, and an extension to the standard Finnis-Sinclair.

$$F_{band}(\rho_{band}) = A_1^{band} \sqrt{\rho_{band}} + A_2^{band} \rho_{band}^2 + A_3^{band} \rho_{band}^4 \quad (5.59)$$

The choice of function for both the pair and d-band density functions was a cubic spline, and a 4s Slater type function was used for the s-band alloy density. The Fe-Fe and Cr-Cr s-band density functions were zero functions. Where the alloy has high concentrations of Iron or Chromium, the respective d-band density functions will dominate, but as the elements mix, the s-band density will also contribute.

$$V(r) = \sum_i a_i (r - r_i)^3 H(r_i - r) \text{ where } H \text{ is the Heaviside Step function} \quad (5.60)$$

$$\phi_d^{CrCr}(r) = \sum_k b_k (r - r_k)^3 H(r_k - r) \text{ where } H \text{ is the Heaviside Step function} \quad (5.61)$$

$$\begin{aligned} \phi_s^{FeCr}(r) &= (H_s r^3 \exp(-\xi_s r))^2 \\ \phi_s^{FeFe}(r) &= 0.0 \\ \phi_s^{CrCr}(r) &= 0.0 \end{aligned} \quad (5.62)$$

The resulting potential, with the s-density fitted to the mixing enthalpy of Iron and Chromium, reproduces the formation of alpha-prime phase Cr with thermal ageing over a range of Cr concentrations.

The Fe-Cr potentials were further developed by Bonny et al. The mixing enthalpy changes sign in Fe-Cr, having a positive mixing enthalpy for Cr concentrations above 10% and a negative mixing enthalpy below. This results

in the solubility of Cr in the alloy at lower concentrations, but as the concentration rises there's a tendency for Cr to form α' precipitates[73].

In the work by Bonny et al, the base Iron and Chromium potentials were those developed by Mendelev and Ackland, and from the Fe-Cr potentials of Olsson and Wallenius. In total, 11 functions make up the Bonny et al potential, higher than the usual 7 functions required for a two species EAM. In a standard EAM potential, the density contribution from an atom of one species is only dependent on that contributing atom, and not the embedded atom. This potential splits the density function into multiple bands and multiple permutations of atom species.

$$\begin{aligned}\rho_{AA}^d(r) &= \rho_{BA}^d = \rho_A^d \\ \rho_{BB}^d(r) &= \rho_{AB}^d = \rho_B^d \\ \rho_{AB}^s(r) &= \rho_{BA}^s \\ \rho_{AA}^s(r) &= \rho_{BB}^s = 0\end{aligned}\tag{5.63}$$

The chosen s-band density function for the Fe-Cr alloy was selected as an exponential style function with a cut-off function.

$$\rho_{FeCr}^s(r) = kr^6 \exp(-2\xi r) g_{cut}(r)\tag{5.64}$$

$$g_{cut}(r) = \begin{cases} 1 & r \leq r_c^i \\ \frac{1}{2} \left(1 - \sin\left(\frac{\pi}{2} \frac{r-r_m}{d}\right)\right) & r_c^i < r < r_c^f \\ 0 & r_c^f < r \end{cases}\tag{5.65}$$

$$F^s(\rho) = A_1\sqrt{\rho} + A_2\rho^2\tag{5.66}$$

The magnetic properties of Chromium were considered during the development of these potentials. First-Principles calculations are equivalent to calculating properties at 0K, and at this temperature Chromium is antiferromagnetic. The Neel temperature for Cr, the point at which it transitions from an antiferromagnetic to a paramagnet, is 310K. The operating temperature of the alloys, within a reactor, will be above the Neel temperature. Chromium has a positive Cauchy pressure as a paramagnet, and a negative Cauchy pressure under the Neel temperature when an antiferromagnetic, and as a result of this and the operating temperature, a positive Cauchy pressure was used to fit the potential.

5.3.5 Many Bands for the Embedded Atom Method

$$\begin{aligned}U_{EAM} &= V_{pair} + \sum_b^B F_b(\rho_b) \frac{1}{2} \sum_{i=1}^N \sum_{j \neq i}^N V_{ij}(r_{ij}) + \sum_{b=1}^B \sum_{i=1}^N F_b[\rho_{b,i}] \\ \text{where } \rho_{b,i} &= \sum_{j=i, j \neq i}^N \rho_{b,ij}(r_{ij})\end{aligned}\tag{5.67}$$

and B total number of bands, 1, 2, 3 etc

5.3.6 Function Types used by EAM and Two-Band EAM

In previous work, there are a wide range of functions used to represent the pair potential, density and embedding functional. These range from those that preserve and attempt to replicate the physics, to those that do away with knowledge of the physics underlying the potentials altogether. The pair potentials already covered here may also be used as the pair function component of the EAM or Two-Band EAM potentials.

A full list of the functions considered and included in the potential fitting code that resulted from this work is included in the appendix. Only polynomial splines will be discussed further here.

Polynomial Splines

The form of polynomial spline used in the literature is a sum of N cubic polynomials that, by way of the Heaviside step function and their form, cutoff neatly at the desired radius. The cutoff radii are fixed and, during the fit of a potential, just the coefficients of each cubic spline are varied.

$$V(r) = \sum_i^N a_i(r - r_i)^3 H(r_i - r) \quad \text{where} \quad (5.68)$$

$$H(x) = \begin{cases} 0 & x < 0 \\ 1 & x \geq 0 \end{cases}$$

Polynomial Knot to Knot Splines

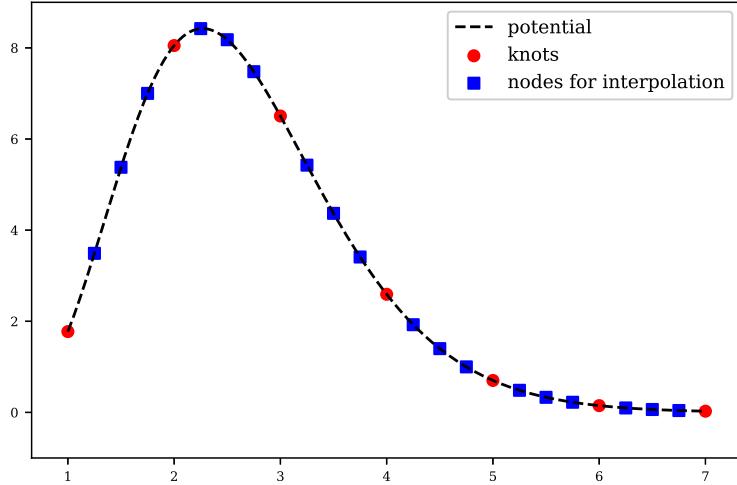
So far, analytic potentials have been discussed. There are existing potentials that do not have an analytic form, but are tabulated data that have the properties that they are continuous and have a continuous first derivatives. In attempting to fit or re-fit tabulated data, it would be problematic to adjust each data point individually given the number of data points and the requirement to have a continuous well behaved function with continuous first derivatives.

The tabulated function is divided into sections, and the start and end of each section forms the knot of the spline. A polynomial may be splined between these knots, but the order of polynomial will depend on the data available. Immediately, the x and y positions are known: $x_a, x_b, f(x_a), f(x_b)$. A first order polynomial has two unknowns, and so these unknowns may be calculated using linear algebra.

$$\begin{aligned} c_0 + c_1 x_a &= y_a \\ c_0 + c_1 x_b &= y_b \end{aligned} \quad (5.69)$$

One of the requirements for the potential is to have a continuous first derivative, and this requires knowledge of the first derivative at each knot. In the figure, the knots are shown as circles. By taking several points near to each knot, shown as squares, the derivative at each knot may be calculated by interpolation. This allows a third order polynomial to be used:

Knot to Knot Spline



[scale=0.6]

Figure 5.7: Polynomial knot to knot spline

$$\begin{aligned}
 P_1 &= (x_A, f(x_A)) \\
 P_2 &= (x_A, f'(x_A)) \\
 P_3 &= (x_B, f(x_B)) \\
 P_4 &= (x_B, f'(x_B))
 \end{aligned} \tag{5.70}$$

$$\begin{aligned}
 c_0 + c_1 x_a + c_2 x_a^2 + c_3 x_a^3 &= y_a \\
 0 + c_1 + 2c_2 x_a + 3c_3 x_a^2 &= y'_a \\
 c_0 + c_1 x_b + c_2 x_b^2 + c_3 x_b^3 &= y_b \\
 0 + c_1 + 2c_2 x_b + 3c_3 x_b^2 &= y'_b
 \end{aligned} \tag{5.71}$$

Rewriting as matrices:

$$\begin{bmatrix} 1 & x_a & x_a^2 & x_a^3 \\ 0 & 1 & 2x_a & 3x_a^2 \\ 1 & x_b & x_b^2 & x_b^3 \\ 0 & 1 & 2x_b & 3x_b^2 \end{bmatrix} \begin{bmatrix} c_0 \\ c_1 \\ c_2 \\ c_3 \end{bmatrix} = \begin{bmatrix} f(x_a) \\ f'(x_a) \\ f(x_b) \\ f'(x_b) \end{bmatrix} \tag{5.72}$$

It is important for the function and its derivative to be continuous. If it is also deemed necessary to have a continuous second derivative, a quintic spline may be used. By interpolating and calculating the second order derivative at each knot, a 5th order polynomial may be splined between pairs of knots.

This approach may be modified to spline functions of the form $\exp(p(r))$ between knots, and this is useful for splining between a ZBL or similar repulsive function at small r and a function of a different form at larger r.

where

$$V(r) = \begin{cases} ZBL(r, q_1, q_2) & r \leq r_a \\ \exp(a + br + cr^2 + dr^3) & r_a < r < r_b \\ v(r) & r_b < r < r_{cut} \\ 0 & r \geq r_{cut} \end{cases} \quad (5.73)$$

To spline the exponential the equations are set up in a similar way, but this time the system of equations are non-linear.

$$\begin{aligned} \exp(c_0 + c_1x_a + c_2x_a^2 + c_3x_a^3) &= y_a \\ (c_1 + 2c_2x_a + 3c_3x_a^2)\exp(c_0 + c_1x_a + c_2x_a^2 + c_3x_a^3) &= y'_a \\ \exp(c_0 + c_1x_b + c_2x_b^2 + c_3x_b^3) &= y_b \\ (c_1 + 2c_2x_b + 3c_3x_b^2)\exp(c_0 + c_1x_b + c_2x_b^2 + c_3x_b^3) &= y'_b \end{aligned} \quad (5.74)$$

Newton-Gauss is used to solve this problem, picking a starting value for the constants in the equation and varying until the function values and derivatives at point A and point B match the values being fitted to.

Tending to Zero at the Cutoff Radius

It is desirable for the functions to be well behaved and continuous. The coulomb force between two charged particles reduces smoothly until theoretically at an infinite separation; it doesn't reach a set separation and abruptly drops to zero. It is impossible and unhelpful to consider an infinite separation which is why the cutoff radius has been introduced. It represents the separation where the potential has reached zero; at the very least it's a trade off between accuracy and the computing time available for the problem.

The pair potential should therefore drop off to 0 at the cut off radius, and be equal to zero for larger separations. The electron density spherically around an atom should also smoothly drop off to zero, and similarly it should be equal to zero for distances larger than the cut off. The embedding energy is dependent on the density at the location of the atom embedded. The embedding function will not necessarily drop off to zero, and it depends on the density and not the separation.

In a molecular dynamics simulation, the neighbour list will usually be built considering atoms slightly more separated than the cut off radius of the functions. This allows the same neighbour list to be used for several time steps, before the need to update.

5.3.7 Modified Embedded Atom Method

All the functions considered so far are spherically symmetrical about the atom, but the probability of finding an electron about an atom isn't always symmetric in space. Whilst s orbitals are spherically symmetric, the p and d (and higher) are not.

The

5.3.8 Calculating Energy, Force and Stress

Molecular dynamics simulations do not need to include the weak or strong force, and the gravitational force between atoms in the simulated material are so weak in comparison to the electromagnetic force, they can

also be ignored. There is a force between all the atoms within a real material, but the electromagnetic force is inversely proportionally to the separation of the atoms. Above a certain separation, the electromagnetic force can also be ignored, in order to simplify the computation.

Neighbour List

A cut off radius can be introduced to limit the number of neighbours. As the lattice parameter decreases, the atoms are brought closer together, and as the cut off radius is increased more atoms are considered to be within the sphere of influence of one another. Both increase the number of neighbours each atom has.

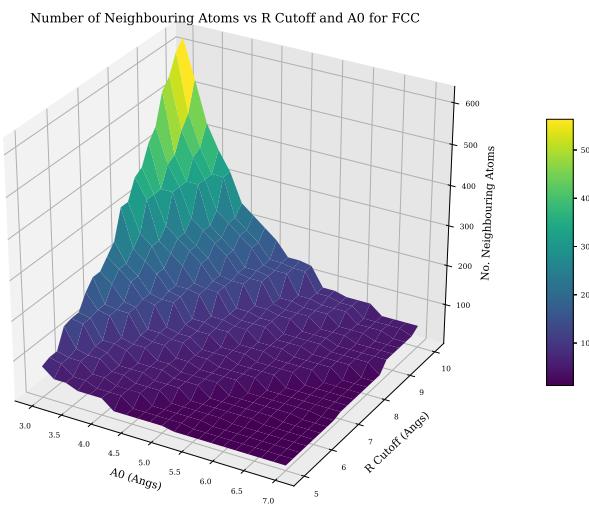


Figure 5.8: Neighbouring atoms with halo

Building a neighbour list may take a long while, depending on the parameters. If a simplistic approach is taken, for N atoms in the supercell, there will be $27N^2$ checks between atoms to see whether they are within the cut off radius of one another. For larger numbers of atoms in a supercell, the whole may be decomposed into smaller domains.

For example, a $16 \times 16 \times 16$ FCC supercell, containing 16,384 atoms, would require looping $27 \times 16,384^2 = 7.25 \times 10^9$ times. Breaking the supercell into 64 smaller domains with 256 atoms in each, reducing the problem to $64 \times 27 \times 256^2 = 1.13 \times 10^8$ loops.

For this work, the supercells used to calculate bulk properties from the interatomic potentials, and the configurations generated by DFT, contain fewer than 1,000 atoms, removing the need to decompose the configuration into smaller sub cells.

Halo Atoms					
04	01	02	03	04	01
16	13	14	15	16	13
12	09	10	11	12	09
Real Atoms					
08	05	06	07	08	05
04	01	02	03	04	01
16	13	14	15	16	13

To build the neighbour list a halo configuration is created such that it lies on top of the real atoms, but extends at least as far as the cut off radius on each side of the real simulation box. Periodic boundary conditions are used to construct the halo.

Once the halo list is computed, the neighbour list may also be computed by looping through the real atoms and halo atoms. A simple pseudo coded sub routine is given below.

Listing 5.1: Simple genetic optimisation subroutine

```

1
2 // real_atoms - array holding coordinates of all real atoms
3 // halo_atoms - array holding coordinates of all halo atoms
4 // real_ids - unique id for each atom
5 // halo_ids - unique atom ids for halo atoms
6 // nlist_ids - array to store ids
7 // nlist_r - array to store separation
8
9 // Define the cutoff and counter start value
10 r_cut = 5.0
11 nl_counter = 1
12
13 // Calculate square
14 r_cut_sq = r_cut ** 2
15
16 // Loop over all real atoms
17 DO n_real = 1, real_atom_count
18   // Loop over all halo atoms
19   DO n_halo = 1, halo_atom_count
20     IF (real_ids(n_real) .LT. halo_ids(n_halo)) THEN
21       r(1:3) = halo_atoms(n_halo, :) - real_atoms(n_real, :)
22       r_sq = SUM(r(1:3) * r(1:3))
23       IF (r_sq .LE. r_cut_sq) THEN
24         r_mag = sqrt(r_sq)
25         nlist_ids(nl_counter, 1) = real_ids(n_real)
26         nlist_ids(nl_counter, 2) = halo_ids(n_halo)
27         nlist_r(nl_counter, 1) = r_mag
28         nlist_r(nl_counter, 2:4) = r(1:3)/r_mag
29         nl_counter = nl_counter + 1
30       END IF
31     END IF
32   END DO
33 END DO

```

The resulting list will contain unique pair combinations; i.e. the pair atom 1 and atom 2 will only be recorded once, and not also as atom 2 and atom 1.

Computing Total Energy

The total energy of the system is the sum of the individual energies for the atoms in the simulation. The type of atom (or pairs of atoms) will determine which function is used.

First, to compute the pair potentials:

- set the total energy of the system equal to zero
- set the starting energy for each atom in the system to zero
- loop through the atom pairs in the entire neighbour list
- for each atom pair, A and B, use the known separation and the potential function to compute the potential energy on atom A due to B and vice versa
- add this potential energy to both A and B
- after looping through the neighbour list, add all the energies due to the pair potential to the total energy

For an EAM or 2BEAM potential, the densities and embedding energies must also be computed:

- set the electron density at the position of (for each) atom to zero
- loop through the atom pairs in the entire neighbour list
- for each atom pair, using the density function for the atom A, the density at atom B due to atom A will be calculated and added to the density at atom B
- if the atoms are both of the same type the same density will be added to atom A due to atom B
- if the atom types are different, using the density function for the atom B, the density at atom A due to atom B will be calculated and added to the density at atom A
- following looping through the neighbour list, to calculate the densities at each atom, the list of atoms will be looped through
- for each atom, the density value at the position of that atom, will be input into the appropriate embedding function to calculate the embedding portion of the energy
- add all the embedding energies to the total energy of the system

For a 2BEAM potential, repeat the above procedure for the second group of density functions and embedding functions (do not repeat calculation of the pair potentials).

Computing Forces on Atoms

In order to calculate the forces on the atoms with an EAM or 2 Band EAM potential, the neighbour list and atom list will need to be looped through several times. First, the pair potential and force due to the pair potential must be calculated by a complete loop through the neighbour list. At the same time, the density at each atom location is also calculated. The embedding energy may then be computed by looping through all the atoms and using the electron density at each atom to give the energy of the atom embedded in that density. The third and final loop will run through the neighbour pairs in the neighbour list once more computing the force on each atom due to it's embedding.

The force on each atom, due to surrounding atoms, may be split into two; force due to the pair potential, and the force due to the embedding of the atom[74].

$$\begin{aligned} F_{pair}^k &= - \sum_{j=1, j \neq k}^N \frac{\partial V_{kj}(r_{kj})}{\partial r_{kj}} \frac{\vec{r}_{kj}}{|\vec{r}_{kj}|} \\ F_{embed}^k &= - \sum_{j=1, j \neq k}^N \left(\frac{\partial F}{\partial \rho_k} + \frac{\partial F}{\partial \rho_j} \right) \frac{\partial \rho(r_{kj})}{\partial r_{kj}} \frac{\vec{r}_{kj}}{r_{kj}} \end{aligned} \quad (5.75)$$

Computing Stress

The stress is be calculated assuming the system is at 0K[75].

$$\tau_{ij} = \frac{1}{2V} \sum_{k, l \in V} \left(x_i^{(l)} - x_i^{(k)} \right) f_j^{(kl)} \quad (5.76)$$

Within the computer code, the individual force components between pairs of atoms are stored as well as an equal sized array of vectors representing their spatial separation. Those pairs with one atom in the halo around the simulation box are used to calculate the stress on the simulation box.

Pseudo Code for the Energy, Force and Stress Subroutine

Listing 5.2: Pseudo Code for Energy and Stress Force Calculation

```

1
2 electron_density[1:n_atoms, 1:bands] = 0.0 // electron density at each atom
3 density_grad_ab[:, :] = 0.0           // grad density function
4 density_grad_ba[:, :] = 0.0           // grad density function
5 energy = 0.0                         // total energy
6 forces[:, :] = 0.0                   // forces each atom, 3D
7 force_between_pairs[:, :] = 0.0       // used to store force between pairs, for stress computation
8 stress[:, :] = 0.0DO
9
10 // LOOP 1 - Pair energy, force and calculate densities
11 DO n = 1, neighbour_count
12   E = get_PairEnergy(atom_a, atom_b, r[n,:])
13   F[:] = get_PairForce(atom_a, atom_b, r[n,:])
14
15 // Save energy
16   energy = energy + E
17
18 // Save Force
19   f[atom_a, :] = f[atom_a, :] - F[:]

```

```

20 f[atom_b, :] = f[atom_b, :] + F[:,]
21 force_between_pairs[:, :] = force_between_pairs[:, :] - F[:,] // Used to calculate stress
22
23 // Loop through density bands
24 DO band = 1, bands
25   // Electron density at A due to atom B
26   electron_density[atom_a, b] = get_Density(atom_b, r)
27   density_grad_ab[n, band] = get_DensityGradient(atom_b, r)
28
29   // Electron density at B due to atom A
30   electron_density[atom_a, band] = get_Density(atom_a, r)
31   density_grad_ba[n, band] = get_DensityGradient(atom_a, r)
32 END DO
33 END DO
34
35 // LOOP 2 - Embedding energy
36 DO n = 1, atom_count
37   // Loop through density bands
38   DO band = 1, bands
39     energy = energy + get_EMBEDDINGEnergy(n, band)
40   END DO
41 END DO
42
43 // LOOP 3 - Embedding force
44 DO n = 1, neighbour_count
45   // Loop through density bands
46   DO band = 1, bands
47     epA = get_EMBEDDINGGradient(atom_a, electron_density(atom_a, b))
48     epB = get_EMBEDDINGGradient(atom_b, electron_density(atom_b, b))
49
50     F[:,] = (epA * density_grad_ba(n, band) + epB * density_grad_ab(n, band)) * r[n, :]
51
52     f[atom_a, :] = f(atom_a, :) - F[:,]
53     f(atom_a, :) = f(atom_a, :) + F[:,]
54
55     force_between_pairs[:, :] = force_between_pairs[:, :] - F[:,] // Used to calculate stress
56   END DO
57 END DO
58
59 // LOOP 4 STRESS
60 DO n = 1, neighbour_count
61   // Only compute if the second atom is in the halo
62   IF(nlisthalo(n)) THEN
63     DO i = 1, 3
64       DO j = 1, 3
65         stress[i, j] = stress[i, j] + (r[n, i] * force_between_pairs[n, j])
66       END DO
67     END DO
68   END IF
69 END DO
70 stress[1:3, 1:3] = stress[1:3, 1:3] / (2.0 * volume)

```

5.3.9 Choice of Functions and Functionals

The EAM potential, for n elements, is made up of n pair functions, $n(n+1)/2$ density functions and $n(n+1)/2$ embedding functionals. The choice of function varies greatly. Irrespective of the choice of function, there are a number of desirable properties:

- the function should be well behaved

- this should be true for the derivative and second derivative - ideally, these should be continuous too
- the pair and density functions should have a cutoff, and the value at this cutoff should be zero

ZBL to Pair Function Spline

Certain functions may behave oddly at small values of atom separation, and this can be seen for the Buckingham potential, where the potential may drop sharply to negative values, pulling the modelled atoms together. The ZBL function is a good choice to model higher energy atoms that do come close together. It may be desirable to keep the characteristics for the

A One Time Neighbour List

As there are only static calculations in this work, the neighbour list will only need to be computed once. The potential functions will change during the fitting process, but the configuration, and neighbourlist, is frozen in place from the start to the end of the fitting procedure.

5.3.10 Sacrificing Physical Elegance

There doesn't exist a generic formula that describes the energy and forces between atoms. Even first principles calculations rely of approximations, and the knowledge of an exact exchange functional eludes us. While it would be satisfying to derive a potential that reflects the fundamental physics, it may be something that is forever out of our reach.

It is more useful to develop a potential that replicates the behavior of DFT calculations (matches the forces, energies, stresses etc) and reproduces material properties such as its elastic constants. If it can do this, and give insights into the material in a way that is not yet possible experimentally, does it matter if the physical elegance is lost?

5.4 Force Matching

To derive a potential, one may approach the problem from first principles in an attempt to replicate reality. It has been more useful, however, to lose any physical elegance [71] to give potentials that work for specific elements under certain conditions. Force data, gathered experimentally or by first-principles calculations, has been used to develop potentials since the 1990s. The force matching method was developed in 1994 by Ercolessi and Adams [76] to link the more accurate, more processor and memory intensive, world of first-principles calculations to Molecular Dynamics.

The force-matching method uses the difference between the actual force (either measured experimentally or calculated by first-principles calculations)

Given a set of M different atomic configurations, and a potential with a set of L parameters (\vec{p}), the function Z_F is a measure of the difference between the the forces calculated using the potential for all configurations and the actual (or DFT generated) forces.

$$Z_F(\vec{\alpha}) = \sum_{k=1}^M \sum_{i=1}^k \sum_{j=1}^3 |F_{i,j}^k(\vec{\alpha}) - F_{i,j}^0|^2 \quad (5.77)$$

This may be extended to include the calculation of other properties, including the cohesive energy of atoms, the lattice parameter, bulk modulus, elastic constants and so on. Each may be weighted depending on how important the property is to the simulation the potential is required for.

$$Z(\vec{p}) = w_F Z_F(\vec{p}) + w_{b0} Z_{b0}(\vec{p}) + w_{e0} Z_{e0}(\vec{p}) + w_{a0} Z_{a0}(\vec{p}) + w_{ec} Z_{ec}(\vec{p}) + w_{ecoh} Z_{ecoh}(\vec{p}) \quad (5.78)$$

5.5 Magnetism

5.5.1 Brief History of Magnetism

Magnetism as a phenomenon has been known to our civilisation for thousands of years. The magnetic mineral Lodestone contains the iron oxide Magnetite Fe_3O_4 and it was used by the ancient Greeks over two thousand years ago.

The magnetic compass had been used since the 12th century, but it wasn't until the 17th Century that the Earth was discovered as a magnet itself by William Gilbert.

In the early 1800s, Hans Christian Oersted discovered that an electric current could move a compass needle, and this lead to much research into the connection between electricity and magnetism. Experimental work by Michael Faraday lead to the discovery that a changing magnetic field produces an electric field. By the mid-1800s the mathematical physicist, James Clerk Maxwell, derived his set of equations describing the connection between magnetism and electricity.

$$\partial \cdot \vec{E} = \frac{\rho}{\epsilon_0} \nabla \cdot \vec{B} = 0 \quad \nabla \times \vec{E} = -\frac{1}{c} \frac{\partial \vec{B}}{\partial t} \quad \nabla \times \vec{B} = \frac{1}{c} \left(4\pi \vec{J} + \frac{\partial \vec{E}}{\partial t} \right)$$

5.5.2 Magnetism: Moving Charges and Spin

We are a product of our environment, and that is made all the more clear in the way our intuition led scientific development until relatively recently. General Relativity and Quantum Mechanics were born in the first part of the 20th Century, and they describe the very large and very small realms we do not notice in every day life.

In describing how magnetism comes to be, it may be useful to use classical analogies, but that's all they are.

When a charged particle moves, it creates a magnetic field. The field at some point j created by a charge moving at \vec{v} at point i is calculated:

$$\vec{B} = \frac{\mu_0 q}{4\pi r_{ij}^3} \vec{v} \times \vec{r}_{ij} \quad (5.79)$$

Electrons do not orbit around a nucleus in the classical sense, but they do have an orbital angular momentum. The overall magnetic field created by an atom is made up of:

- orbital angular momentum of the electron
- spin of electrons
- spin of nucleons

The contribution of the nucleus to the magnetic field is small, and the orbital and spin of electrons are mostly responsible.

$$\vec{m}_z = \mu_B m_l$$

where $m_l = 0, + - 1, + - 2, \dots$

and $\mu_B = \frac{e\hbar}{2m_e}$

(5.80)

Spin is a property of quantum mechanics and it doesn't have an equivalent in classical mechanics, although it may be thought of similar to a spinning top or a rotating planet. Electrons are point like particles, and do not actually rotate, but they do have an intrinsic angular moment (spin). Where a spinning top might have a range of angular velocities, electrons have a quantised value and, as fermions, they have half integer spin.

5.5.3 Ferromagnetism and Antiferromagnetism

Hund's Rule

Electrons fill the shells of a ground state atom such that the energy is minimised. Four quantum numbers are used to describe electrons bound to an atom: n , l , m_l and m_s . The Pauli exclusion principle states that the electrons, which are fermions, cannot have the same quantum numbers as another electron bound to that atom.

Electrons may be spin up or spin down so, two electrons can have the same n , l , m_l with two m_s "slots" available. The energy is minimised by not pairing up and down electrons until all the free "slots" due to the coulomb interaction between the electrons. Once each slot contains one electron, they begin to pair.

The electronic configuration of Iron highlights this:

$$\begin{aligned}
 & 1s2 \uparrow\downarrow \\
 & 2s2 \uparrow\downarrow \\
 & 2p6 \uparrow\downarrow \uparrow \uparrow \\
 & 3s2 \uparrow\downarrow \\
 & 3p6 \uparrow\downarrow \uparrow \uparrow \\
 & 4s2 \uparrow\downarrow \\
 & 3d7 \uparrow\downarrow \uparrow \uparrow \uparrow \uparrow
 \end{aligned} \tag{5.81}$$

Rather than fill the 3d shell from left to right, the first five slots take electrons with spin in the same direction, and the remaining two electrons fill two of the slots with spin opposite to the first five of the shell.

Being fermions the wavefunction for the two electrons must be antisymmetric, with spins opposite to one another. Electrons with the same spin will repel one another and this causes an increase in the screening between the electrons and the nucleus. The screening lowers the attraction between the electrons and the nucleus that then results in the total energy of the atom decreasing[77].

5.6 Density Functional Theory

Density Functional Theory (DFT) is a branch of quantum chemistry that approximately solves the Schrödinger equation using electron density, rather than the coordinates of each electron in the system. A number of

simplifications are also applied in order for DFT to be practical to use, but despite this calculations are limited to just hundreds or thousands of atoms. A calculation of a hundred or so atoms may take thousands of CPU hours at the time of writing, depending on the type of calculation and complexity of the electron structures of the atoms involved.

It is through DFT that the first principles energy, stress and force calculations will be made, and it's these results that the EAM potentials will be trained and fit to using the force matching method. This will allow much larger scale modelling using the extrapolated behaviour of accurate DFT calculations.

5.6.1 Brief Overview of DFT

Several important theories and approximations are used by DFT with the aim of calculating and minimising energies and forces. The Born-Oppenheimer approximation separates the electron-nucleus wave function. It treats the nuclei as fixed points, and the system of electrons in a fixed potential created by the nuclei.

The DFT of Kohn, Sham and Hohenberg proved that the potential of a system is uniquely determined by its ground state density. This makes solving the Schrodinger equation significantly easier.

5.6.2 Time Independent Schrdinger Equation

The Schrodinger equation is a linear partial differential wave equation, and it was proposed by Erwin Schrodinger in 1925. There is a time-dependent and time independent form of the equation, and as the DFT calculations in this work are static, only the time-independent version will be discussed.

$$\hat{H}|\Psi\rangle = E|\Psi\rangle \quad (5.82)$$

The Hamiltonian \hat{H} is an operator, and it is the total energy in the system. Ψ is the wave function (for example, the wave function of an electron) and this contains all the measurable information possible about whatever it represents. E is the energy eigenvalue of this system and this will depend on the eigenstate of the system. The wave function is also connected to the probability of a particle being found at a certain point in space, and the integral over all space is equal to 1 i.e. the probability of it being found somewhere in space is equal to 1.

$$\int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \Psi(x, y, z) dx dy dz \quad (5.83)$$

The Schrodinger equation is set up depending on the system being studied. Starting with the simplest, a free particle, the only non zero energy of then Hamiltonian is kinetic.

$$\begin{aligned} \hat{H} &= \frac{\hbar^2}{2m} \nabla^2 \\ -\frac{\hbar^2}{2m} \nabla^2 \Psi(\vec{r}) &= E \Psi(\vec{r}) \end{aligned} \quad (5.84)$$

If the particle is in a potential, it will have both kinetic energy \hat{T} and potential energy \hat{V} .

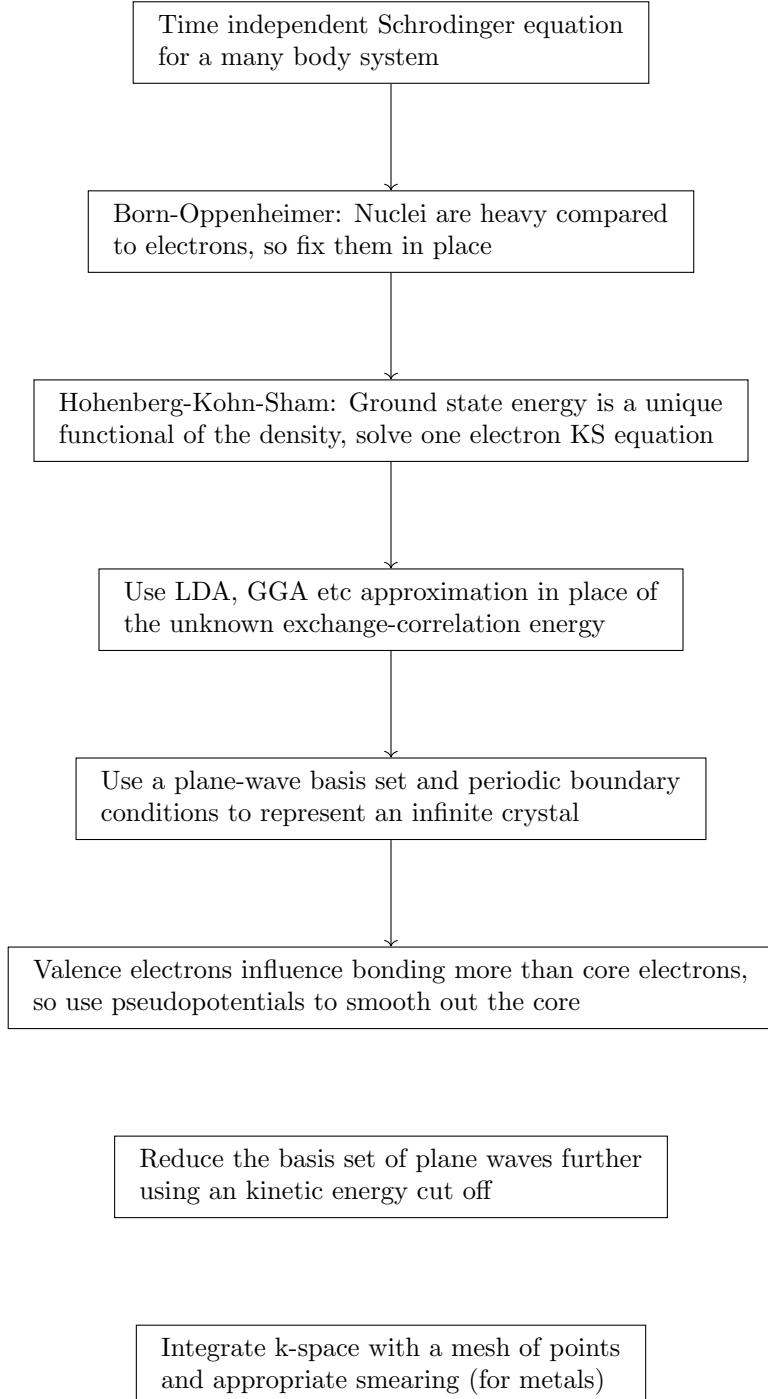


Figure 5.9: Common approximations used to enable DFT calculations

$$\hat{H} = \hat{T} + \hat{V} = \frac{\hbar^2}{2m} \nabla^2 + V(\vec{r})$$

$$\left[-\frac{\hbar^2}{2m} \nabla^2 \right] \Psi(\vec{r}) = E \Psi(\vec{r}) \quad (5.85)$$

5.6.3 Many Body TISE

Electronic structure calculations are important and allow the calculation of material properties that may be difficult or impossible to measure with current technology. The next step is to set up the Schrodinger equation (time-independent) for nuclei and electrons in a crystal.

Relative to the strong force, the electromagnetic force is 1/137th as strong, but the strong force acts over a range of approximately 1.0×10^{-5} angstrom. In the calculations carried out in this section, the atoms will never be arranged close enough for the strong force to be considered at all. The gravitational force, as with the electromagnetic force, acts over an infinite range. The electromagnetic force is more than 1.0×10^{36} times greater than the gravitational force, so it too can be neglected. Finally, the weak interacting force has a range of approximately 1.0×10^{-8} angstrom which, as with the strong force, is a range small enough that the weak force may be neglected.

The energy operators required are kinetic and electromagnetic potential.

$$\hat{H} = \hat{T}_e + \hat{T}_n + \hat{V}_{e-e} + \hat{V}_{e-n} + \hat{V}_{n-n} \quad (5.86)$$

The first two terms are the kinetic energy of the electrons and nuclei respectively.

$$\begin{aligned} \hat{T}_e &= -\sum_i \frac{\hbar^2}{2m} \nabla^2 \text{Sum of kinetic energy of electrons} \\ \hat{T}_n &= -\sum_k \frac{\hbar^2}{2M} \nabla^2 \text{Sum of kinetic energy of nuclei} \end{aligned} \quad (5.87)$$

The last three terms are potential energy terms due to the electromagnetic force:

$$\begin{aligned} V_{e-e} &= \frac{1}{2} \sum_{i,j,i \neq j} \frac{1}{|\vec{r}_i - \vec{r}_j|} \text{sum of potential energy between electrons} \\ V_{e-n} &= \sum_{i,k} \frac{z_i}{|\vec{r}_i - \vec{r}_l|} \text{sum of potential energy between electrons and nuclei} \\ V_{n-n} &= \frac{1}{2} \sum_{k,l,k \neq l} \frac{z_k z_l}{|\vec{r}_l - \vec{r}_k|} \text{sum of potential energy between nuclei} \end{aligned} \quad (5.88)$$

These operators are now input into the TISE.

$$\left[\left(-\sum_i \frac{\hbar^2}{2m} - \sum_k \frac{\hbar^2}{2M} \right) \nabla^2 + \sum_{i,k} \frac{z_i}{|\vec{r}_i - \vec{r}_l|} + \frac{1}{2} \left(\sum_{i,j,i \neq j} \frac{1}{|\vec{r}_i - \vec{r}_j|} + \sum_{k,l,k \neq l} \frac{z_k z_l}{|\vec{r}_l - \vec{r}_k|} \right) \right] |\Psi\rangle = E|\Psi\rangle \quad (5.89)$$

5.6.4 Born-Oppenheimer

The TISE arrived at is far too complicated to solve, even for the simplest of systems. It represents the many body system of electrons and nuclei, and the variables it takes are the positions of all nuclei and electrons.

$$\begin{aligned} \hat{H}|\Psi(\vec{r}_e, \vec{r}_n)\rangle &= E|\Psi(\vec{r}_e, \vec{r}_n)\rangle \\ \text{where } \vec{r}_e &= r_{e,1}, r_{e,1}, \dots, r_{e,i} \text{ (electron positions)} \\ \text{where } \vec{r}_n &= r_{n,1}, r_{n,1}, \dots, r_{n,i} \text{ (electron positions)} \end{aligned} \quad (5.90)$$

In 1927 the Born-Oppenheimer approximation was proposed to separate the electron components from the nuclei in the Hamiltonian.

Protons and Neutrons are almost 2,000 times the mass of electrons. With respect to the electrons, they move much slower and may be considered to be fixed or frozen in place. Simplifying for a moment to a single electron and proton, due to Newton's second law, we can see that the acceleration of the electron would be similarly 2,000 times that of the proton: $f_e = f_p$ and $m_e a_e = m_p a_p$ which leads to $a_e = \frac{m_p}{m_e} a_p$. As the nuclei move, the electrons are assumed to respond instantly, remaining in the ground state and not being promoted into higher energy levels.

The Hamiltonian of the electrons may be written with the electron co-ordinates as a variable, and the nuclei co-ordinates as a parameter.

$$\hat{H}_e(\vec{r}_e; \vec{r}_n) = \hat{T}_e(\vec{r}_e) + \hat{V}_{e-e}(\vec{r}_e) + \hat{V}_e - n(\vec{r}_e; \vec{r}_n) \quad (5.91)$$

The wavefunction and energy for the electrons may be calculated, although if the nuclear coordinates are changed, i.e. by changing the parameter r_n , the wavefunction and energy will need to be recalculated.

$$\hat{H}_e(\vec{r}_e; \vec{r}_n) \psi_e(\vec{r}_e; \vec{r}_n) = E_e(\vec{r}_n) \psi_e(\vec{r}_e; \vec{r}_n) \quad (5.92)$$

$$\Psi(\vec{r}_e, \vec{r}_n) = \chi_{ne}(\vec{r}_e) \psi_e(\vec{r}_e; \vec{r}_n) \quad (5.93)$$

The electronic Hamiltonian is still dependent on the electronic co-ordinates. As there are three co-ordinates per electron, there are 3 dimensions when solving for a hydrogen atom. For an Iron atom, with 26 electrons, there are 78 dimensions, and for a 4x4x4 FCC supercell of Iron there would be 256 atoms, 26 electrons per atom and 3 dimensions per electron, giving a total of 19,968 dimensions. Even for small numbers of electrons, solving this equation is impractical.

5.6.5 Crystals, Reciprocal Space and Bloch Theorem

A Bravais lattice is a construct used to describe a periodic crystal lattice. It has the following properties:

$$\begin{aligned} \vec{R} &= n_1 \vec{a}_1 + n_2 \vec{a}_2 + n_3 \vec{a}_3 \\ n_1, n_2, n_3 &\in \mathbb{Z} \\ \vec{a}_1, \vec{a}_2, \vec{a}_3 &\text{are independent} \end{aligned} \quad (5.94)$$

There are 14 Bravais lattices and 7 families of these lattices. This work

Bloch Theorem

Metals are made up from grains which in turn are crystal lattices of atoms. Despite the majority of metals being composed of a large collection of microscopic crystals, rather than being a single perfect crystal, most of their properties may be calculated as if the metal were a single crystal.

The grain size of water quenched SS304 is approximately 30 micrometers across[78]. If the crystal were a cube, it would contain 2.0×10^{15} atoms and it would have sides over 100,000 atoms in length.

Table 5.4: Bravais Lattices

Class	Lengths	Angles
Cubic	$a = b = c$	$\alpha = \beta = \gamma = 90$
Hexagonal	$a = b, c$	$\alpha = \beta, \gamma = 120$
Rhombohedral	$a = b = c$	$\alpha = \beta = \gamma \neq 90$
Tetragonal	$a = b, c$	$\alpha = \beta = \gamma = 90$
Orthorhombic	a, b, c	$\alpha = \beta = \gamma = 90$
Monoclinic	a, b, c	$\alpha = \beta = 90, \gamma \neq 90$
Triclinic	a, b, c	$\alpha, \beta, \gamma,$

In solid state physics, in order to solve the TISE for such a crystal, it is useful to consider an infinite sized crystal. It may be useful to visualise this as a "ring" of atoms in 1D5.10, or as a super-cell with periodic boundary conditions in 3D.

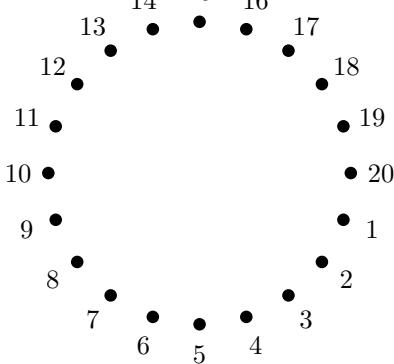


Figure 5.10: A useful, albeit incorrect, way of visualising an "infinite" chain of atoms in 1D

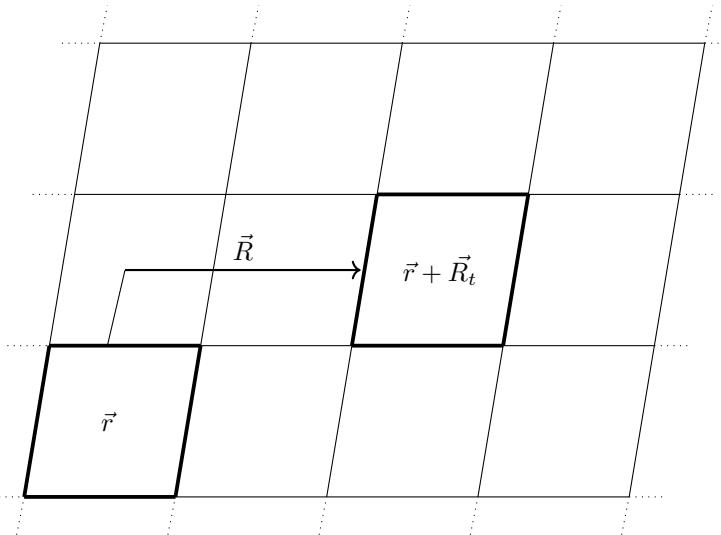


Figure 5.11: A 2D example of a translation by an integer multiple of the unit cell from the origin unit cell

As the structure is a repeating lattice, any point within a unit cell is equivalent to any other point translated by an integer multiple of the lattice vector 5.11.

$$\psi(\vec{r}) = \psi(\vec{r} + \vec{R}_t) \quad (5.95)$$

where $\vec{R}_t = n_1 \vec{R}_1 + n_2 \vec{R}_2 + n_3 \vec{R}_3$ where \vec{R} is the real lattice vector

Reciprocal space, also known as k-space or momentum space, is a mathematical construct. It is an imaginary space where the lengths, and volumes, are the inverse of real space and planes of atoms are represented as points.

Starting with a lattice of points in real space \vec{R} , points in reciprocal space \vec{G} only are valid points if $\exp(i\vec{G} \cdot \vec{R}) = 1$ [79]. A real space vector \vec{R} is transformed to its reciprocal in the following way, where Ω is the volume of the primitive cell in real space.

$$\begin{aligned}
\vec{g}_1 &= 2\pi/\Omega \vec{r}_2 \times \vec{r}_3 \\
\vec{g}_2 &= 2\pi/\Omega \vec{r}_3 \times \vec{r}_1 \\
\vec{g}_3 &= 2\pi/\Omega \vec{r}_1 \times \vec{r}_2 \\
\Omega &= \vec{r}_1 \cdot (\vec{r}_2 \times \vec{r}_3)
\end{aligned} \tag{5.96}$$

If the crystal structure is modelled as an infinitely repeating lattice, the potential also has the same periodicity.

$$V(\vec{r}) = V(\vec{r} + \vec{R}_t) \tag{5.97}$$

As the potential is a periodic function, it may be written as a fourier series in reciprocal space.

$$V(\vec{r}) = \sum_{\vec{G}} V_{\vec{G}} \exp(i\vec{G}\vec{r}) \tag{5.98}$$

The wave function for an electron in the lattice is also periodic and may be written as the product of a plane wave.

$$\psi_k(\vec{r}) = u_{\vec{k}}(\vec{r}) \exp(i\vec{k}\vec{r}) \tag{5.99}$$

$$\begin{aligned}
u_{\vec{k}}(\vec{r}) &= u_{\vec{k}}(\vec{r} + \vec{r} + \vec{R}_t) \\
\psi_k(\vec{r}) &= \psi_k(\vec{r}) \exp(i\vec{k}\vec{R}_t)
\end{aligned} \tag{5.100}$$

The Born-von Karman boundary conditions apply to the wave function such that:

$$\begin{aligned}
\psi_k(\vec{r} + \vec{R}_t) &= \psi_k(\vec{r}) \\
\text{where } \vec{R}_t &= \sum_i N_i \vec{a}_i \text{ and } N_i \in \mathbb{Z}
\end{aligned} \tag{5.101}$$

With these boundary conditions and use of plane waves, Bloch theorem replaces an enormous number of electrons with an infinite number electrons in a periodic lattice where only those in the unit cell need to be considered.

The (first) Brillouin Zone is a volume in (3D) reciprocal space and it is analogous to the Wigner-Seitz cell in real space. It is the primitive lattice cell in reciprocal space. In real space, any point anywhere in the lattice is a integer multiple of the lattice vector away from the same point in the origin Wigner-Seitz cell. There are an infinite number of wave vectors \mathbf{k} in reciprocal space, but by virtue of the periodicity of the lattice, they are also all found within the Brillouin Zone.

Body Centered Cubic

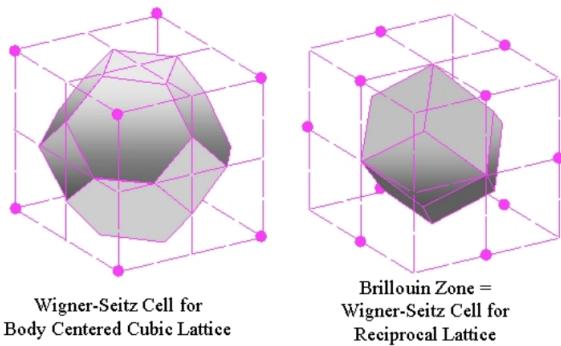


Figure 5.12: BCC Wigner Seitz Cell and BZ [80]

Face Centered Cubic

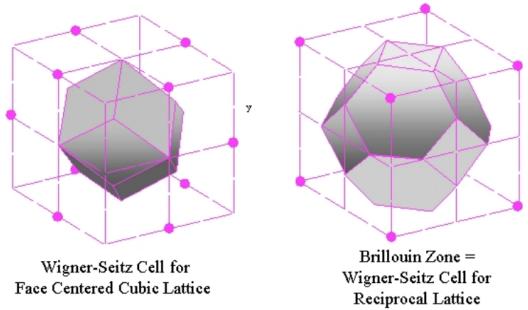


Figure 5.13: FCC Wigner Seitz Cell and BZ [80]

5.6.6 Hohenberg-Kohn Theorem

In the 1960s, Hohenberg and Kohn simplified the problem in an exact way by proving

- in an external potential $v(\vec{r})$, the potential is uniquely determined by the density of the ground state $n_0(\vec{r})$ assuming that the particles are non-degenerate
- a uniquely defined functional $E[\rho(\vec{r})]$ exists and the ground state energy, $\min(E[\rho(\vec{r})])$, is found by varying the density

The proof starts by picturing a box of electrons that interact with each other through coulomb repulsion within an external potential $\vec{v}(r)$, for example the potential of "fixed" nuclei following the Born Oppenheimer approximation.

$$\begin{aligned} \hat{H} &= \hat{T} + \hat{V} = \frac{1}{2}\nabla^2 + V(\vec{r}) \\ \left[-\frac{1}{2}\nabla^2 \right] \Psi(\vec{r})_0 &= E_0 \Psi(\vec{r}) \end{aligned} \quad (5.102)$$

$$\hat{H} = \hat{T} + \hat{V} + \hat{U} \quad (5.103)$$

where

$$T = \frac{1}{2} \int \nabla \psi^*(\vec{r}) \nabla \psi(\vec{r}) d\vec{r} \quad (5.104)$$

$$V = \int v(\vec{r}) \psi^*(\vec{r}) \psi(\vec{r}) d\vec{r} \quad (5.105)$$

$$U = \frac{1}{|\vec{r} - \vec{r}'|} \psi^*(\vec{r}) \psi^*(\vec{r}') \psi(\vec{r}) \psi(\vec{r}') d\vec{r} d\vec{r}' \quad (5.106)$$

Variational method shows that, given a system $\hat{H}|\Psi_n\rangle = E_n|\Psi_n\rangle$, the expectation value of \hat{H} for an arbitrary state $|\Psi_a\rangle$ must satisfy:

$$\hat{H}\Psi_n = E_n\Psi_n \quad (5.107)$$

Take the inner product then rearrange.

$$\langle\Psi_n|\hat{H}|\Psi_n\rangle = E_n\langle\Psi|\Psi\rangle \quad (5.108)$$

$$\langle\hat{H}\rangle = \frac{\langle\Psi_n|\hat{H}|\Psi_n\rangle}{\langle\Psi|\Psi\rangle} \geq E_n \quad (5.109)$$

Minimise to find the ground state energy.

Now assume that a second, different, external potential exists with the ground state Ψ' and the same density $n(\vec{r})$. $\Psi \neq \Psi'$

As a result of the Hohenberg-Kohn theorem, we are able to calculate the electronic energy of a system from the charge density.

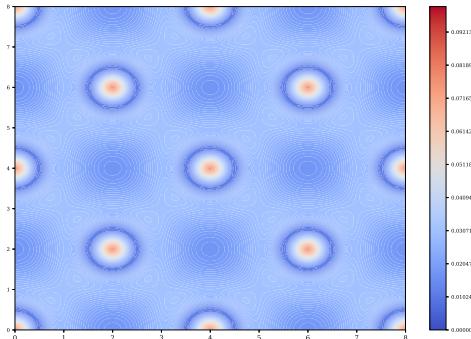


Figure 5.14: Charge density FCC aluminium xy plane at $z = 0.0a_0$

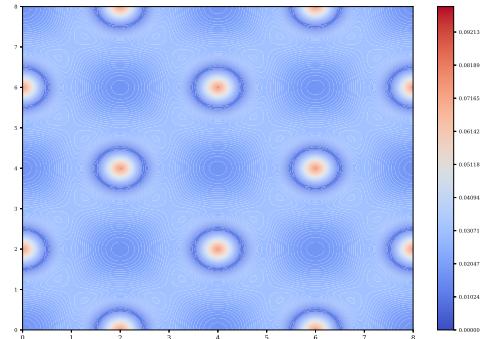


Figure 5.15: Charge density FCC aluminium xy plane at $z = 0.5a_0$

5.6.7 Kohn-Sham Equations

The year following the Hohenberg-Kohn theorem, a set of self-consistent equations were derived by Kohn and Sham. The ground state energy of interacting jellium in the potential of fixed nuclei, an external potential, can be written as follows[81]:

$$E = T_s + U + V_{n-e} + E_{xc} \quad (5.110)$$

$$E = T_s[\rho(\vec{r})] + \int d\vec{r}v(\vec{r})\rho(\vec{r}) + \frac{1}{2} \int \int d\vec{r}d\vec{r}' \frac{\rho(\vec{r})\rho(\vec{r}')}{|\vec{r} - \vec{r}'|} + E_{xc}[\rho(\vec{r})] \quad (5.111)$$

The kinetic energy, T_s , is that of a system of non interacting and E_{xc} is the exchange and correlation of an interacting system. The exchange and correlation energy functional exists, but is unknown.

The Kohn-Sham equations are used to calculate the energy of a system. The Schrodinger equation is not for all the atoms in the system; it's a one electron equation

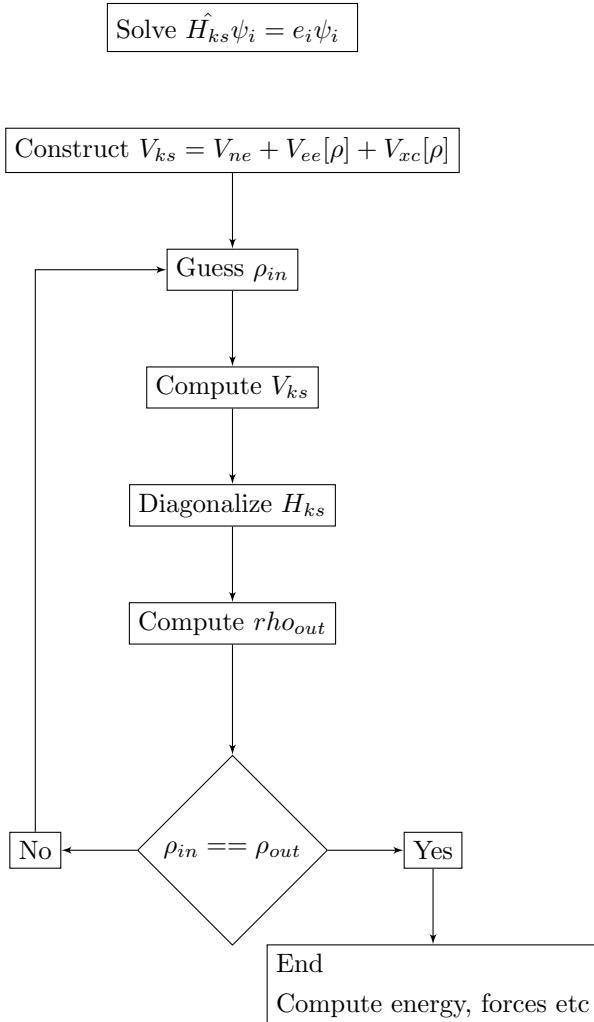
$$\hat{H}_{KS}\psi_i = E_i\psi_i \quad (5.112)$$

$$(-\frac{1}{2}\nabla^2 + \hat{v}_{KS})\psi_i = E_i\psi_i \quad (5.113)$$

$$\hat{v}_{KS}(\vec{r}_e, \vec{r}_n) = v_{n-e}(\vec{r}_e, \vec{r}_n) + \int d^3\vec{r}' \frac{\rho(\vec{r}')}{|\vec{r}_e - \vec{r}'|} + v_{xc}[\rho](\vec{r}_e) \quad (5.114)$$

5.6.8 Self-Consistent Solution

The Kohn-Sham equations cannot be solved traditionally, as they require the density, and the density is calculated by solving the equation. It can be solved self consistently: an initial density is guessed, and this is repeatedly updated until the density in and out values are the same (or within a set convergence threshold). The basic algorithm used by PWscf from the Quantum Espresso suite is shown below[82].



5.6.9 Exchange-Correlation Energy

The Kohn-Sham equations have an exchange-correlation energy and this functional is used to collect together the electron energy not captured in the non-interacting energy functionals.

The Pauli exclusion principle states that two fermions, half integer spin particles, cannot occupy the same quantum state. This is why electrons occupy a unique orbital within an atom defined by the quantum numbers n, l, m_l and m_s . Consider two particles at points \vec{r}_a and \vec{r}_b ; the probability amplitude of the wavefunction of these particles equals 1.

$$|\Psi(\vec{r}_a, \vec{r}_b)|^2 = 1 \quad (5.115)$$

Exchanging the particles must also give the same result; they still must exist somewhere with probability 1.

$$|\Psi(\vec{r}_b, \vec{r}_a)|^2 = 1 \quad (5.116)$$

Bosons, integer spin particles, are symmetric when exchanged:

$$\Psi(\vec{r}_a, \vec{r}_b) = \Psi(\vec{r}_b, \vec{r}_a) \quad (5.117)$$

Fermions, on the other hand, are antisymmetric:

$$\Psi(\vec{r}_a, \vec{r}_b) = -\Psi(\vec{r}_b, \vec{r}_a) \quad (5.118)$$

By setting up a wavefunction for two Fermions, and exchanging them, it can be seen why they cannot exist in the same state.

$$\Psi_{ab} = \psi_1(\vec{r}_a, \vec{r}_b) - \psi_2(\vec{r}_b, \vec{r}_a) \quad (5.119)$$

$$\Psi_{ab} = \psi_1(\vec{r}_a, \vec{r}_b) - \psi_2(\vec{r}_a, \vec{r}_b) = 0 \quad (5.120)$$

The XC term combines the difference between the real system and the fictitious non-interacting system set out in the Kohn-Sham equations. It also includes the difference between the quantum mechanical electron-electron repulsion and classical electron-electron repulsion. Unfortunately, whilst we know a functional exists, we do not know the exact form of it.

5.6.10 Pseudopotentials

Plane wave basis sets are used to help solve the TISE. The DFT code used in this work is Quantum Espresso, and the binary that carries out the calculations is named PWscf: plane wave self consistent field.

Where \vec{G} is the reciprocal lattice vector, a summation of plane waves may be used to construct each electronic wave function^{5.121}[83].

$$\Psi_{\vec{k},n} = \sum_{\vec{G}} | \vec{G} | < G_{max} c_{\vec{k} + \vec{G},n} \exp(i(\vec{k} + \vec{G}) \cdot \vec{r}) \quad (5.121)$$

Unfortunately, the plane-wave basis sets required are too large to be used in practice, as they would need to represent the tight, rapidly oscillating inner orbitals as well as the valence electrons.

Bonding and material properties are primarily determined by valence electrons, not core electrons. Iron, for example, has two valence electrons in the 4s shell; however, it is a transition metal and the partially empty 3d shell is also important to consider. The core electrons do not contribute as much to the bond and the model may be simplified using pseudopotentials.

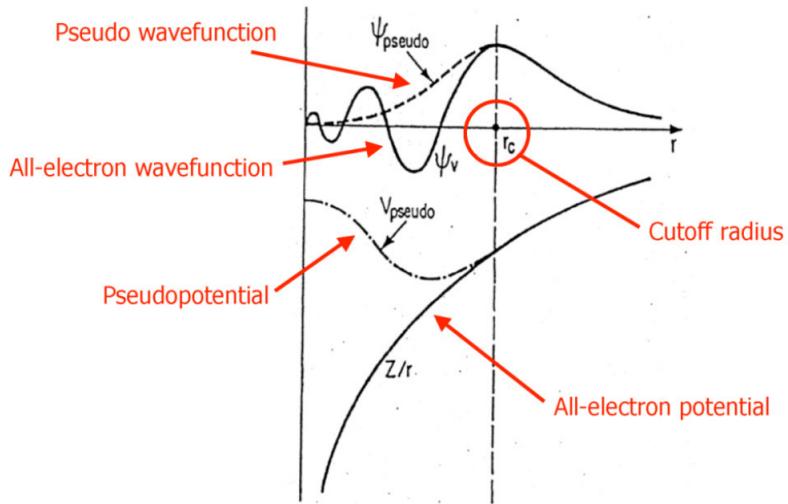


Figure 5.16: Replacing the complex potentials with a pseudopotential[84]

The eigen energies of core electrons are also much larger than properties we see in materials, such as the cohesive energies and this also raises a concern that including the core electrons may introduce errors that are on a similar scale to the energies we are calculating.

The rapidly oscillating function in the core is replaced by a smoother pseudopotential. The same material properties are calculated as the valence electron functions are preferred, but a much smaller basis set is required to do so.

There are a wide range of pseudopotentials available to use in DFT calculations, and one major distinguishing feature is how they treat the XC potential $v_{xc}([n]; \vec{r})$.

$$\left(\frac{1}{2} + V_{ext}(\vec{r}) + V_H(\vec{r}) + V_{xc}(\vec{r}) \right) \psi_{k,\sigma}(\vec{r}) = \epsilon_{k,\sigma} \psi_{k,\sigma}(\vec{r}) \quad (5.122)$$

The KS equation consists of several potentials including the Exchange-Correlation energy. This term represents the many-electron effects. It does exist as a universal functional[85], but the exact form of this functional is unknown.

LDA

The Local Density Approximation replaces the electron density with jellium, a homogeneous electron gas. Kohn and Sham, in their original 1965 work, introduced jellium as the density model and, despite its simplicity in comparison to the real electron density distribution, it has worked well for simple metals.

$$E_{xc}^{LDA}[\rho] = \int d^3\vec{r}\rho(\vec{r})[e_x(\rho(\vec{r})) + e_c(\rho(\vec{r}))] \quad (5.123)$$

LSDA

Electrons are fermions and have half integer spin. When trapped in a potential, such as an atom or a crystal lattice, electrons must take difference quantum states and one of the parameters is the spin of the electron. The LDA does not treat spin at all, but the LSDA splits the density into up and down spin (eq. ??[85]).

$$\begin{aligned} \rho_{\uparrow}(\vec{r}) &= \sum_k^{occ} |\phi_{k,\uparrow}(\vec{r})|^2 \\ \rho_{\downarrow}(\vec{r}) &= \sum_k^{occ} |\phi_{k,\downarrow}(\vec{r})|^2 \\ \rho(\vec{r}) &= \rho_{\uparrow}(\vec{r}) + \rho_{\downarrow}(\vec{r}) \end{aligned} \quad (5.124)$$

The XC energy is now dependent on the spin up and spin down density as well as the position (eq. ??[85]).

$$\begin{aligned} E_{xc}^{LSDA}[\rho_{\uparrow}, \rho_{\downarrow}] &= \int d^3\vec{r}\rho(\vec{r})[e_x(\rho(\vec{r}))f(\zeta(\vec{r})) + e_c(\rho(\vec{r}), \zeta(\vec{r}))] \\ \zeta &= \frac{\rho_{\uparrow} - \rho_{\downarrow}}{\rho_{\uparrow} + \rho_{\downarrow}} \\ f(\zeta) &= \frac{1}{2} \left((1 + \zeta)^{4/3} + (1 - \zeta)^{4/3} \right) \end{aligned} \quad (5.125)$$

The energy of the system now being solved is also dependent upon the charge density and spin polarization (eq ??).

$$E = T[\rho, \zeta] + E_{ext}[\rho] + \frac{1}{2}E_{ee}[\rho] + E_{xc}[\rho, \zeta] \quad (5.126)$$

GGA

LSDA vs GGA Potentials

Typically, GGA is an improvement over LSDA, with improvements in the total energies and structural energy differences[86]. Another short coming of LSDA is that it predicts FCC to be the optimal structure for pure iron at 0K, which is incorrect[87].

In the case of the compound Titanium Disilicide, $TiSi_2$, GGA is in better agreement, overall, with experiment than LDA, although several values are better predicted by the latter.

Parameter	Experimental	LDA	GGA
a/angs	8.27	8.08	8.21
b/angs	4.80	4.74	4.81
c/angs	8.55	8.53	8.64
B_0 (Reuss)/GPa	146.8	156.9	141.9
B_0 (Voigt)/GPa	150.9	159.1	145.0
C_{11} /GPa	317.5	377.2	326.0
C_{22} /GPa	320.4	341.1	298.4
C_{33} /GPa	413.2	425.3	371.9
C_{44} /GPa	112.5	136.5	123.5
C_{55} /GPa	75.8	93.7	85.3
C_{66} /GPa	117.5	154.6	135.9
C_{12} /GPa	29.3	27.8	22.4
C_{13} /GPa	38.4	21.3	26.5
C_{23} /GPa	86.0	95.1	105.5

Table 5.5: Experimental vs LSDA vs GGA for $TiSi_2$ [64]

As shown in table 5.5, the data shows that LDA is within 1.26% of the experimental lattice parameters, 6.16% of the experimental bulk modulus and 18.3% the value of the experimental elastic constants; several of these were particularly poor, with an almost 45% disagreement. On the other hand, GGA is within 0.66% of the experimental lattice parameters, 3.62% of the experimental bulk modulus and 14.97% the value of the experimental elastic constants[64].

Lanthanum aluminate ($LaAlO_3$) has also been studied using DFT, but this material is better modelled (to reproduce lattice parameters and bulk modulus) with either LDA or a specific form of GGA, the PBESOL potential, revised to better reproduce lattice parameters for solids, as shown in table 5.6.

Parameter	Experimental	LDA	GGA	GGA (PBESOL)
a/angs	3.78	3.74	3.82	3.77
B_0 /GPa	215	224.00	190.35	206.41

Table 5.6: Experimental vs LSDA vs GGA vs GGA PBESOL for $LaAlO_3$ [88]

A complete library of pseudopotentials is available through the Quantum Espresso website. Several of these have been tested with the PWscf code to calculate the lattice parameter and bulk modulus of simple metals including Aluminium and Sodium.

	Element	a/angs	B_0 /GPa
Experimental	Na	4.29[89]	6.3[89]
Na.pz-spn-kjpaw-psl.1.0.0	Na	4.06	8.72
Na.pbe-spn-kjpaw-psl.1.0.0	Na	4.20	7.67
Na.pbessol-spn-kjpaw-psl.1.0.0	Na	4.17	7.50
Experimental	Al	4.05[90]	76[90]
Al.pz-nl-kjpaw-psl.1.0.0	Al	3.98	78.6
Al.pz-n-kjpaw-psl.1.0.0	Al	3.98	78.6
Al.pbe-nl-kjpaw-psl.1.0.0	Al	4.04	91.3
Al.pbe-n-kjpaw-psl.1.0.0	Al	4.04	75.0
Al.pbessol-nl-kjpaw-psl.1.0.0	Al	4.01	87.7
Al.pbessol-n-kjpaw-psl.1.0.0	Al	4.01	79.0

Table 5.7: Experimental vs LSDA vs GGA - the DFT values were computed with a PWscf[91] using a 2x2x2 cell, 7x7x7 kpoints and $\text{ecutwfc}=50$

As can be seen from table 5.7, the PBE and PBESOL GGA type potentials are in general better at reproducing the lattice parameters. However, for Aluminium, the PZ LSDA type potential is in better agreement when

calculating the bulk modulus. Overall, the GGA type potentials with the total self-consistent potential pseudized (these are the pseudopotential files that contain -n- in the name, rather than -nl-) perform best for these simple metals.

Whilst DFT is an exact method for solving the TISE, there are a number of approximations that still need to be made in addition to there being gaps in our knowledge (i.e. a lack of an exact functional for the XC).

5.6.11 Brillouin Zone, Smearing and K-Points

In a real lattice, the Wigner-Seitz cell

The Brillouin zone is integrated over to calculate the electronic density $\rho(\vec{r})$

$$\rho(\vec{r}) = \frac{1}{\Omega_b z} \int_{bz} d\vec{k} \approx \sum_{\vec{k}} \omega_{\vec{k}} \quad (5.127)$$

5.6.12 Smearing

Electrons are fermions, half integer spin particle, and when bound two cannot have the same state. At absolute zero, the electrons fill from the lowest available state, filling each increasing state up to the Fermi level.

The Fermi energy is the energy of the Fermi level, and it is used in Fermi-Dirac statics to calculate the probability that a fermion has a specified energy.

$$P(E) = \frac{1}{\exp(E - E_f)/kT + 1} \quad (5.128)$$

- Gaussian
- Fermi-Dirac
- Methfessel-Paxton
- Marzari-Vanderbilt

5.7 DFT: Magnetism

5.7.1 Collinear

5.7.2 Non-Collinear

It is simpler to simulate either antiferromagnetic or ferromagnetic configurations where the Modern Nuclear Data Evaluation With The TALYS Code System

5.8 Collisions and Interatomic Potentials

5.9 Classical Molecular Dynamics

5.9.1 Introduction

Ab initio (DFT) calculations approximately solve the schrodinger equation to calculate the energy, forces and stress of a given simulation. They take a relatively long time to calculate, have relatively small numbers of atoms (hundreds to thousands) and are fixed at one moment in time, although there are now programs that will run DFT MD. In comparison, Molecular Dynamics model a collection of atoms over a specified period of time. Much larger collections of atoms are possible (thousands to millions) and the interaction between atoms are predefined by interatomic potentials.

Molecular Dynamics has been used to investigate the effects of radiation on materials. The damage event on the atomic scale is rapid in time, in the femtosecond to picosecond range, and affects a small volume of the material whereas the effects of the damage to the material on a mesoscopic and macroscopic scale may range from days to decades. In reactor pressure vessels, for instance, the damage and defect production will cover a large range of time and size scales[19]:

- $10^{-10}m$ to $10^{-3}m$ (10^{18} to 10^{25} atoms)
- $10^{-15}s$ to 10^9s (femtoseconds to 30 years)

As a major element of steel, iron has often been the subject of MD simulations. Damage cascades have been studied in iron at a range of temperatures and PKA energies.

5.9.2 Neighbour List

The simple neighbour list creation has already been discussed earlier in this chapter, but many modern MD codes are designed to run on computers with many thousands of processor cores. It would be in-efficient for the neighbour list to be computed in series, so it is decomposed into many domains that are

5.9.3 Verlet Timestep

- Build a neighbourlist
- Calculate the force on each atom
- Start loop with a time step Δt
- Calculate the half time step velocity $\vec{v}(t + 0.5\Delta t) = \vec{v}(t) + 0.5\Delta t \times a(t)$
- Using the half time step velocity, move the atoms to their new position
- Update or rebuild the neighbour list
- Recalculate the forces (and acceleration) at time $t + \Delta t$
- Calculate the end velocity
- ...

5.9.4 Molecular Dynamics Codes

5.9.5 Selecting a Time Scale

Take, for example, a 50x50x50 BCC Iron supercell with a lattice parameter of 2.87 angstrom. The supercell measures almost 15nm on each side. Simulating a thermal neutron, travelling at 2,200 m/s, would require an overall time period of at least 7 picosecond to capture the neutron passing through the supercell. Higher energy particles would require smaller overall time periods, divided up into as many time steps as the user requires.

If PKA damage is being modelled, there may be a point where the depth of the damage cascade becomes larger than the supercell itself. Damage cascades of 500KeV iron atoms into an iron target travel to a depth of up to 300nm, which would require a simulation supercell at least 1,000 BCC cells deep. Kinetic Monte Carlo would be better suited at simulating higher energy damage cascades.

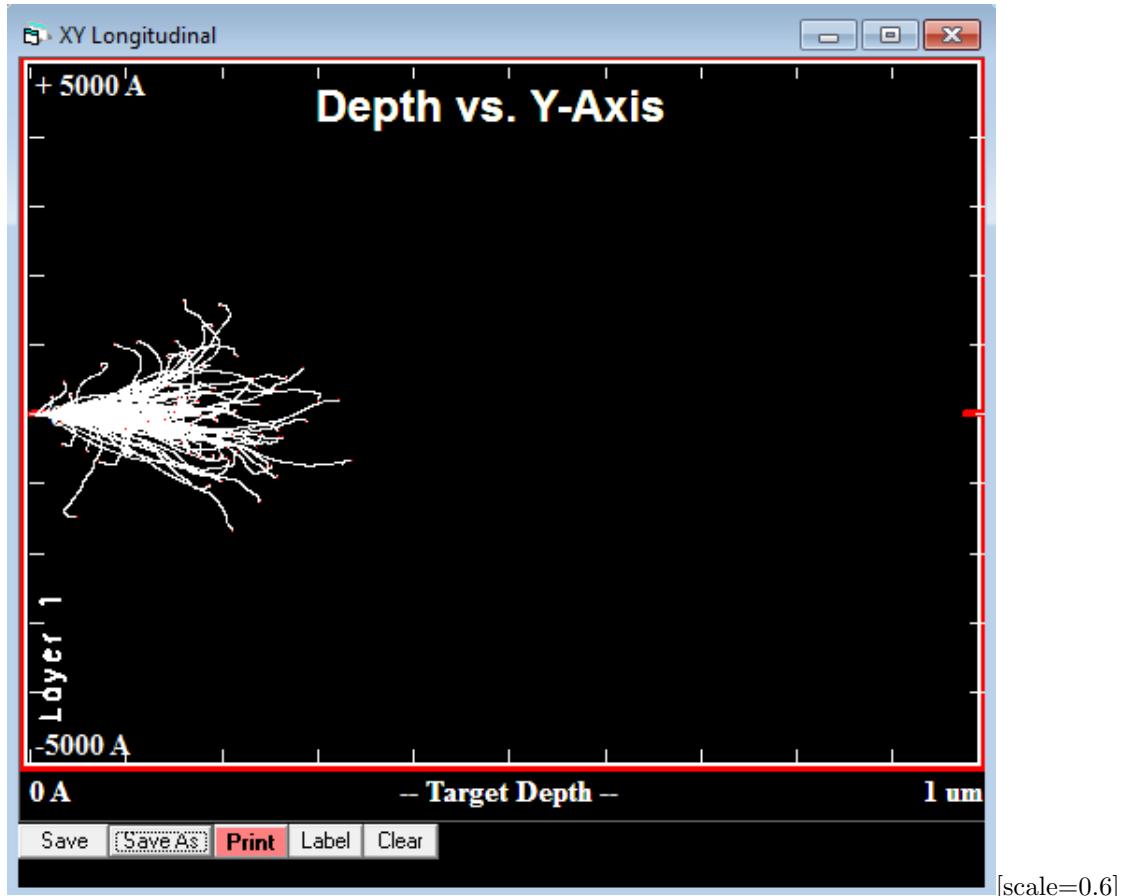


Figure 5.17: Damage cascade: 500kev iron projectiles into an iron target calculated with SRIM

5.10 Optimization

5.10.1 Introduction

Optimization is the process of finding the best solution for a problem that may also need to satisfy a number of imposed constraints. In terms of this work, there are several points in which optimization plays a key role, whether it be in the DFT and MD codes use or the fitting codes developed here.

The mathematical description of optimisation, as written in Numerical Optimization by Nocedal and Wright, is as follows:

Equations here

This work relies on optimization in several stages throughout. The DFT code used (PWScf) employs the Broyden-Fletcher-Goldfarb-Shanno algorithm to relax structures to give the optimum volume and optimal structure of an arrangement of atoms. During the fitting of the variety options of equation of state to the energy-volume points, the Levenberg-Marquart Algorithm (LMA) is used. Finally, a genetic algorithm and simulated annealing algorithm are coupled with LMA to locate the global optimal parameters for Interatomic Potential functions.

5.10.2 Continuous and Discrete Optimization

Problems that may be solved using discrete optimisation have, as the name suggests, discrete solutions. An example would be the traveling salesman problem, where the aim is to minimise the distance travelled by the salesman between cities. An example of the solution may be city E to A to B to C to D. There's a set of possible solutions to the problem. If the problem was changed such that any point within a 1 mile radius of the center of each city is a valid solution, and the aim is to find the optimal points to travel between within the area of each city, it would become a continuous problem.

Fitting interatomic potentials to data is an example of continuous optimisation. The Morse potential takes three parameters plus the separation r between the two atoms. If the parameters are being optimised such that Morse potential matches the experimental forces (assuming they could be measured experimentally) between two atoms for a set of separation values, the choice of parameters isn't discrete; they are taken from a continuous range of reals.

5.10.3 Constrained and Unconstrained Optimization

There are constraints that may need to be considered, and constraints that apply to the fitting of interatomic potentials are as follows:

- continuous well behaved function
- continuous first-derivative, smooth change in force wrt r
- positive electron density
- smooth cutoff at some $r = r_{cut}$
- repulsion (ZBL/exponential) as r approaches 0
- continuous second-derivative
- enough parameters coupled with a good choice of functions so the potential is flexible enough to replicate all the experimental data well
- a small enough range of parameters to keep the parameter space small enough to search in a reasonable amount of time
- maintain physical elegance
- ignore physical elegance

Certain constraints listed will battle one-another, while others may be dropped, such as restricting to positive electron-densities if the choice to ignore physical elegance is selected.

5.10.4 Global and Local Optimization

An example to discuss the difference between local and global optimization will be given in a very literal sense. I am located in Birmingham, and I wish to find the highest point local to me. Using a modern smartphone and map I could quickly find the nearest hill to my current position; this would be the local maxima. I would, however, need to go up and down many peaks and troughs, scouring the entire surface of the Earth, until I found the global maxima, somewhere amongst the many peaks and troughs in the Himalayas.

When optimising the parameters of a function, it is relatively easy to find a local extreme that would give me the optimum parameters locally. It is a much harder task to find the global optimum, especially if very little, or nothing at all, is known about the function for which the parameters are being optimised.

5.10.5 Global Optimization Algorithms

Simulated Annealing

A large amount of computing power is required to problems in Physics. The optimum arrangement of Iron atoms, at various concentrations and temperatures is one such example, but nature "calculates" the correct structure and reacts accordingly as the conditions change. It makes perfect sense to look to nature when designing optimisation algorithms.

Simulated annealing mimics the way a solid cools, with its atoms settling into the optimum relaxed positions as time passes. Initially, the solid is hot and the atoms are able to take non-optimal positions, but as the solid cools the atoms take their optimal places. The heating allows the atoms to move from non-optimal positions in the initial configuration.

The simulated annealing algorithm takes in a starting set of parameters. These are varied and, if the new parameters give a better solution to the problem, the parameters are updated. However, there is a chance that a worse set of parameters are used, and this depends on how bad they are and the current value of the temperature in the algorithm. By allowing a worse set of parameters to be used, it gives the solution the chance to jump out of a local minimum and explore other solutions that it would otherwise be oblivious to.

Listing 5.3: Simple simulated annealing subroutine

```
1 // Simulated Annealing
2 subroutine simulated_annealing(f, p, pv, x, y, t, p_best)
3   // f -      (IN) the function f(p, x) for which the parameters are being optimised
4   // p -      (IN) array containing parameter/parameters
5   // pv -     (IN) array of maximum parameter/s variance
6   // x -      (IN) array of x points
7   // y -      (IN) array o fy points
8   // t -      (IN) starting temperature
9   // p_best - (OUT) optimised parameters
10  //
11
12  rss = (f(x[:,], p[:]) - y[:])**2
13
14  rss_best = rss
15  p_best[:] = p[:]
16
17  // These may be modified as required
18  outer_loops = 100
19  inner_loops = 1000
20  t_decrease = 0.99
21  pv_decrease = 0.99
22
23  for outer = 1, outer_loops
```

```

24     for inner = 1, inner_loops
25
26         r[:] = 0.5 - random_float(0.0, 1.0) // array of random floats (same size as p)
27
28         pt[:] = p[:] + r[:] * pv[:]
29         rss = (f(x[:], pt[:]) - y[:])**2
30
31         // If better, always use new parameters
32         if (rss < rss_best) then
33             rss_best = rss
34             p[:] = pt[:]
35             p_best[:] = pt[:]
36
37         // If worse, sometimes use parameters (based on how good they are, and the temperature)
38         else
39             // as t -> 0 exp((rss_best - rss)/t) -> 0
40             // as (rss_best - rss) -> 0 exp((rss_best - rss)/t) -> 1 (where t = 1)
41             rn = random_float(0.0, 1.0)
42             if (rn <= exp((rss_best - rss) / t)) then
43                 p[:] = pt[:]
44             end if
45         end if
46
47     end for
48
49     // Decrease temperature
50     t = t_decrease * t
51
52     // Decrease variance
53     pv[:] = pv_decrease * pv[:]
54
55 end for
56 end subroutine

```

Genetic Algorithm

It is unclear whether life started on this planet, or was brought to this planet, and this idea only removes the problem of how life as we are able to understand it was conceived to a different time and place. However it started, there is evidence of life on this planet approximately 3.5 billion years ago. Prokaryotes were the simple single cell organisms that inhabited our planet for billions of these. Their cell does not have a nucleus and the genetic information is contained in RNA and DNA within the cell.

Approximately 1.5 billion years ago, life made a leap forward in complexity and evolved into the first Eukaryotic cells. These are larger and more complex than Prokaryotes, and the genetic material is stored within a nucleus.

Through processes of inheritance, variation, natural selection and the vast expanse of time, more complex organisms developed.

As simulated annealing takes inspiration from cooling solids, genetic algorithms take inspiration from evolution.

Listing 5.4: Simple simulated annealing subroutine

```

1 // Simulated Annealing
2 subroutine simulated_annealing(f, p, pv, x, y, t, p_best)
3 // f -      (IN) the function f(p, x) for which the parameters are being optimised
4 // p -      (IN) array containing parameter/parameters
5 // pv -     (IN) array of maximum parameter/s variance
6 // x -      (IN) array of x points
7 // y -      (IN) array o fy points
8 // t -      (IN) starting temperature

```

```

9 // p_best - (OUT) optimised parameters
10 //
11
12 // Set important variables
13 pop_size = 32
14 fresh_size = 8
15 generations = 100
16 chance_to_mutate = 0.1
17 extinction_chance = 0.1
18
19 // Create population array
20 pop = make_pop(pop_size, p, pv) // Some function that initialises the population
21
22 // Calc rss for each parameter set in the pop array
23 for n=1,pop_size
24     pop_rss[n] = sum((y[:] - f(pop[n,:]-x[:]))**2)
25 end for
26
27 // Loop through generations
28 for n=1, generations
29
30     // Breed and replace if improvements
31     key = shuffled_list(1,generations) // Make a shuffled list of integers, unique, 1 to generations
32     for m=1, pop_size//2
33         parent_a = pop[key[2 * m - 1]]
34         parent_b = pop[key[2 * m]]
35         child_a, child_b = breed(parent_a, parent_b, chance_to_mutate) // Breed, possibly mutate
36
37         child_a_rss = sum((y[:] - f(child_a[:] - x[:]))**2)
38         child_b_rss = sum((y[:] - f(child_b[:] - x[:]))**2)
39
40         no_clones(pop, child_a, child_b) // if either already exists in pop, mutate to avoid clones
41         replace(pop) // some subroutine to replace parent/s with child parameters if better
42
43         update(p_best) // some subroutine to update p_best
44     end for
45
46     // Fresh solutions
47     fresh = make_pop(fresh_size, p, pv)
48     key = shuffled_list(1,generations) // Make a shuffled list of integers, unique, 1 to generations
49
50     // Breed with fresh parameters
51     for m=1, fresh_size
52         parent_a = pop[key[m]]
53         parent_b = fresh[m]
54
55         child_a, child_b = breed(parent_a, parent_b) // Breed
56
57         child_a_rss = sum((y[:] - f(child_a[:] - x[:]))**2)
58         child_b_rss = sum((y[:] - f(child_b[:] - x[:]))**2)
59
60         no_clones(pop, child_a, child_b) // if either already exists in pop, mutate to avoid clones
61         replace(pop) // some subroutine to replace parent with best child
62
63         update(p_best) // some subroutine to update p_best
64
65     end for
66
67     // Chance of extinction event to kill off bad parameters
68     extinction_event(pop, extinction_chance) // remove worst performing, replace with slight mutations of best
69 end for

```

```
70  
71 end subroutine
```

5.10.6 Local Optimization Algorithms

5.10.7 Gradient Descent

The gradient descent method is a reliable way of searching for a local minimum. It requires only the first derivative coupled with a line search to move closer to the minimum point. Close to the minimum, it is slower than the Newton-Gauss and LMA, but these require the computation of the Jacobian to estimate the Hessian matrix, so these are more computationally intensive.

Listing 5.5: Simple simulated annealing subroutine

```
1 // Simulated Annealing  
2 subroutine simulated_annealing(f, p, p_best)  
3  
4 // f is the function being optimised  
5 // p is an array of the parameters  
6  
7 p_search = True  
8 rss_best = calc_rss(f, p)  
9  
10 do while(p_search)  
11  
12 // Some function to calculate the  
13 df(:) = gradient(f,p(:))  
14  
15 // Back track line search  
16 alpha = 1.0  
17 alpha_search = True  
18 last_rss = -1.0  
19 do while(alpha_search)  
20     rss = calc_rss(f, p(:) - alpha * df(:))  
21  
22     // If it's the first step, store rss into last_rss  
23     if(last_rss == -1.0)then  
24         last_rss = rss  
25     end if  
26  
27     // If the new one is worse, stop searching and step alpha back  
28     if(rss > ls_rss)then  
29         alpha_search = False  
30         alpha = 2.0 * alpha  
31         // If it's better, reduce alpha and keep searching  
32     else  
33         alpha = 0.5 * alpha  
34         last_rss = rss  
35     end if  
36 end do  
37  
38 // Check new parameters  
39 rss = calc_rss(f, p(:) - alpha * df(:))  
40  
41 // If it's better, save and keep searching  
42 if(rss < rss_best)then  
43     p(:) = p(:) - alpha * df(:)  
44     rss_best = rss  
45
```

```

46 // Break out
47 else
48 p_search = False
49 end if
50 end do
51
52
53 end subroutine

```

5.10.8 Newton-Gauss

The Newton-Gauss method is an algorithm that finds the local minimum of a function by approximating the Hessian matrix with the Jacobian and its transpose. By estimating the Hessian, the algorithm only requires the computation of the first order derivatives with respect to each parameter, at each point pair x, y . It is possible to use the function and the analytical derivative, but as this work requires the calculation of functions that have no simple analytical form, the first derivative will be approximated within the algorithm.

$$\begin{aligned} \vec{J}^T \vec{J} \vec{p} &= \vec{J}^T \vec{r} \\ \vec{r} &= \vec{y} - f(\vec{x}, \vec{p}) \end{aligned} \tag{5.129}$$

Close to a minima, the NG method quickly moves to the optimum point. Depending on the starting point, it may converge only to a local minimum, and not the global, and it may also be unstable, not converging at all.

5.10.9 Levenberg Marquardt

The LMA is a NG type algorithm that includes the addition of a dampening term that helps to increase the robustness of the algorithm. The particular algorithm here uses a number of other schemes to improve the overall effectiveness of the algorithm.

The starting value for lambda is calculated as a function of the estimate of the Hessian and a cutoff for lambda is introduced to remove dampening altogether. A delayed gratification scheme has also been introduced to increase lambda by 50 percent if the trial solution is worse, and to decrease lambda by a factor of 5 if the trial solution is better. Finally, a diagonal weighting matrix has been included to allow certain parameters to be preferential during their optimisation.

$$\begin{aligned} (\vec{J}^T \vec{W} \vec{J} + \lambda \text{diag}(\vec{J}^T \vec{W} \vec{J})) \vec{p} &= (\vec{J}^T \vec{W} \vec{r}) \\ \vec{r} &= \vec{y} - f(\vec{x}, \vec{p}) \end{aligned} \tag{5.130}$$

Chapter 6

Methodology: Proton Activation and Radioactive Decay

A computer program was developed to calculate the radioactivity of an ion irradiated target. The choice of projectile was limited to protons, but there is the option to increase this to other light ions. The decay equations were derived to be able to calculate the activity of isotopes within the target at time t. The simulation is split into two sections: the first with the beam on, where there will be source terms for isotopes, and the second with the beam off and no source terms. SRIM, an ion transport code, was used to calculate ion trajectory data. Iron was irradiated experimentally by 36MeV protons with the University of Birmingham cyclotron, and its activity was measured and calculated using the activity code. Two versions of the code were created.

6.1 Introduction

High flux neutron reactors are expensive to use, whereas proton accelerators capable of producing beams of adequate fluence and energy are more readily available, cheaper to buy and cheaper to run. Ion beams, due to the Coulomb interaction, are more controllable in terms of energy, direction and fluence. They may be concentrated on a desired target at a set fluence and energy.

Depending on the energy of the ion beam, the target material will become radioactive. The stable nuclei are transmuted and, if unstable, will decay. An equation was derived to predict the activity of isotopes for any decay chain with source terms and branching factors included. Two versions of a computer code were developed to compute the reaction rates and activity of the irradiated targets.

6.2 Activation by Ion Irradiation

The Bateman equation was derived using Laplace transforms, and this same method has been used to develop a modified equation that incorporates branching factors and production rates for each isotope in the decay chain, as illustrated by Figure 6.1.

6.2.1 Laplace Transform

Laplace Transforms (6.1) are a useful mathematical tool, and allow ordinary differential equations to be solved by simple algebraic manipulation in the s domain. Bateman took advantage of Laplace Transforms in deriving his equation, and this is the method that has been taken here as well.

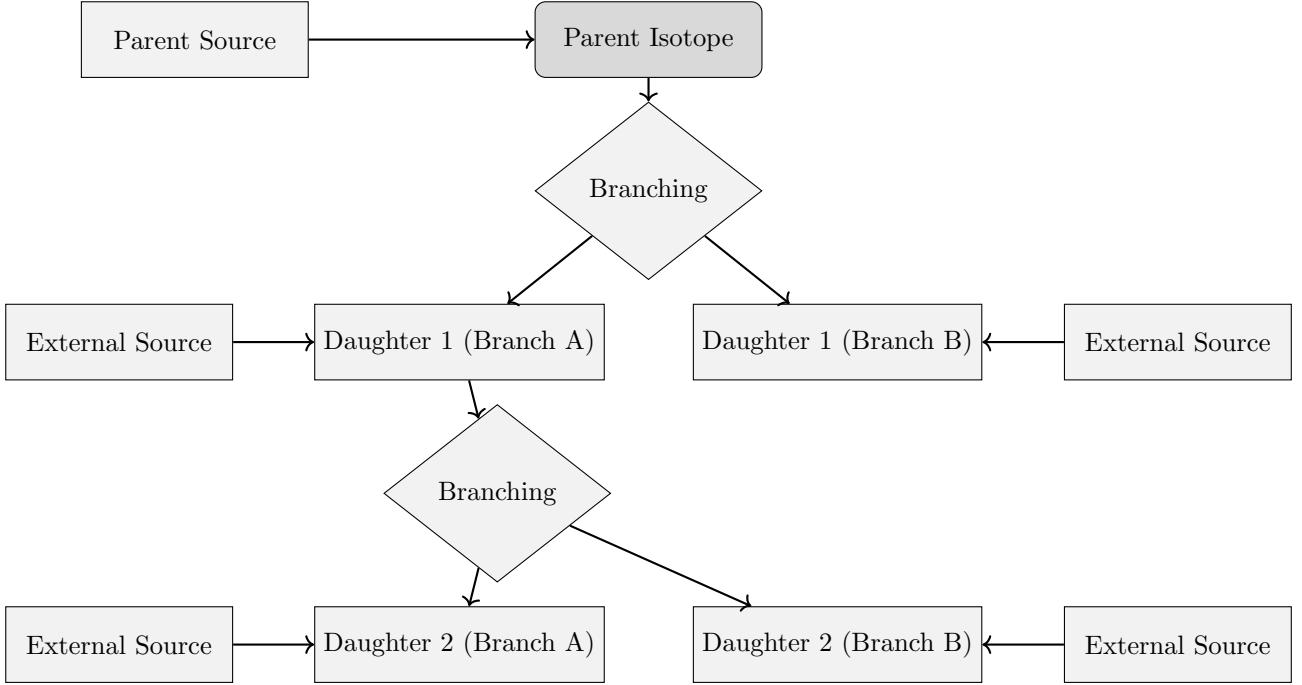


Figure 6.1: An example of several decay chains including branching factors and possible external source terms for each isotope on each chain.

$$F(s) = \int_0^{\infty} f(t) \exp(-st) dt \quad (6.1)$$

6.2.2 Constructing the Differential Equations

The first step is to set up differential equations for the parent isotope, unstable daughter isotopes and stable daughter isotope. The parent isotope has a source term, due to production, and a loss term, due to decay. The unstable daughter isotopes have two source terms, from the production by irradiation induced transmutation and the decay of preceding isotopes in the decay chain, and a loss term, due to decay. Finally, the stable daughter that finalizes the decay chain has two source terms (the same as the unstable daughters) but no loss term.

The variables (and vectors) used in these equations are defined as follows:

- $\vec{\lambda}$ vector containing isotope decay constants λ_i
- \vec{b} vector containing isotope to isotope branching factors b_i
- \vec{w} vector containing isotope production rates w_i
- t time at which activity/amount of isotope is measured
- $N_i(0)$ starting amount of the i^{th} isotope
- $N_i(t)$ amount of the i^{th} isotope at time t
- $N'_i(t)$ change in amount of the i^{th} isotope, with respect to time, at time t

The differential equations for the parent isotope (first isotope), unstable daughter isotopes (i^{th} isotopes) and stable, final, daughter isotope (z^{th} isotope) in the time domain are as follows:

$$N'_1(t) = \omega_1 - \lambda_1 N_1(t) \quad (6.2)$$

$$N'_i(t) = \omega_i + b_{i-1} \lambda_{i-1} N_{i-1}(t) - \lambda_i N_i(t) \quad (6.3)$$

$$N'_z(t) = \omega_z + b_{z-1} \lambda_{z-1} N_{z-1}(t) \quad (6.4)$$

Applying the Laplace Transform to these three differential equations allows them to be manipulated and solved algebraically in the s-domain.

$$N_1(s) = \frac{1}{s + \lambda_1} N_1(0) + \frac{1}{s(s + \lambda_1)} \omega_1 \quad (6.5)$$

$$N_i(s) = \frac{1}{s(s + \lambda_i)} (\omega_i) + \frac{1}{s + \lambda_i} (b_{i-1} \lambda_{i-1} N_{i-1}(s)) + \frac{1}{s + \lambda_i} N_i(0) \quad (6.6)$$

$$N_z(s) = \frac{1}{s^2} \omega_z + \frac{1}{s} b_{z-1} \lambda_{z-1} N_{z-1}(s) + \frac{1}{s} N_z(0) \quad (6.7)$$

6.2.3 Numerical Inversion of the Laplace Transform

The Gaver-Stehfest[92] algorithm was developed in the 1960s and 1970s and is a method of calculating the inverse of a Laplace Transform in the real number domain. It is an easy to implement and reasonably accurate method, although it is an approximation to the real value. A comparison between an analytic and numeric inversion for the unstable isotope Po-218 is discussed at the end of this section (figure 6.2).

$$f(t) \approx f_n(t) = \frac{\ln(2)}{t} \sum_{k=1}^{2n} a_k(n) F(s) \text{ where } n \geq 1, t > 0 \quad (6.8)$$

$$s = \frac{k \ln(2)}{t} \quad (6.9)$$

$$a_k(n) = \frac{(-1)^{(n+k)}}{n!} \sum_{j=\text{Floor}(\frac{k+1}{2})} j^{n+1} \binom{n}{j} \binom{2j}{j} \binom{j}{k-j} \quad (6.10)$$

The equation for the i^{th} isotope may be calculated by recursively calculating the equations by numeric inversion, starting from the first (parent isotope) and inserting the result into each subsequent recursion until the i^{th} isotope is reached (changing the equations appropriately for the parent, unstable daughter and stable daughter isotopes).

6.2.4 Analytic Solution by Partial Fraction Expansion

The equation for the i^{th} isotope in the s domain can be written in full by substituting the preceding equation until the parent isotope is reached, and this full equation may be rearranged with the production amount of each isotope and starting amount of each isotope in individual terms. Each of these terms is multiplied by a fraction that can be expanded, using partial fractions, and inverted analytically.

This is illustrated with an example unstable isotope, fourth in the decay chain (including the parent isotope):

$$\begin{aligned}
N_4(s) = & \frac{1}{(s + \lambda_1)(s + \lambda_2)(s + \lambda_3)(s + \lambda_4)} b_2 b_3 b_4 \lambda_1 \lambda_2 \lambda_3 N_1(0) \\
& + \frac{1}{(s + \lambda_2)(s + \lambda_3)(s + \lambda_4)} b_3 b_4 \lambda_2 \lambda_3 N_2(0) \\
& + \frac{1}{(s + \lambda_3)(s + \lambda_4)} b_4 \lambda_3 N_3(0) \\
& + \frac{1}{(s + \lambda_4)} N_4(0) \\
& + \frac{1}{s(s + \lambda_1)(s + \lambda_2)(s + \lambda_3)(s + \lambda_4)} b_2 b_3 b_4 \lambda_1 \lambda_2 \lambda_3 \omega_1 \\
& + \frac{1}{s(s + \lambda_2)(s + \lambda_3)(s + \lambda_4)} b_3 b_4 \lambda_2 \lambda_3 \omega_2 \\
& + \frac{1}{s(s + \lambda_3)(s + \lambda_4)} b_4 \lambda_3 \omega_3 \\
& + \frac{1}{s(s + \lambda_4)} \omega_4
\end{aligned} \tag{6.11}$$

An example stable isotope, fourth (last) in the decay chain (including the parent isotope):

$$\begin{aligned}
N_4(s) = & \frac{1}{s(s + \lambda_1)(s + \lambda_2)(s + \lambda_3)} b_2 b_3 b_4 \lambda_1 \lambda_2 \lambda_3 N_1(0) \\
& + \frac{1}{s(s + \lambda_2)(s + \lambda_3)} b_3 b_4 \lambda_2 \lambda_3 N_2(0) \\
& + \frac{1}{s(s + \lambda_3)} b_4 \lambda_3 N_3(0) \\
& + N_4(0) \\
& + \frac{1}{s^2(s + \lambda_1)(s + \lambda_2)(s + \lambda_3)} b_2 b_3 b_4 \lambda_1 \lambda_2 \lambda_3 \omega_1 \\
& + \frac{1}{s^2(s + \lambda_2)(s + \lambda_3)} b_3 b_4 \lambda_2 \lambda_3 \omega_2 \\
& + \frac{1}{s^2(s + \lambda_3)} b_4 \lambda_3 \omega_3 \\
& + \frac{1}{s^2} \omega_4
\end{aligned} \tag{6.12}$$

By using partial fraction expansion and standard Laplace Transforms, the set of equations below is used to calculate the amount of the m^{th} isotope in the decay chain, providing the m^{th} isotope is unstable.

$$N_m(t; \vec{\lambda}, \vec{b}, \vec{w}) = \sum_{k=1,m} r(k; \vec{\lambda}, \vec{b}) \left[f(t; k, m, \vec{\lambda}) N_k(0) + g(t; k, m, \vec{\lambda}) w_k \right] \tag{6.13}$$

$$r(k, m, \vec{\lambda}) = \begin{cases} \prod_{i=k, m-1} (b_{i+1} \lambda_i), & \text{if } k < m \\ 1, & \text{if } k = m \end{cases} \tag{6.14}$$

$$f(t; k, m, \vec{\lambda}) = (-1)^{m-k} \sum_{i=k, m} \left[\exp(-\lambda_i t) \prod_{j=k, m; j \neq i} \left(\frac{1}{\lambda_i - \lambda_j} \right) \right] \tag{6.15}$$

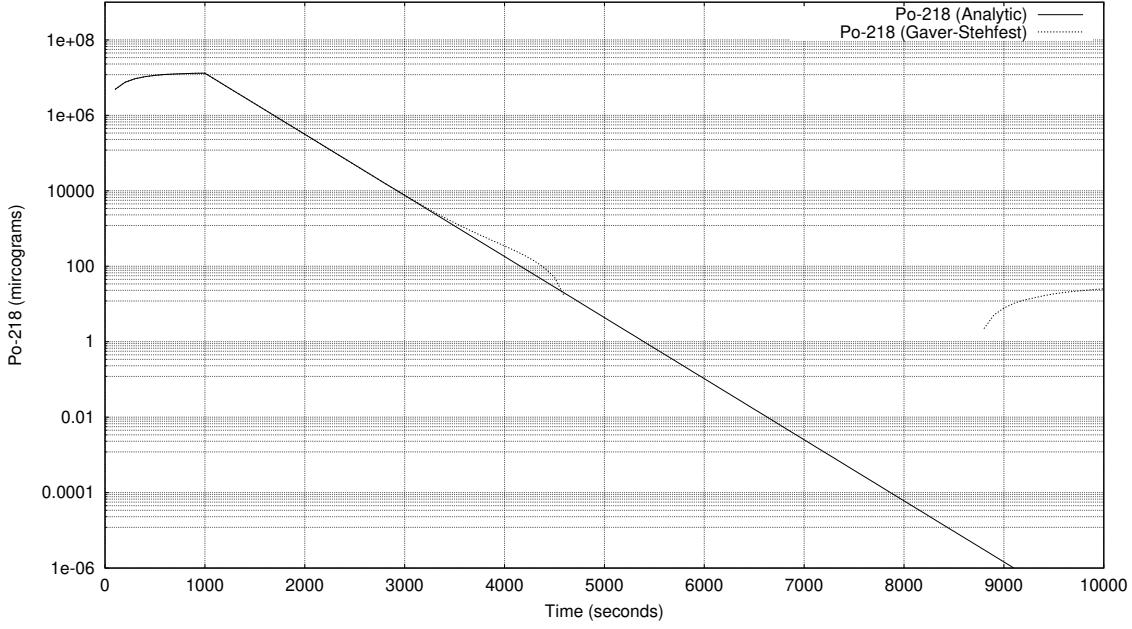


Figure 6.2: Decay of Po-218: Analytic and Gaver-Stehfest Calculations [56]

$$g(t; k, m, \vec{\lambda}) = \frac{1}{\prod_{i=k, m} \lambda_i} + (-1)^{m-k+1} \sum_{i=k, m} \left[\frac{1}{\lambda_i} \exp(-\lambda_i t) \prod_{j=k, m; j \neq i} \left(\frac{1}{\lambda_i - \lambda_j} \right) \right] \quad (6.16)$$

The set of equations below is used to calculate the amount of the m^{th} isotope in the decay chain, where the m^{th} isotope is stable.

$$N_m(t; \vec{\lambda}, \vec{b}, \vec{w}) = N_m + w_m t + \sum_{k=1, m-1} r(k; \vec{\lambda}, \vec{b}) \left[f(t; k, m-1, \vec{\lambda}) N_k(0) + g(t; k, m, \vec{\lambda}) w_k \right] \quad (6.17)$$

$$r(k, m, \vec{\lambda}) = \begin{cases} \prod_{i=k, m-1} (b_{i+1} \lambda_i), & \text{if } k < m \\ 1, & \text{if } k = m \end{cases} \quad (6.18)$$

$$f(t; k, m, \vec{\lambda}) = \frac{1}{\prod_{i=k, m} \lambda_i} + (-1)^{m-k+1} \sum_{i=k, m} \left[\frac{1}{\lambda_i} \exp(-\lambda_i t) \prod_{j=k, m; j \neq i} \left(\frac{1}{\lambda_i - \lambda_j} \right) \right] \quad (6.19)$$

$$g(t; k, m, \vec{\lambda}) = \frac{1}{\prod_{i=k, m} \lambda_i} t + \frac{\sum_{i=k, m} \left[\prod_{j=k, m; j \neq i} \lambda_j \right]}{\prod_{i=k, m} \lambda_i^2} + (-1)^{m-k+1} \sum_{i=k, m} \left[\frac{1}{\lambda_i^2} \exp(-\lambda_i t) \prod_{j=k, m; j \neq i} \left(\frac{1}{\lambda_i - \lambda_j} \right) \right] \quad (6.20)$$

6.2.5 Preference: Analytic over Numeric

The numeric solution only requires the equation to be solved in the s-domain; the Gaver-Stehfest algorithm performs the inversion. It is worth the extra effort to derive and implement an analytic solution, as the numeric is only an approximation. Examples of the pitfalls of the numeric solution are that it can give negative amounts of an isotope and the difference between the numeric and analytic calculated amounts can become quite large when the isotope decays away to a very small value. Figure 6.2 shows the predicted decay of a sample of Po-218 irradiated for 1,000s, and sampled until 10,000s. In the region between 4,000s and 9,000s the amount from the numeric calculation drops below zero, whereas the analytic calculation remains above zero, as would be expected.

6.3 Computational Methods

The Activity program has been developed in Fortran and takes advantage of MPI (Message Parsing Interface) to speed up calculation times by allowing the use of multiple processes in parallel. It has a self contained maths library, although this could be improved in the future by using optimised maths libraries for certain functions (e.g. matrix operations).

The code was developed on a Debian based distribution of Linux, but it should be supported on other variants of Linux and Unix, and does not require any specialist hardware.

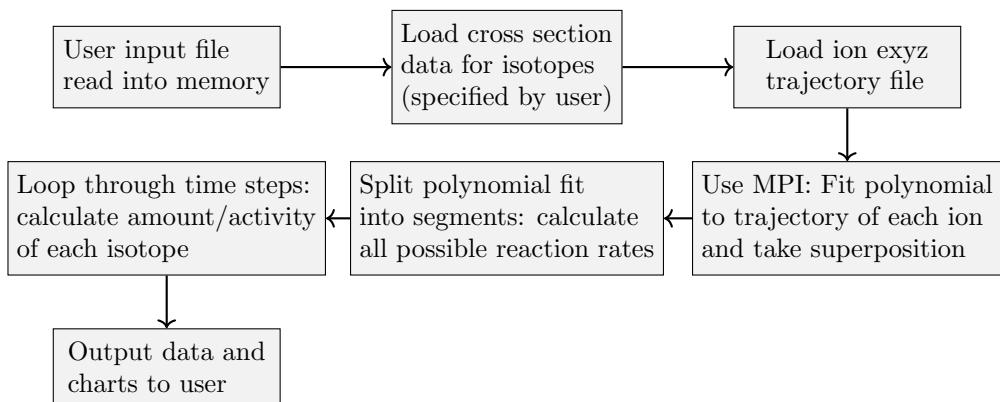


Figure 6.3: Flow chart of major processes in the Activity code

The user is required to prepare an input file that contains the instructions required to perform a calculation. In addition to the input file, the user must provide an EXYZ ion trajectory file output by SRIM. Activity will read in the user input file, and the SRIM and data files listed within, before performing the calculation. Figure 6.3 shows a flowchart of the major steps the code performs.

There are various settings in the user input file, but the main ones relating to the simulated experiment are:

- Element composition of target (percentage by mass).
- Beam flux (current), energy, duration and area on target.
- Activity measurement time (end of the “experiment”).
- Material density.
- Target thickness.

6.4 Ion Trajectory Data

The SRIM code is used to generate ion trajectory data. This uses a quantum mechanical treatment of ion-atom collisions and statistical algorithms to speed up the process[93].

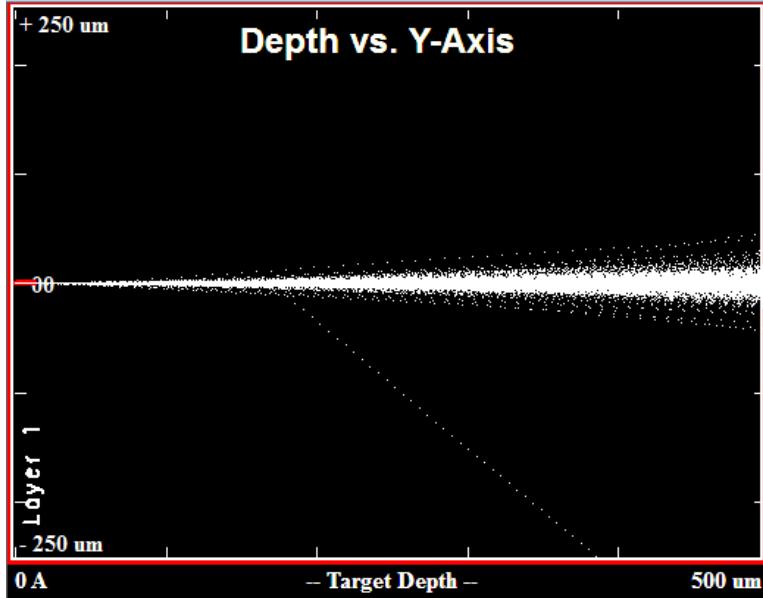


Figure 6.4: 36MeV Ion Track

The trajectory data files used by both versions of the Activity code will be created using SRIM for the required beam energy and target combinations.

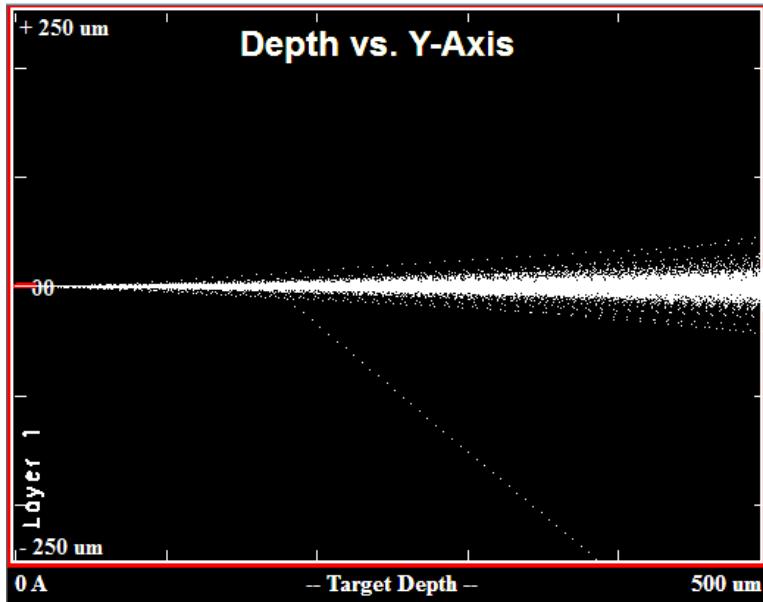


Figure 6.5: 36MeV Ion Track

The target is set as pure Iron, and due to the way SRIM functions, the structure type is inconsequential. At just 0.5mm in thickness, the 36MeV ions pass through the targets and exit with an energy above 30MeV, and the exyz.txt data file covers this range of ion energies.

6.5 Activation by Neutron Irradiation

6.5.1 Introduction

A short program was created, based on the existing code developed to calculate ion activation. It is a simple non-transport neutron activation code, used to estimate the activity and subsequent cooling of materials irradiated by neutrons. A single energy neutron flux or Maxwell-Boltzmann distribution may be selected by the user and the TENDL-2019 data file is used to provide cross sections for calculating reaction rates. Finally, the previously derived extended Bateman equations are used to calculate isotopes at time t after irradiation begins.

6.6 Cyclotron Irradiated Iron

6.6.1 Cyclotron Beam Line

The Scanditronix MC40 Cyclotron at the University of Birmingham has several beamlines and is capable of accelerating protons, deuterons, Helium 3 and Helium 4 and fluxes and energy ranges detailed below. After running a simulation with SRIM, the 36MeV protons were expected to create an average of 15.8 vacancies each.

Particle	Energy (MeV)	Max Current (micro A)	Flux (ions per second)
p	8-40	60	3.75×10^{14}
d	8-40	30	1.87×10^{14}
$^4He^{2+}$	8-53	30	9.36×10^{13}
$^3He^{2+}$	4-20	60	1.87×10^{14}

Table 6.1: Beam Characteristics of the Scanditronix MC-40

The cyclotron is located in one room, and several beam lines run from the cyclotron to other rooms in the cyclotron building. The operation room is separate from the cyclotron and beam line rooms (fig 6.6).

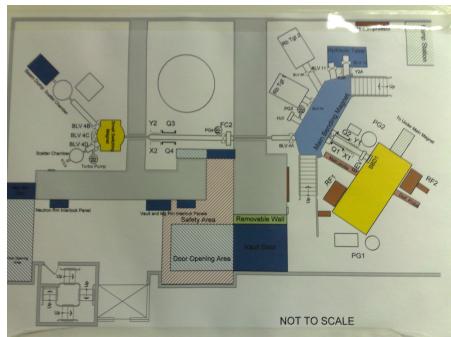


Figure 6.6: Cyclotron layout - main hall and beam lines

The iron sample was held perpendicular to the beam line, with the ions passing through the 0.5mm thickness of the sheet. The beam area was approximately $6.4 \times 10^{-5} m^2$, irradiating a volume of approximately $3.2 \times 10^{-8} m^3$. The beam, irradiating the target for 300 seconds at 0.5 micro amps was expected to cause over 1.4×10^{16} displacements within the volume of iron targeted by the beam. With a number density of approximately 8×10^{28} atoms per cubic meter, giving a relatively low damage dose, when compared to that expected over the lifetime of a component within the reactor, of 5^{-6} DPA.

During irradiation, the amount of Gamma radiation released was enough to trip the alarm for the room (6.7). The proton fluence was less than 1% of the maximum fluence capable of being produced by the cyclotron, so



Figure 6.7: Gamma warning alarm during 0.5 microamp 36MeV proton irradiation of iron

this was definitely a concern. To increase the damage dose, and keep to a shorter period of time, the current would most probably be increased to a much higher percentage, also increasing the rate of Gammas produced during irradiation.

6.6.2 Measurement of Sample Activity

The sample was too radioactive to safely handle immediately after irradiation, so it was left to cool for several days before taking measurements. After it was cooled, a high purity germanium detector (fig 6.8) was used to measure the activity of the irradiated sample.



Figure 6.8: High Purity Germanium Detector

The detector and preamplifier are both cooled by liquid nitrogen to 77K, as the band-gap of germanium is 0.7eV and the thermal motion of electrons and nuclei at higher temperatures would induce currents causing noise and interference in the detector. As a gamma ray enters the detector it creates an electron-hole pair in the medium of the detector. The holes and electrons are attracted to the center and to the outer cylinder of the detector, or vice versa.

6.6.3 Prediction of Activity

Following the measurement of the foil target, in the first instance the predicted activity will be estimated using the average ion energy, the cross section of this energy and the two reactions that result in Co-55 ($^{54}_{26}Fe(p,\gamma)^{55}_{27}Co$

and $^{56}_{26}Fe(p, 2n)^{55}_{27}Co$. The activity code (version 1 and version 2) are then used to predict the activity of the sample as well as the expected gamma spectra.

Chapter 7

Methodology: Interatomic Potential Fitting

In order to perform MD ion damage simulations, an Iron-Palladium potential is needed. Experimental data is available for FCC palladium, but DFT is needed to compute the properties of FCC iron, as it does not exist in this phase at standard conditions without the addition of an austenite stabilizer. Additional DFT calculations will be used to compute the energy, stress and forces of slightly perturbed atoms which will in turn be used to fit the potentials. A new computer code will be created to fit the potentials.

7.1 Interatomic Potential Fitting

A summary of the process, step by step, towards a new potential has been outlined.

1. Collate experimental values (a_0 , e_{coh} , B_0 , elastic constants etc)
2. Generate or choose pseudopotentials for iron and palladium
3. Converge ecutwfc and ecutrho for these pseudopotentials
4. Converge k-points and smearing
5. Calculate relaxed positions and energies
6. Calculate equation of state where experimental data aren't available
7. Calculate elastic constants where experimental data aren't available
8. Randomise positions of atoms and calculate total energy and the forces between atoms with DFT
9. Insert vacancies, interstitials or both then calculate total energy and the forces between atoms with DFT
10. Select the type of potential to be used (FS, EAM, 2BEAM)
11. Select forms of the potential functions
12. Choose a start point
13. Fit potential

Several computer codes are developed to complete a number of the steps in the list.

7.2 Experimental & DFT Bulk Property Data

The bulk properties for Iron (BCC) and Palladium are available on several websites and in published work. The lattice parameter, bulk modulus and elastic constants are available for Iron BCC and Palladium FCC, but as pure Iron FCC is theoretical under normal conditions, this will need to be predicted using DFT.

A large amount of computing power is required to run the DFT calculations. If the input parameters are not accurate enough to begin with, it will either take a long time for the DFT calculation to run, or it will fail to converge completely. The lattice parameter for each element, using the density of that pure element in its state under normal conditions, is predicted for that element either for both the FCC and BCC structures.

Element	Atomic Mass	Density kg/m ³	Atoms/m ³	FCC (Bohr/Angstrom)	BCC (Bohr/Angstrom)
Al [94]	26.98	2700	6.02×10^{28}	7.66/4.05	6.08/3.22
Cr [95]	52.00	7140	8.27×10^{28}	6.89/3.64	5.47/2.89
Fe [95]	55.84	7874	8.47×10^{28}	6.83/3.61	5.42/2.87
Ni [96]	58.69	8908	9.09×10^{28}	6.67/3.53	5.30/2.80
Pd [97]	106.42	12023	6.83×10^{28}	7.34/3.88	5.83/3.08

Table 7.1: Predicted lattice parameters based on the density, atomic number and type of structure

Aluminium appears here because it has a simpler electronic structure and no magnetic properties to complicate the DFT calculations. The DFT calculations for Aluminium complete much faster than for the other elements listed, so it was used throughout to develop and test the computer codes created.

7.3 DFT Calculations

7.3.1 Quantum Espresso

There are a choice of several different DFT programs to choose from, including VASP, Siesta and QuantumEspresso. The open source QuantumEspresso includes the PWscf binary that solves electronic structure calculations with plane wave pseudopotentials. It calculates the total energy of a system as well as the forces between atoms and the stress on the simulation box.

As the calculations contain Iron, it is important to take into account the magnetism caused by the filling of the d-shell. PWscf gives three modes: non-magnetic, collinear spin and non-collinear spin.

7.3.2 Pseudopotential Selection

The pseudopotentials were downloaded from the Quantum Espresso PSLibrary with url:

<http://www.quantum-espresso.org/pseudopotentials/pslibrary>

There are several categories of pseudopotential available. The pseudopotentials may be fully relativistic or not. The DFT calculations will be collinear or no spin calculations, and the element with the largest number of electrons will be Palladium, with electrons in the s, p, d shells. There will be no elements used in this work with f shell electrons, so a non-relativistic pseudopotentail will be used for each element.

There are a number of choices of exchange-correlation functional, depending on the element:

- pz: Perdew-Zunger (LDA)
- vwn: Vosko-Wilk-Nusair (LDA)

- pbe: Perdew-Burke-Ernzerhof (GGA)
- blyp: Becke-Lee-Yang-Parr (BLYP)
- pw91: Perdew-Wang 91 gradient-corrected

In the literature, LDA and BLYP are used for organic DFT calculations. LDA and GGA type pseudopotentials have been used to model solids and, in particular, the GGA type have been used to study metals as they are more reliable at calculating parameters such as a_0 .

Several elements were tested to compare the results of LDA and GGA pseudopotentials to the experimental values for the lattice parameter of each.

The settings used were:

- ecutwfc: 71
- ecutrho: 430
- k-points: 9 9 9 Monhurst-Pack grid offset
- smearing: 0.04 Ry Mazari-Vanderbilt cold smearing
- calculation type: non-polarised calculation
- pseudopotentials: PZ for LDA, PBE for GGA

The results were computed as follows:

Element	Crystal	a_0 exp. (ang)	a_0 LDA (ang)	a_0 GGA (ang)
Al	FCC	4.05	3.98	4.04
Fe	BCC	2.87	2.69	2.75
Pd	FCC	3.89	3.84	3.93
Cu	FCC	3.61	3.49	3.59

Table 7.2: Predicted lattice parameters based on the density, atomic number and type of structure

The simpler LDA pseudopotentials consistently underestimate the lattice parameter a_0 for the four metals tested, and on average they underestimate by 3%. The GGA calculations are within 1.5% of the experimental value

7.3.3 Smearing Type

The material being studied here is a metal/metal alloy, and this has important consequences when setting up the calculation. The conduction and valence bands overlap in a metal, and the fermi energy passes through this overlap. When the occupied states are summed, integrating over all k-points k[98], those bands that cross the Fermi energy will drop to zero. This abrupt drop off has two practical solutions: increase the number of k-points, or add a smearing to smooth the occupation function.

Previous work has given the following suggested smearing types and values as a start point.

- Degauss values 0.01Ry Methessel-Paxton [99]
- Fermi-Dirac 0.01eV (0.00074Ry) [100]
- Marzari-Vanderbilt 0.01Ry [101]

- Degauss 0.03 + 0.05 [102]
- Marzari-Vanderbilt 0.05Ry [103]

This suggests that a range between 0.01Ry to 0.05Ry should be investigated as a sensible range of smearing energies. There are a choice of four smearing functions in QuantumEspresso.

- Gaussian
- Fermi-Dirac
- Methfessel-Paxton
- Marzari-Vanderbilt

Functions such as the Gaussian and Fermi-Dirac are not cold smearing, and will converge to the wrong energy whereas the Mazari-Vanderbilt and Methfessel-Paxton are designed to reduce and heating whilst smearing the electron wavefunction. An author of QuantumEspresso, N. Marzari, developed the cold smearing Marzari-Vanderbilt function, and this was selected as it had appeared a number of times in the literature and throughout the QuantumEspresso tutorials.

7.4 Parameter Convergence

The energy cutoff, charge density, k-point and smearing values must be selected carefully for any similar DFT calculation. There is a balancing act between the accuracy of the result of a particular calculation and computational time, and this will also vary with the pseudopotential, complexity of the electron configuration of the atoms, whether or not the calculation is with or without spin, and so on.

The pseudopotentials and smearing type have already been selected, and the first set of calculations will not include magnetism. The parameters must work well with high symmetry geometries, such as a BCC or FCC crystal with the atoms in their exact position, but also where the atoms have been slightly perturbed from their exact position in the lattice.

7.4.1 Ecutwfc and Ecutrho

A primitive FCC or BCC is created and the atoms are slightly perturbed, breaking symmetry but also putting forces on each atom within the cell. A high reasonably high number of k-points are used and the smearing value should be low.

Pick a low ecutwfc and set the ecutrho equal to four times the value of the ecutwfc. Depending on the pseudopotential, the calculation may fail because the values are too small. If this is the case, continue increasing the value in small (perhaps 5Ry) increments until the calculation runs successfully.

Next, continue to increase the ecutwfc value, also increasing the value of ecutrho as four times the value,in steady increments. Once the calculated force and energy per 1Ry increment fall below a threshold set by the user, the values have converged. In this work, the convergence values were:

- Energy convergence per 1Ry: 1.0×10^{-6} Ry (1.36×10^{-5} eV)
- Force convergence per 1Ry: 1.0×10^{-5} Ry/Bohr (2.57×10^{-4} eV/Ang)

At this point, it may be possible to reduce the ecutrho value further while remaining within this convergence threshold, so the ecutwfc value is held fixed while the ecutrho value is reduced in steps of 5Ry.

Finally, attempt to reduce the value of ecutwfc further in 1Ry steps while holding the ecutrho value while remaining within the convergence thresholds.

7.4.2 K-Points and Smearing Value

A configuration with a similar size and number of atoms as those in the calculations should be created. In this work, the reference database and bulk properties will be calculated using 32 atom configurations in an approximately 7x7x7 angs box. The Iron FCC k-points and smearing are calculated using a 32 atom FCC box with sides length 7.2 angs and the Palladium with a 32 atom FCC box with sides length 7.8 angs.

As with the ecutwfc and ecutrho convergence calculations, the atoms are slightly perturbed from their exact crystal positions, putting a force on the atoms and breaking the symmetry (as will be the case for the configurations used for fitting).

The convergence process is more complex for k-points and smearing. There might be a "better" convergence (a smaller convergence value and a decrease in the computation time and memory requirements) for combinations with a higher smearing value, but it is preferred to have a smaller smearing value with a higher number of k-points, while maintaining a reasonable computing time, as the greater the smearing the further the calculated value is from the actual calculated value (with no smearing and a very high number of k-points).

A 2D plot of energy and force values as a function of number of k-points and smearing may then be used to choose reasonable settings that balance the requirements.

7.4.3 Automating Convergence

The process requires the creation of many input files, and manually editing these files is time consuming and vulnerable to human error. Over 100 input files may be required to converge the ecutwfc and ecutrho values, with a further 50-100 required to generate the 2D k-point and smearing plots.

A python code is developed to automate this process and plot the results automatically for the user (section 9.2).

7.5 Bulk Property Calculations

7.5.1 QEEOS Python Program

A Python program (section 9.2) has been developed to calculate bulk properties of a structure using the DFT code Quantum Espresso. It is able to fit the equation of state for a cubic structure, and elastic constants for orthorhombic structures (which includes the subset of cubic structures). In brief, the user provides a template PWscf input file, and the program adjusts the crystal basis vectors, applying the homogenous strain and 9 distortions required to calculate the equation of state and elastic crystals, as discussed earlier in this work. The highlights of this code and output data are detailed in the results section and appendix.

7.5.2 DFT Calculations: Austenitic Iron Properties

Experimental data was available for the bulk properties of pure BCC iron and pure FCC palladium. The properties of theoretical (under normal conditions) FCC iron are calculated using the QEEOS code with Quantum Espresso.

7.6 Reference Database

7.6.1 QEFORFIT Python Program

7.6.2 DFT Calculations: Iron and Palladium

7.6.3 Atomic Configurations for DFT Calculations

The interatomic potentials are designed to reproduce the forces, energies, stresses and bulk properties

7.7 DFT Errors and Convergence Failures

7.7.1 Adjustment of DFT Parameters

Throughout the DFT segment of this work, there were a number of issues that either resulted in the the calculations failing or completeing but not converging. There were a number of causes and solutions that were used throughout the process.

If the parameters for the pseudopotential and system aren't converged, there is a danger that not enough k-points or the ecut parameters are too small to converge.

There are a selection of mixing modes available in PWscf, giving a choice of Thomas-Fermi screening for homogeneous systems, local density dependent Thomas Fermi screening for inhomogeneous screening and Broyden mixing as the default. In this work, most success was found with plain Broyden mixing and a smaller than the default mixing factor (mixing_beta).

7.7.2 A Lack of Memory

The DFT calculations are performed on a supercomputer with many processors and many cores per processor. The program uses two implementations of parallelization.

- OpenMP - many threads have access to the same shared memory
- OpenMPI/MPICH - many processes, each with it's own memory

Whilst some favourable results have been shown using a hybrid of OpenMP and OpenMPI, it was more straight-forward to use in OpenMPI mode only in this work. Unfortunately, the amount of memory per process overwhelmed the computing nodes for certain calculation. The job script was modified to reserve entire nodes and all the memory on each node, but to only use the cores on one of the two processors to halve the processes per node, but double the memory per process. It's quicker to do this than wait in the high memory queue, where the memory per process would be perhaps double than that required.

7.7.3 Not Straying Too Far From Reality

It is important to provide the calculation with reasonable and realistic configurations and settings. Where a pure element such as Aluminium, Iron or Platinum was being investigated, the known experimental lattice parameters were used as a starting point. If the parameters were too far from these values, for example inputting a lattice parameter of either 2 or 8 angs for Aluminium rather than 4, as a starting approximation, the DFT calculation might struggle to relax the structure, or to even successfully run or converge a calculation.

The FCC Iron structure lattice parameter was estimated from the density of FCC steel, and the lattice parameter for alloys could similarly be estimated within a reasonable margin to use as a start point for the calculation.

7.8 EAMPA: Potential Analysis and Fitting Code

A python-fortran based computer code is developed to fit interatomic potentials to bulk properties and DFT calculated forces and energies. It has been designed to take advantage of both Fortran and Python. Fortran is used to compute the neighbour lists, energies, stresses and forces, and Python is used to read and write data, produce plots and control the potential fitting.

Highlighted code is discussed in the results (section 10.1) and the appendix of this work.

7.9 Iron-Palladium Potentials

The experimental data and DFT computed bulk properties will be used along with the database of reference configurations and the EAMPA potential fitting code to produce the potentials. Following this, the EAMPA code will then be used to check the bulk properties that the potentials give, as well as the cohesive energy and surface energy plots.

7.10 DL_POLY Contribution

DL_POLY is a Molecular Dynamics code developed by W. Smith, T.R. Forester and I.T. Todorov at Daresbury Laboratory in Warrington. It is written in Fortran and, before the modifications, included a number of potential types for metals including

- Finnis Sinclair
- EAM
- EEAM

The Finnis Sinclair is a particular form of the EAM potential, and EEAM is a modification where, if the metal is an alloy, the density and embedding functional for each atom type are treated separately.

7.10.1 Modifying DL_POLY: 2BEAM

A meeting was held with Dr. Todorov at Daresbury Laboratory, where a brief overview of the relevant code was given. After this point, and corresponding by email, the two band EAM type was added (2BEAM). Two major additions to the code were an array used to store the d-band density function and d-band embedding functional, as well as the s-band function/functional (which are stored in the existing two arrays). The calculation of energy and forces on atoms was also altered to make use of these arrays when a two band EAM potential is used.

7.10.2 Mailshot Extract

Listing 7.1: Mailshot Extract

```
1 DL_POLY_4.05: New Release & Events - MAILSHOT 013
2
3 NEW FEATURES \& IMPROVEMENTS
4 -----
5 1. New two band (2B) EAM and EEAM potentials for metals (TEST45 and TEST46).
6
7 Acknowledgements
8 -----
9 Ben Palmer \@ University of Birmingham (UK) for contributing to the
10 development and testing of the 2BEAM for metals;
11
12 Regards,
13
14 Ilian Todorov
15 July 2013
```

Chapter 8

Results: Activity Code

The aim of the first part of this work was to derive an equation capable of calculating the radioactivity of an isotope and all the subsequent isotopes in the decay chain, taking into account branching factors and source terms. This was achieved and a computer code was developed in Fortran to predict the radioactivity of a target irradiated by an ion beam.

The equation has been written into several Python 3 functions and a second code was developed to estimate radioactivity of a thin foil irradiated by neutrons passing through perpendicular to the plane of the foil.

8.1 Extended Bateman Equations

The Bateman equations were derived from scratch to include a source term for each isotope and branching factors from parent isotope to daughter isotope(s). This process along with the numerical and analytic solutions are outlined in Chapter 6.

8.2 Activity V1 Computer Package

A large proportion of the results for this part of the work was the development of a computer program to implement the decay equations and calculate reaction rates of isotopes, protons and production rates of resultant isotopes.

The source code and instructions on how to use the program are available to download from GitHub.

Program	Repository
Activity	https://github.com/BenPalmer1983/activity

Table 8.1: Activity V1 Code on GitHub

Activity V1 uses the TENDL-2009 cross section database, as well as several other data files that contain element, isotope and decay data. The user provides an ion trajectory file and an input file that specifies beam parameters and target composition.

8.3 Activity V2 Computer Package

There were a number of issues with version 1 of the computer code. It was written in Fortran, but over time this has been a stumbling block for new users, especially those new to Linux and unfamiliar with Fortran.

In the first version of the code, the trajectory data was replaced with a series of polynomials, but this method was not a reliable solution and it was preferred to use the data directly to calculate the reaction rates. Finally, a language such as Python3 is more widely used and contains very useful features such as dictionaries and a plotting module (matplotlib).

The source code and instructions on how to use the program are available to download from GitHub.

Program	Repository
Activity	https://github.com/BenPalmer1983/activity_v2

Table 8.2: Activity V2 Code on GitHub

Activity V2 uses the TENDL-2019 cross section database for cross section reaction data and the JEFF3.1.1 datafile for decay data. As with version 1, the user provides an ion trajectory file and an input file that specifies beam parameters and target composition.

8.4 Neutron Activation

A small side-project was the development of a simple neutron activation code that used the TENDL-2019 neutron cross section data file to estimate the activity of a neutron irradiated target. The reason for this was to be able to estimate the activity of components irradiated in a high flux neutron reactor and compare these to the radioactivity of proton irradiated components.

The source code and instructions on how to use the program are available to download from GitHub.

Program	Repository
Neutron Activation	https://github.com/BenPalmer1983/neutron_activation

Table 8.3: Neutron Activation Code on GitHub

8.5 Proton Irradiated Iron

In this work, Iron has been irradiated with a proton beam. Using the TENDL cross section libraries, the Cobalt-55 activity has been estimated and this, along with other expected gammas, have been predicted using the Activity code.

8.5.1 Cobalt-55

Cobalt-55 is a relatively short lived isotope, with a half life of 17.5 hours, that is a concern to health during and shortly after irradiation. Cobalt-55 decays into another radioactive isotope, Iron-55, which then decays into Manganese-55. The second step in the decay chain is 2.73 years, so the first step with a much shorter half life will be responsible for the majority of the radiation shortly after being irradiated by protons.

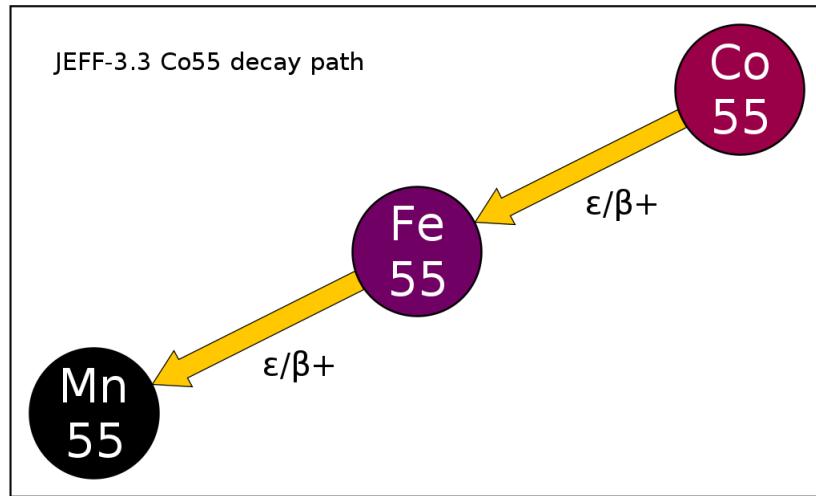


Figure 8.1: Decay path of Cobalt-55

The decay of Cobalt-55 will produce a range of gammas, but one in particular to focus on when measuring the radioactivity is the 931KeV gamma with an intensity of 75%.

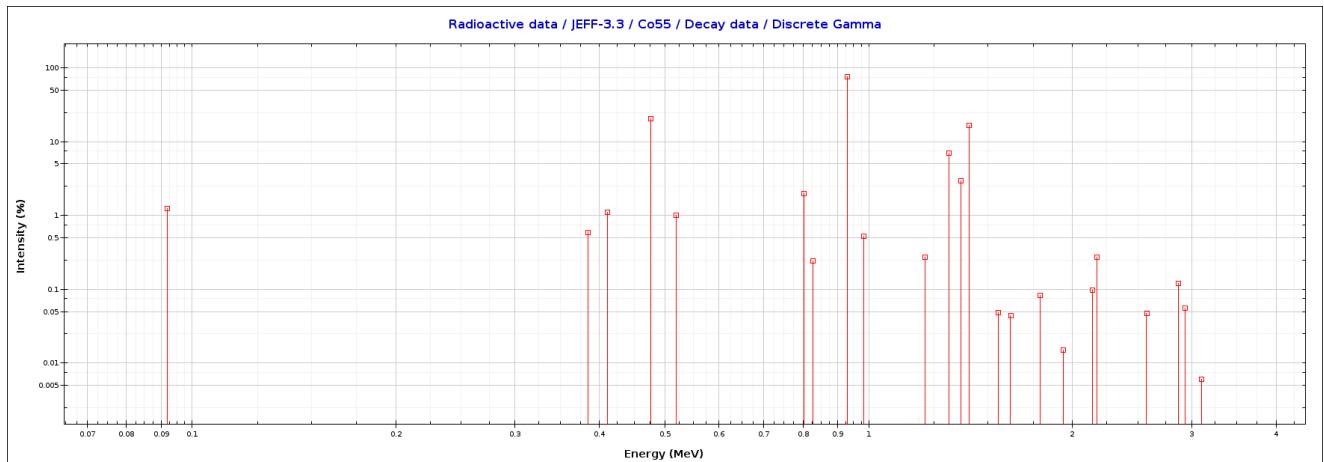


Figure 8.2: Cobalt-55 Gamma Intensity Plot

The stable isotope Iron-54 makes up almost 6% of natural iron, and one reaction possibility with a proton results in Cobalt-55 and a gamma, $^{54}_{26}Fe(p, \gamma)^{55}_{27}Co$. The most abundant isotope in natural Iron also has a route to transmute to Cobalt-55 through the capture of a proton and the loss of two neutrons, $^{56}_{26}Fe(p, 2n)^{55}_{27}Co$.

Due to the short half-life and intensity of the 931KeV gamma, this peak was measured in particular after irradiating the iron target. It was also estimated manually using the TENDL-2009 and TENDL-2019 cross section databases.

8.6 Cyclotron Beam Line - Proton Irradiated Iron

A 0.5mm thick pure iron target was irradiated for five minutes at a current of 0.5 microamps with 36MeV protons. The proton beam formed a $64mm^2$ square that was taken from the cyclotron through one of the beam lines, with the target perpendicular to the beam.

The target was left to cool for several days before being safe to handle. The radioactivity was measured using a high purity germanium detector which was calibrated to detect approximately 4 out of every 100 gammas emitted.

```

1 RANGE: 617 = 911.51keV to 649 = 958.79keV
2 AREA : Gross = 2127168 Net = 1342565 +/- 2207
3 CENTROID: 631.74 = 933.29keV
4 SHAPE: FWHM = 6.82 FW(1/5)M = 10.06
5 ID: Bi-214 at 934.05keV
6 Corrected Rate = 35349.26 +/- 58.11 cA

```

Listing 8.1: Maestro 931KeV Peak Measurement

Correcting the measured count for the peak, to account for the geometry and detector, the count rate for 931KeV gamma rays from Cobalt-55 was measured at $4.43 \times 10^4 \pm 1.05 \times 10^3$ counts per second.

8.7 SRIM Proton Irradiated Iron

The trajectory data used by the activity code(s), and also to estimate the Co-55 activity, was provided by SRIM. An input configuration was set with Iron as the target material and protons as the projectile ion. The sample depth was set to at least the target thickness, 0.5mm. The activity code truncates the data to fit the target thickness, so a much larger depth, say 1m, could be set in SRIM to ensure all the ions stop within the simulated target, and this data file may be reused for targets of varying thicknesses.

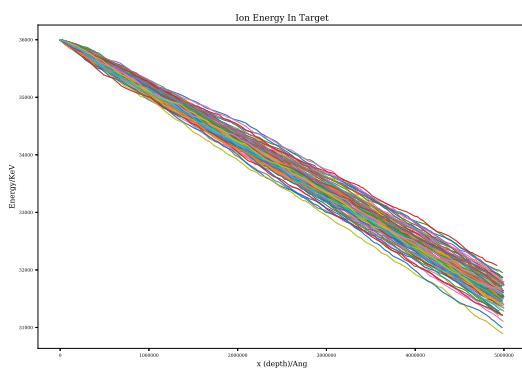


Figure 8.3: Energy vs target depth - protons through a 0.5mm thick iron target

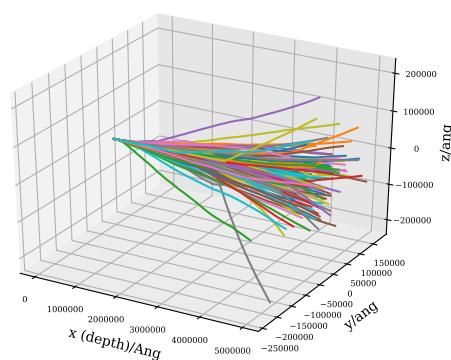


Figure 8.4: Ion trajectory - 0.5mm thick iron target

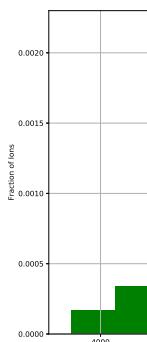


Figure 8.5:

The ions travelling through the thin 0.5mm target lose between 4MeV and 5MeV as they pass, and they do so quite smoothly. The majority of the energy is also lost smoothly through electronic stopping. As the thickness of the iron is increased, all the energy is lost to the iron target, and occurs within a shorter range in the remaining quarter of a millimetre depth of the iron target.

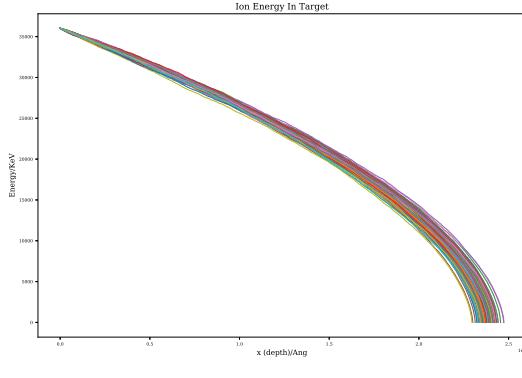


Figure 8.6: Energy vs target depth - protons in iron target (1m thick)

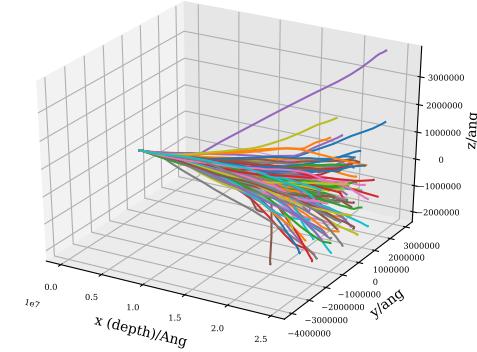


Figure 8.7: Ion trajectory - iron target (1m thick)

Figure 8.8:

8.8 Estimate of Activity Using TENDL Data

There are over ten TENDL libraries available, and the available data and the range of data points isn't consistent. The earlier 2009 Library, used in the first version of the Activity code, covers a range up to 200MeV for Iron cross sections, but the more recent 2019 version used in the second version of the activity code only covers up to 30MeV for Iron.

The 2019 data file ranges up to 30MeV for the $^{54}_{26}Fe(p, \gamma)^{55}_{27}Co$ and $^{56}_{26}Fe(p, 2n)^{55}_{27}Co$ reactions. As the data cuts off at 30MeV, the cross section data for a 28-30MeV proton is used in the estimation.

Parameter	$^{54}_{26}Fe(p, \gamma)^{55}_{27}Co$	$^{56}_{26}Fe(p, 2n)^{55}_{27}Co$
ND/atoms	4.6×10^{27}	7.4×10^{28}
J/amps	0.5×10^{-7}	0.5×10^{-7}
Q/C	1.60×10^{-19}	1.60×10^{-19}
d/m	5.0×10^{-4}	5.0×10^{-4}
sigma/barns	2.0×10^{-4}	2.0×10^{-2}
irradiation time/s	3.0×10^2	3.0×10^2
Co55 t-half/s	6.3×10^4	6.3×10^4
Co55 λ	1.1×10^{-5}	1.1×10^{-5}
931KeV Gamma Intensity	0.75	0.75
Reaction Rate/atoms per second	1.4×10^5	2.3×10^8
N(Co55 at 300s)	4.3×10^7	6.9×10^{10}
Activity(Co55 at 300s)	4.8×10^2	7.5×10^5
N(Co55 at 3 days)	2.5×10^6	4.0×10^9
Activity(Co55 at 3 days)	2.8×10^1	4.4×10^4
931KeV Gamma	2.1×10^1	3.3×10^4

Table 8.4: Estimation using 28-30MeV protons and the TENDL-2019 database

The 2009 data file ranges from a fraction of an eV to 200MeV for the $^{54}_{26}Fe(p, \gamma)^{55}_{27}Co$ and $^{56}_{26}Fe(p, 2n)^{55}_{27}Co$ reactions, and the average cross section is taken for each reaction based on a 36MeV proton that loses up to 5MeV travelling through the 0.5mm thick Iron target.

The largest contribution to the 931KeV gamma is due to the $^{56}_{26}Fe(p, 2n)^{55}_{27}Co$ reaction, and the overall activity is predicted to be 33,000Bq using 28-30MeV protons with the TENDL-2019 data file, and 49,000Bq using 36MeV protons with the the TENDL-2009 data file.

Parameter	$^{54}_{26}Fe(p, \gamma)^{55}_{27}Co$	$^{56}_{26}Fe(p, 2n)^{55}_{27}Co$
ND/atoms	4.6×10^{27}	7.4×10^{28}
J/amps	0.5×10^{-7}	0.5×10^{-7}
Q/C	1.60×10^{-19}	1.60×10^{-19}
d/m	5.0×10^{-4}	5.0×10^{-4}
sigma/barns	1.5×10^{-4}	3.0×10^{-2}
irradiation time/s	3.0×10^2	3.0×10^2
Co55 t-half/s	6.3×10^4	6.3×10^4
Co55 λ	1.1×10^{-5}	1.1×10^{-5}
931KeV Gamma Intensity	0.75	0.75
Reaction Rate/atoms per second	1.1×10^5	3.4×10^8
N(Co55 at 300s)	3.3×10^7	1.0×10^{11}
Activity(Co55 at 300s)	3.6×10^2	1.1×10^6
N(Co55 at 3 days)	1.9×10^6	6.0×10^9
Activity(Co55 at 3 days)	2.1×10^1	6.6×10^4
931KeV Gamma	1.6×10^1	4.9×10^4

Table 8.5: Estimation using 36MeV protons and the TENDL-2009 database

8.9 Activity V1 - Simulated Proton Irradiated Iron

The activity code with the derived activity equations were used to calculate the predicted radioactivity and predicted gamma lines.

Listing 8.2: Activity V1 Results 36MeV Protons and Iron

```

1 Fe36MeV.in
2 User Input Sim Details:
3 Beam Duration/s:          0.3000000000D+03
4 Beam Current/uA:          0.5000000000D+00
5 Beam Energy/MeV:           0.3600000000D+02
6 Beam Area/mm2:            0.1000000000D+03
7 Target Thickness/Angstrom: 0.5000000000D+07
8 Target Density/kgm-3:      0.8000000000D+04
9 Activity Measurement Time/s: 0.2600000000D+06
10
11 Most Active Isotopes     1.0907
12   Symbol   Z    A    M      Activity/Bq
13   CO       27   55   0      0.5942036078E+05
14   MN       25   52   0      0.4510767081E+05
15   CR       24   51   0      0.1444370965E+05
16   FE       26   55   0      0.1280863433E+05
17   MN       25   54   0      0.9738993802E+04
18 Total Activity/Bq:      0.1514219906E+06    1.1291
19 Run time: 1.1292
20
21 Total Activity/Bq:      0.1514219906E+06
22 Total Gamma Power/eV/s: 0.2559334931E+12
23 Total Gamma Power/Watts: 0.4100506435E-07
24 Absorbed Dose*/Grays/s: 0.4078849177E-10
25 Absorbed Dose*/Grays/hr: 0.1468385704E-06
26 Fraction of annual dosage if exposed for 1 hr: 0.1468385634E-03
27
28 Absorbed dose, assumes all energy absorbed, 80kg human, 1m from point-target, 1m surface area exposed to
   irradiation.
29
30 Dose Limits:
31 employees 18+ 20 millisieverts/year
32 trainees 18+ 6 millisieverts/year

```

```

33 public and under 18s 1 millisieverts/year
34 public and under 18s millisieverts averaged per hour: 0.1140771128E-03
35 Dose averaged over area of skin not exceeding 1cm2
36 Source: http://www.hse.gov.uk/radiation/ionising/doses/

```

The full output file and details of all isotopes and their activities are in Appendix E.

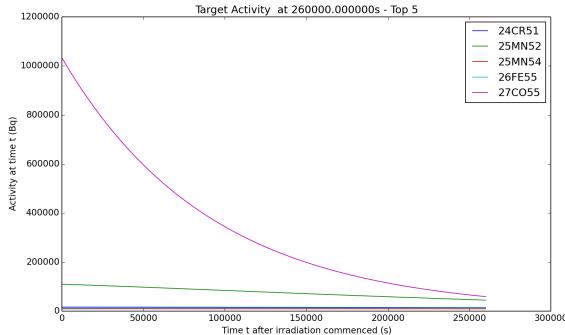


Figure 8.9: Isotopes with the top 5 radioactivity plotted over approximately 3 days

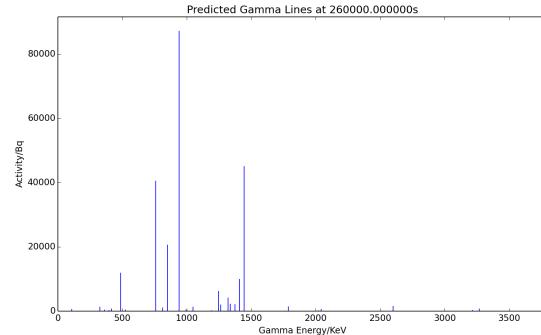


Figure 8.10: Predicted gamma lines approximately 3 days after irradiation

The code predicts a final 931KeV gamma activity of over 59,000 Bq and highlights several other isotopes of interest, including Mn-52, Cr-51, Fe-55 and Mn-54, all of which have an activity of approximately 10,000 Bq or above. After 3 days of cooling, the target is predicted to dose an 80KG human, 1 meter away from the target, with 4.0×10^{-11} Gy/s which is less than the limit for a member of the public (6.34×10^{-10} Si/s).

8.10 Activity V2 - Simulated Proton Irradiated Iron

Whilst the TENDL-2009 database does extend to several hundred MeV, the latest database does not (at least, for the isotopes that were of interest). This issue will be discussed in more detail in the final two chapters but, in short, I would like to give the user the option of using multiple data files as well as cross section data produced by Talys[104] to cover proton energies up to at least 100MeV.

If a calculation was set with a 36MeV beam and trajectory data for 36MeV protons, with the current TENDL-2019 database used by Activity V2, it would return zero activity as the database does not extend high enough. As a result, a lower energy (28MeV) beam was used for the calculation. The lower energy protons do have a higher reaction cross section, and the activities computed are higher than measured and calculated by Activity V1.

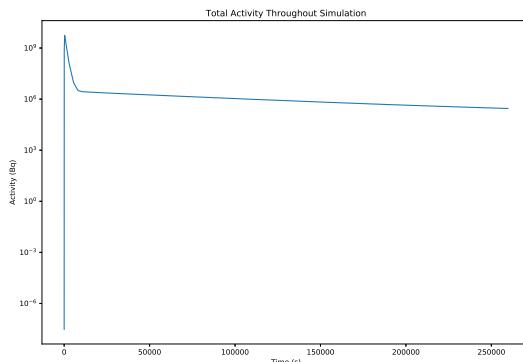


Figure 8.11: Total activity over time of the proton irradiated iron target

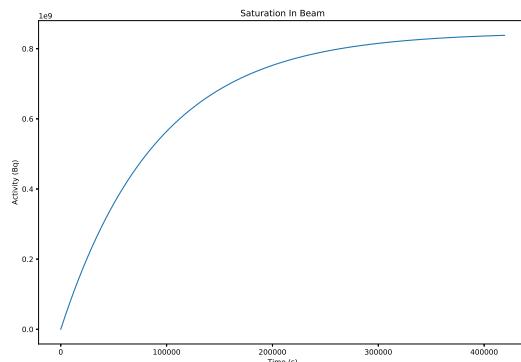


Figure 8.12: Saturation curve for Co55 with the current beam settings

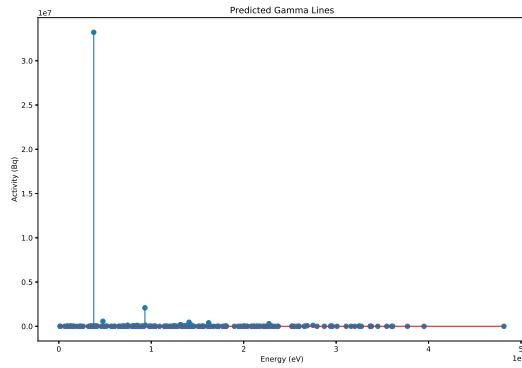


Figure 8.13: Energy vs target depth - protons in iron target thick enough to stop all ions

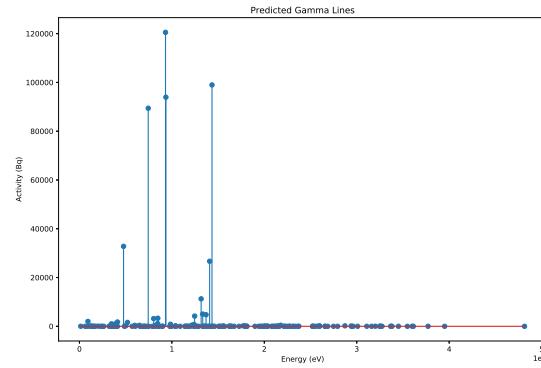


Figure 8.14: Ion trajectory - iron target thick enough to stop all ions

Listing 8.3: Activity V2 Results 28MeV Protons and Iron

```

1 3.209E-03 ### Start
2 4.769E-01 ### Loading isotope tendl xs
3 5.289E+00 ### Load complete
4 5.292E+00 ### Load EXYZ.txt
5 5.500E+00 ### Run Simulation
6 5.502E+00 ### Prep sim1
7 1.424E+01 #####
8 1.424E+01 #### Number Density
9 1.424E+01 #####
10 1.425E+01 ### Fe54      5.095579876290546e+27
11 1.425E+01 ### Fe56      7.713195369931147e+28
12 1.425E+01 ### Fe57      1.750086860505843e+27
13 1.425E+01 ### Fe58      2.2920662292036265e+26
14 2.136E+01 #####
15 2.137E+01 ### End of Beam Tally
16 2.137E+01 #####
17 2.141E+01 ### 27055     Co55      253567495453.0          2785060.44487
18 3.152E+01 #####
19 3.152E+01 ### End of Sim Tally
20 3.153E+01 #####
21 3.158E+01 ### 27055     Co55      14632033320.7          160711.045225
22 4.344E+01 #####
23 4.344E+01 ### Gamma Dose - Beam End
24 4.344E+01 ### Activity/Bq           82822873.7128
25 4.345E+01 ### Power eV/s            1.86841008076e+16
26 4.345E+01 ### Power J/s             0.00299352926319
27 4.345E+01 ### Dose Gy/s            2.97685885361e-06
28 4.345E+01 ### Dose Gy/hr           0.010716691873
29 4.345E+01 ### Percentage of annual dose/hr 9394.25237015
30 4.346E+01 ### Gamma Dose - Sim End
31 4.346E+01 ### Activity/Bq           283043.751191
32 4.346E+01 ### Power eV/s            5.1310279645e+14
33 4.346E+01 ### Power J/s             8.22083038417e-05
34 4.347E+01 ### Dose Gy/s            1.96294899335e-07
35 4.347E+01 ### Dose Gy/hr           0.000294301803729
36 4.347E+01 ### Percentage of annual dose/hr 257.984968681

```

Listing 8.4: Activity V1 Results 36MeV Protons and Iron

1	25052	Mn52	1246250000.0	0.0423	98986.0848283	4187.11138824
2	27055	Co55	1369800000.0	0.029499975	160711.045225	4740.97181637
3	25052	Mn52	1333620000.0	0.0507	98986.0848283	5018.5945008
4	27055	Co55	1316500000.0	0.070100025	160711.045225	11265.8482881

5	27055	Co55	1408500000.0	0.16599975	160711.045225	26677.9933297
6	27055	Co55	477200000.0	0.204	160711.045225	32785.053226
7	25052	Mn52	744214000.0	0.9034	98986.0848283	89424.0290339
8	25052	Mn52	935520000.0	0.949	98986.0848283	93937.7945021
9	25052	Mn52	1434050000.0	0.99987	98986.0848283	98973.2166373
10	27055	Co55	931300000.0	0.75	160711.045225	120533.283919

Due to the limitation of the TENDL-2019 datafile, the proton energy was set to 28MeV. The code predicted a similar Co55 931KeV gamma activity of 160,000 Bq. The predicted dose is much higher than that predicted by the first activity code, and this may be in part due to a much more comprehensive data file, JEFF 3.1.1, being used in the second version of the code.

The saturation activity is also plotted by the code for Cobalt-55. The experiment consisted of just five minutes within the beam and, with the current beam settings, the target would not be fully saturated until over a week in the beam. This would result in a dangerously radioactive sample.

8.11 Comparing Co-55 Peaks

Whilst there are a number of peaks predicted by the activity codes, the primary peak calibrated for and measured was of the 931KeV peak for Co-55.

Co-55 Gammas/second at approximately 3 days	
Experimental (36MeV)	$4.43 \times 10^4 + / - 1.05 \times 10^3$
Estimate (28MeV TENDL-2019)	3.28×10^4
Estimate (36MeV TENDL-2009)	4.92×10^4
Activity V1 (36MeV)	5.94×10^4
Activity V2 (28MeV)	1.21×10^5

Table 8.6: Comparison of proton irradiated iron Co55 931KeV gamma rates

The first version of the activity code is in better agreement with the experimental value, but this may be due to the data file used by the second version of the code not extending past 30MeV.

8.12 Predicted Activity for 100DPA Irradiated Iron

Generation IV reactors are required to withstand 100-150DPA throughout the lifetime of the plant, which may span 30 years or more. The Scanditronix cyclotron is capable of creating a 60 microamp beam and, due to the small area (and volume of material) this may be concentrated on, it is able to create high damage rates over a shorter period of time.

The Activity V2 code was used to calculate how radioactive a 100DPA iron sample would be after irradiation at the maximum current setting of 60 microamps. The simulated target is 0.5mm thick, pure iron and the beam is protons and has an area of 64mm^2 . The projectile fluence is 3.75×10^{14} protons per second, and the number of atoms within the volume of iron are 2.56×10^{21} . Five different energy settings were used (5MeV, 10MeV, 15MeV, 20MeV, 25MeV) and the vacancies per ion as well as the ion trajectories were generated by SRIM.

The doses are given as a multiple of the safe limit for the public. This is 1 millisievert per year, or approximately 1.14×10^{-4} millisieverts per hour.

It is clear to see from the predicted activities in table ?? that the higher energy beams would take longer to reach 100DPA, as they pass through the target leaving at a high energy, whilst also irradiating the target several orders of magnitude more than 5MeV and 10MeV. Detailed plots resulting from this Activity V2 simulation are in appendix ??.

	5MeV	10MeV	15MeV	20MeV	25MeV
VPI	38.5	58.8	51.0	23.7	20.2
DPA/year	178	272	236	109	93
Days for 100 DPA	205	134	155	333	391
Dose	23	234	4970	286,000	1,020,000
Dose (1 week of cooling)	18	192	4530	71,400	208,000

Table 8.7: Comparison of proton irradiated iron up to 100DPA at a range of proton energies

Chapter 9

Results: DFT Reference Database

A set of configurations of crystals are created, and DFT is used to computer the energy, forces and stresses on these crystals. FCC iron does not exist at normal room temperature and pressure, without the addition of a austenite stabiliser. A set of DFT calculations are used to compute the bulk modulus, lattice parameter and cohesive energy of pure Iron FCC. This data will be used to derive an Iron-Palladium potential.

9.1 Introduction

While computer technology has advanced almost in accordance with Moore's law, the calculations involved to approximately simulate simple structures with Quantum Mechanics are barely accessible. Ideally, crystal structures with several hundred atoms, containing Iron, Chromium, Nickel and Palladium would have been used to derive a potential describing all four elements.

The computer used did not have the resources required for such calculations, and so the model was simplified to Iron-Palladium. The ferromagentism and antiferromagnetism of Iron and Chromium were explored using collinear spin DFT calculations.

9.2 QECONVERGE Python Code

9.2.1 Purpose of Code

The process of converging cutoff parameters is one that may be automated. A program was developed in Python to automatically converge the ecutwfc and ecutrho values within a specified threshold.

The program reads an input file into memory, and this contains the settings for the convergence run. A template PWscf input file is also loaded, and this will have any required PWscf settings, such as the pseudopotentials, atom species and so on.

Once these files have been loaded, the first stage of the program runs to determine the ecutwfc and ecutrho values that are within the force and energy convergence thresholds. The program creates a crystal based on the settings, for example an FCC, BCC or SC crystal supercell. The exact atomic positions are likely to be the optimal, relaxed, positions and so the overall forces on each atom will be zero. The program randomly varies the atomic positions, and this results in a configuration with forces between the atoms.

The wfc value starts with the user defined ecutwfc starting value, and this is increased until the change in energy and force both converge to within the specified thresholds. The ecutrho value is increased simultaneously, and

is four times the ecutwfc value. In the second stage, the ecutrho value is decreased and the ecutwfc value is held constant while the energy and force remain within their respective convergence thresholds. Finally, the program attempts to decrease the ecutwfc value further.

Following the energy and charge density cutoff, the program then runs a number of calculations in order to produce a number of colour density plots of energy cutoff vs density cutoff, plotting the convergence of total energy and total force. This allows the user to visualise the convergence surface and select cutoff values manually.

The final part of the program is to help the user decide upon degauss and k-point values. The converged ecutwfc and ecutrho values from the first part of the code are used, and the k-point start, end and increment amounts are set by the user, as is a list of smearing degauss values. 2D colour plots of force and energy convergence as smearing changes and k-point value changes are prepared and saved so the user may study these to determine the best combination for their application.

The randomised atom positions use a seed that may be set in the input file. This means that, although they are pseudo-rng generated, they are repeatable given the same random seed. Every successful PWscf calculation is logged and saved, and if the program runs multiple times, the cached output files are used rather than recalculating every input file.

9.2.2 Source Code and Instructions

The source code and instructions on how to use the program are available to download from GitHub.

<https://github.com/BenPalmer1983/qeconverge>

9.3 Convergence of Key DFT Settings and Testing

9.3.1 Pseudo-potential

Prior to generating a database of atom configurations, calculated forces, stresses and energies, key DFT settings needed to be selected and tested. The Python code Qeconverge was written and used to converge the ecutwfc, ecutrho, k-points and degauss smearing values for each of the pseudopotentials required.

As discussed in previous chapters, the GGA type pseudo-potential more reliably calculated the bulk properties of a material when compared to LDA/LSDA, and a copy of the earlier table is given in table 9.1.

	Element	a/angs	B ₀ /GPa
Experimental	Na	4.29[89]	6.3[89]
Na.pz-spn-kjpaw_psl.1.0.0	Na	4.06	8.72
Na.pbe-spn-kjpaw_psl.1.0.0	Na	4.20	7.67
Na.pbesol-spn-kjpaw_psl.1.0.0	Na	4.17	7.50
Experimental	Al	4.05[90]	76[90]
Al.pz-nl-kjpaw_psl.1.0.0	Al	3.98	78.6
Al.pz-n-kjpaw_psl.1.0.0	Al	3.98	78.6
Al.pbe-nl-kjpaw_psl.1.0.0	Al	4.04	91.3
Al.pbe-n-kjpaw_psl.1.0.0	Al	4.04	75.0
Al.pbesol-nl-kjpaw_psl.1.0.0	Al	4.01	87.7
Al.pbesol-n-kjpaw_psl.1.0.0	Al	4.01	79.0

Table 9.1: Experimental vs LSDA vs GGA - the DFT values were computed with a PWscf[91] using a 2x2x2 cell, 7x7x7 kpoints and ecutwfc=50

The following pseudo-potential files were used for each element:

- Aluminium: a plane wave GGA potential Pd.pbe-spn-kjpaw_psl.1.0.0.UPF
- Chromium: a plane wave GGA potential Cr.pbe-spn-kjpaw_psl.1.0.0.UPF
- Iron: a plane wave GGA potential Fe.pbe-spn-kjpaw_psl.1.0.0.UPF
- Palladium: a plane wave GGA potential Pd.pbe-spn-kjpaw_psl.1.0.0.UPF

For each pseudo-potential, the `ecutwfc`, `ecutrho` and `kpoint` and `degauss` settings were converged. Ideally, the `k`-points settings would have been increased, but doing so would have been prohibitive due to the amount of time taken to perform each calculation.

By the time calculations were being executed for a 32 atom supercell, one node of 20 processor cores would run for several days up to a week or more, requiring the calculation to be submitted to continue multiple times. Aluminium is also considered in the results section, as it is a simpler atom to run trial calculations with, and to compare to experimental results.

9.3.2 Collinear Spin Calculations

The relaxed crystal lattice parameters were calculated for iron, chromium and palladium. Chromium will not be used in the Fe-Pd fitting calculations. However, there is a large increase in computation time between non-spin and collinear spin calculations. The ferromagnetic structure of BCC iron, antiferromagnetic structure of FCC iron and BCC chromium were investigated.

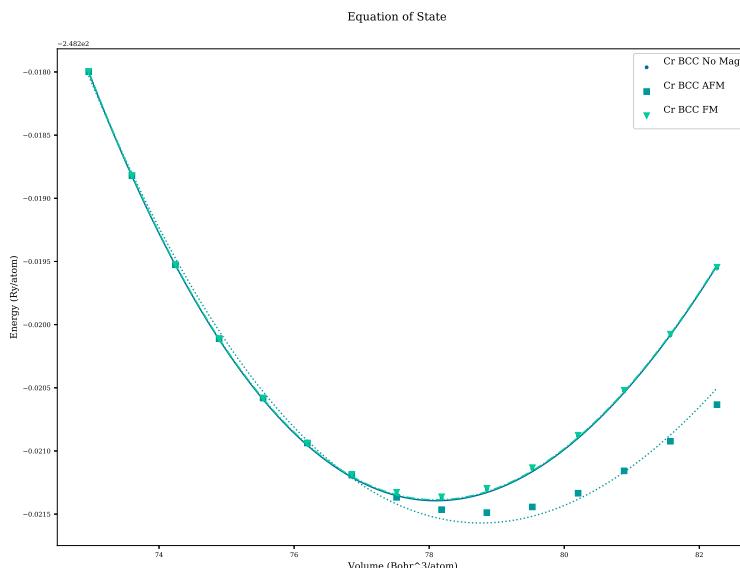


Figure 9.1: Equation of state fit through data points for Chromium BCC with no magnetism, ferromagnetic and anti-ferromagnetic configurations

Description	A0 (ang)	V0 (bohr ³)	B0 (GPA)	E0 (Ry) (DFT Only)
Exp.	2.91	83.23	160	-
AFM	2.86	78.77	217	-248.2216
FM	2.85	78.10	267	-248.2214
No Mag	2.85	78.102	267	-248.2214

Table 9.2: Chromium properties with and without collinear spin

The equation of state was calculated for Chromium BCC in three ways: (1) with magnetism switched off, (2) atoms set in a ferromagnetic configuration (collinear, all in the same direction), (3) atoms set in an antiferromagnetic configuration (collinear, atoms in the same cell with spin in opposing directions).

The DFT calculated values for the bulk modulus were larger than expected, but the antiferromagnetic calculation was much closer to the experimental value. The E0 values do not reflect the actual energies, but show the relative calculated values. The antiferromagnetic configuration gives the lowest, optimum, energy.

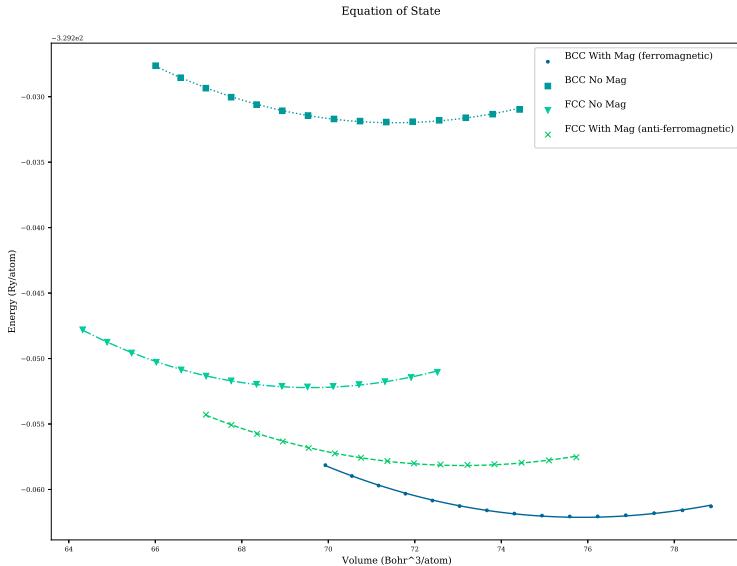


Figure 9.2: Equation of state fit through data points for Iron BCC (no magnetism, ferromagnetic) and Iron FCC (no magnetism and antiferromagnetic)

Description	A0 (ang)	V0 (bohr ³)	B0 (GPA)	E0 (Ry) (DFT Only)
Exp.	2.86	79.01	170	-
No Mag	2.77	71.53	286	-329.23
FM	2.82	75.9	239	-329.26
AFM	(failed)	(failed)	(failed)	(failed)

Table 9.3: Iron properties with and without collinear spin

In the case of Iron, both BCC and FCC configurations were used. The antiferromagnetic configurations for both BCC and FCC failed in that the SCF calculations failed to converge. As expected, the optimal configuration was BCC with ferromagnetism, and second to this was FCC with ferromagnetism.

Despite the increase in computational time, and other demands on resources such as scratch space and RAM per node, it was clear that collinear spin calculations were required for iron. The lattice parameter value was within 2% of the experimental value with the BCC structure set in a ferromagnetic state. Whilst the bulk modulus is only within 41

9.3.3 Convergence Results

The QE Converge program was used to suggest the parameters to use, given the convergence threshold of $1.0^{-5} RY/Bohr$ for forces and $1.0^{-6} RY$ for energy.

Element	Pseudopotential	Ecutwfc	Ecutrho	K-points	Degauss
Al	Al PBE KJPAW	50	200	11	0.04
Fe	FE PBE KJPAW	71	431	9	0.04
Pd	PD PBE KJPAW	71	431	9	0.04

Table 9.4: DFT Settings - pseudo-potentials, ecutwfc, ecutrho, k-points, smearing

These settings in table 9.4 were used throughout the remainder of the work. The maximum values were selected from Iron and Palladium as they were to be used for calculations of only Iron, only Palladium and an alloy of the two.

The other settings used in PWscf throughout are listed in table 9.5. Several of these values were arrived at by trial and error. It was recommended by the authors of Quantum Espresso to adjust parameters such as the mixing beta if a calculation fails to converge. The mixing mode was also changed from time to time, but overall the plain mode seemed to work best in most cases.

etot_conv_thr	0.0001
forc_conv_thr	0.001
conv_thr	1.0D-6
diagonalization	david
mixing_beta	0.1
mixing_mode	plain

Table 9.5: DFT Settings - other settings

The choices of parameters are supported by the convergence plots produced by the QECONVERGE code in figures I.1 and I.2 (appendix I).

9.4 Preliminary Calculations

9.4.1 Relaxed Crystal Calculations

The cohesive energy is an important value in relation to the interatomic potential. The energies calculated by DFT code depend on many factors including the energy under which plane-wave are kept, the degauss value, k-point settings and the SCF convergence parameters. What is more important in the DFT calculations is the difference in energy between calculations. As the cohesive energy is known, a DFT calculations can be performed for each species to determine the relaxed energy. This value may then be used to calibrate other DFT calculations, so they are given relative to the energy of each atom spaced infinitely far from one another, with a binding energy of 0ev.

Measurement	Al FCC	Fe BCC	Fe FCC	Pd FCC
Element	Aluminium	Iron	Iron	Palladium
Structure	FCC	BCC	FCC	FCC
Ecutwfc (Ry)	50	71	71	71
Ecutrho	200	430	430	430
K-points	11 11 11	9 9 9	9 9 9	9 9 9
Smearing (Ry)	0.04	0.04	0.04	0.04
NSpin	0	2	2	0
Starting Magnetism	None	FM	AFM	None
No. Atoms	32	16	32	32
Energy (Ry)	-1264.08749654	-5268.18846365	-10536.25040753	-16384.95803030
Energy/Atom (Ry)	-39.502734267	-329.261778978	-329.257825235	-512.029938447
Energy/Atom (eV)	-537.462275214	-4479.836349416	-4479.782555982	-6966.524743211
Known Cohesive Energy (eV)	-3.36	-4.316	-4.26 (Calculated)	-3.91
Adjustment/atom (eV)	534.102275214	4475.520349416	4475.520349416	6962.614743211
Relaxed a_0 (Bohr)	15.265	10.594	12.937	14.845
U_{xx}	1.000	1.000	1.000	1.000
U_{yy}	1.000	1.000	1.054	1.000
U_{zz}	1.000	1.000	1.000	1.000

Table 9.6: Relaxed energies calculated in Quantum Espresso

9.4.2 Iron-Palladium Atom Positions

The stainless steel being studied is to be doped with 1% or less of a platinum group metal. Ideally, a 256 4x4x4 FCC supercell would be used, with 2 or 3 randomly placed Palladium atoms, and the remainder Iron. This calculation would require more time and memory than was available, especially for repeat calculations, and a compromise of 1 Palladium and 31 Iron was used instead.

Listing 9.1: Relaxed Iron Palladium Configuration

```

CELL_PARAMETERS (alat= 13.017452732)
 1.0000000000  0.0000000000  0.0000000000
 0.0000000000  1.054129443  0.0000000000
 0.0000000000  0.0000000000  0.999992919

ATOMIC_POSITIONS (crystal)
Pd    0.000000000  0.000000000  0.000000000
Fe2   0.255184314  0.255092166  0.000000000
Fe3   0.257059507  -0.000000000  0.257051795
Fe4   0.000000000  0.255082799  0.255177990
Fe1   0.000000000  0.000000000  0.500000000
Fe2   0.250476117  0.250764583  0.500000000
Fe3   0.257059507  0.000000000  0.742947204
Fe4   0.000000000  0.255082799  0.744821008
Fe1   0.000000000  0.500000095  0.000000000
Fe2   0.255184314  0.744908119  0.000000000
Fe3   0.250799614  0.500000095  0.250799503
Fe4   0.000000000  0.744917486  0.255177990
Fe1   0.000000000  0.500000095  0.500000000
Fe2   0.250476117  0.749235702  0.500000000
Fe3   0.250799614  0.500000095  0.749200497
Fe4   0.000000000  0.744917486  0.744821008
Fe1   0.499999800  0.000000000  0.000000000
Fe2   0.744815185  0.255092166  0.000000000
Fe3   0.742940093  0.000000000  0.257051795
Fe4   0.499999800  0.250771574  0.250467682
Fe1   0.499999800  0.000000000  0.500000000
Fe2   0.749523382  0.250764583  0.500000000

```

```

Fe3    0.742940093 -0.000000000 0.742947204
Fe4    0.499999800  0.250771574  0.749531317
Fe1    0.499999800  0.500000095  0.000000000
Fe2    0.744815185  0.744908119  0.000000000
Fe3    0.749199985  0.500000095  0.250799503
Fe4    0.499999800  0.749228711  0.250467682
Fe1    0.499999800  0.500000095  0.500000000
Fe2    0.749523382  0.749235702  0.500000000
Fe3    0.749199985  0.500000095  0.749200497
Fe4    0.499999800  0.749228711  0.749531317
End final coordinates

```

The start configuration is FCT Iron with a slightly longer y axis than x or z, and the positions of the atoms and the structure were relaxed in Quantum Espresso.

9.5 QEEOS Python Code

9.5.1 Purpose of Code

A code was developed to automate the process of calculating the equation of state and elastic constants of a material using Quantum Espresso. It requires an input file and a starting PWscf input file.

- the input configuration is relaxed using the vc-relax option in PWscf
- configuration files are created to compute the equation of state, and PWscf is used to calculate the energies of these configurations
- to compute the elastic constants, nine distortions are applied to the relaxed configuration with the required number of steps for each strain applied to each distortion, and these are processed with PWscf
- once all DFT work has completed, the Birch-Murnaghan equation of state is fit to the first set of energies, and the nine orthorhombic elastic constants are fit to the results of the second set of energies

9.5.2 Source Code and Instructions

The source code and instructions on how to use the program are available to download from GitHub.

https://github.com/BenPalmer1983/qe_eos

9.6 Calculated Elastic Properties

The planewave cutoff and k-points were converged to within the required parameters for energy and force but it is known that, in general, LDA and GGA pseudopotentials under and over estimate the lattice parameters. The bulk modulus and elastic constants were calculated for FCC Aluminium and BCC Iron in order to compare to experimental values.

9.6.1 FCC Aluminium

The parameters used for the aluminium calculations were arrived at earlier using the convergence code. The remainder of the parameters are either default settings, or were selected following trial and error (for example, reducing the mixing beta to help achieve convergence).

Ecutwfc (Ry)	50
Ecutrho	200
Smearing (Ry)	0.04
K-points	11 11 11 1 1 1
Nspin	0 (Non-magnetic)
Pseudopotential	Al.pbe-nl-kjpaw_psl.1.0.0.UPF
etot_conv_thr	0.0001
forc_conv_thr	0.001
conv_thr	1.0D-6
diagonalization	david
mixing_beta	0.1
mixing_mode	plain

Table 9.7: Aluminium DFT settings

	Al Experimental	Al DFT (this work)
Structure	Face Centered Cubic	Face Centered Cubic
a_0 (Angs)	4.05 Angstrom	4.04 Angstrom
Nearest Neighbour	2.86 Angstrom	2.86 Angstrom
Basis vectors	$\begin{bmatrix} 1.0 & 0.0 & 0.0 \\ 0.0 & 1.0 & 0.0 \\ 0.0 & 0.0 & 1.0 \end{bmatrix}$	$\begin{bmatrix} 1.0 & 0.0 & 0.0 \\ 0.0 & 1.0 & 0.0 \\ 0.0 & 0.0 & 1.0 \end{bmatrix}$
E_{coh} (eV)	-3.36 eV	Not Calculated
Bulk Modulus B_0 (GPA)	76	77.6
Bulk Modulus $B_{0,r}$ (GPA)	-	75.4
Bulk Modulus $B_{0,g}$ (GPA)	-	75.4
Young's modulus E (GPA)	70	81.4
Shear Modulus G (GPA)	26	30.8
Poisson Ratio ν	0.35	0.32
Elastic Constants (GPA)	$\begin{bmatrix} 114 & 62 & 62 & 0 & 0 & 0 \\ 62 & 114 & 62 & 0 & 0 & 0 \\ 62 & 62 & 114 & 0 & 0 & 0 \\ 0 & 0 & 0 & 32 & 0 & 0 \\ 0 & 0 & 0 & 0 & 32 & 0 \\ 0 & 0 & 0 & 0 & 0 & 32 \end{bmatrix}$	$\begin{bmatrix} 110.9 & 57.7 & 57.7 & 0 & 0 & 0 \\ 57.7 & 110.8 & 57.5 & 0 & 0 & 0 \\ 57.7 & 57.5 & 110.9 & 0 & 0 & 0 \\ 0 & 0 & 0 & 34.0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 34.0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 34.0 \end{bmatrix}$

Table 9.8: Aluminium: Experimental Values vs DFT Calculated Variables

9.6.2 BCC Iron

The equation of state and elastic constants were calculated for BCC Iron, with no-spin and collinear spin. The full calculated results are included in the appendix ??.

Ecutwfc (Ry)	71
Ecutrho	430
Smearing (Ry)	0.04
K-points	9 9 9 1 1 1
Nspin	0 (Non-magnetic) and 2 (Collinear Spin)
Pseudopotential	Fe.pbe-spn-kjpaw-psl.1.0.0.UPF
etot_conv_thr	0.0001
forc_conv_thr	0.001
conv_thr	1.0D-6
diagonalization	david
mixing_beta	0.1
mixing_mode	plain

Table 9.9: Iron DFT settings

	Fe Experimental	Fe DFT No Magnetism (this work)	Fe DFT Ferromagnetic (this work)
Structure	Body Centered Cubic	Body Centered Cubic	2.80 Angstrom
a_0 (Angs)	2.86 Angstrom[105]	2.75 Angstrom	2.42 Angstrom
Nearest Neighbour	2.48 Angstrom[105] $\begin{bmatrix} 1.0 & 0.0 & 0.0 \\ 0.0 & 1.0 & 0.0 \\ 0.0 & 0.0 & 1.0 \end{bmatrix}$	2.38 Angstrom $\begin{bmatrix} 1.0 & 0.0 & 0.0 \\ 0.0 & 1.0 & 0.0 \\ 0.0 & 0.0 & 1.0 \end{bmatrix}$	$\begin{bmatrix} 1.0 & 0.0 & 0.0 \\ 0.0 & 1.0 & 0.0 \\ 0.0 & 0.0 & 1.0 \end{bmatrix}$
Basis vectors			Not Calculated
E_{coh} (eV)	-4.32 [105]		
Bulk Modulus B_0 (GPA)	170	287.3	239.2
Bulk Modulus $B_{0,r}$ (GPA)	-	257.5	205.3
Bulk Modulus $B_{0,g}$ (GPA)	-	253.7	205.3
Young's modulus E (GPA)	211	-11960.7	211.1
Shear Modulus G (GPA)	82	-643.1	79.4
Poisson Ratio ν	0.29	8.29	0.33
Elastic Constants (GPA)	$\begin{bmatrix} 243 & 145 & 145 & 0 & 0 & 0 \\ 145 & 243 & 145 & 0 & 0 & 0 \\ 145 & 145 & 243 & 0 & 0 & 0 \\ 0 & 0 & 0 & 116 & 0 & 0 \\ 0 & 0 & 0 & 0 & 116 & 0 \\ 0 & 0 & 0 & 0 & 0 & 116 \end{bmatrix}$	$\begin{bmatrix} 63.0 & 329.2 & 315.3 & 0 & 0 & 0 \\ 329.2 & 176.6 & 316.4 & 0 & 0 & 0 \\ 315.3 & 316.5 & 121.7 & 0 & 0 & 0 \\ 0 & 0 & 0 & 176.9 & 0 & 0 \\ 0 & 0 & 0 & 0 & 180.8 & 0 \\ 0 & 0 & 0 & 0 & 0 & 180.0 \end{bmatrix}$	$\begin{bmatrix} 250.1 & 182.3 & 183.3 & 0 & 0 & 0 \\ 182.3 & 249.1 & 182.9 & 0 & 0 & 0 \\ 183.2 & 182.9 & 251.2 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 139.2 & 0 \\ 0 & 0 & 0 & 0 & 0 & 139.7 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$

Table 9.10: Iron: Experimental Values vs DFT Calculated Variables

The non-magnetic Iron calculation was particularly poor. Although the lattice parameter was within 5% of the experimental value, the calculated bulk modulus was significantly lower than the experimental value. With collinear spin enabled, the structure is stable. The lattice parameter is still slightly under the experimental, but is now within almost 2% of the value. The bulk modulus predicted by fitting the Birch-Murnaghan equation of state is more than 40 GPa.

9.6.3 FCC Palladium

Ecutwfc (Ry)	71
Ecutrho	430
Smearing (Ry)	0.04
K-points	9 9 9 1 1 1
Nspin	0 (Non-magnetic)
Pseudopotential	Pd.pbe-spn-kjpaw_psl.1.0.0.UPF
etot_conv_thr	0.0001
forc_conv_thr	0.001
conv_thr	1.0D-6
diagonalization	david
mixing_beta	0.1
mixing_mode	plain

Table 9.11: Iron DFT settings

	Palladium FCC	Palladium FCC (this work)
Structure	Face Centered Cubic	Face Centered Cubic
a_0 (Angs)	3.89	3.92
Nearest Neighbour	2.75	2.77
Basis vectors	$\begin{bmatrix} 1.0 & 0.0 & 0.0 \\ 0.0 & 1.0 & 0.0 \\ 0.0 & 0.0 & 1.0 \end{bmatrix}$	$\begin{bmatrix} 1.0 & 0.0 & 0.0 \\ 0.0 & 1.0 & 0.0 \\ 0.0 & 0.0 & 1.0 \end{bmatrix}$
E_{coh} (eV)	-3.91	not calculated
Bulk Modulus B_0 (GPA)	195.5	184.4
Bulk Modulus $B_{0,r}$ (GPA)	-	173.7
Bulk Modulus $B_{0,v}$ (GPA)	-	173.7
Young's modulus E (GPA)	121	153.1
Shear Modulus G (GPA)	44	56.6
Poisson Ratio ν	0.39	0.35
Elastic Constants (GPA)	$\begin{bmatrix} 234 & 176.1 & 176.1 & 0 & 0 & 0 \\ 176.1 & 234 & 176.1 & 0 & 0 & 0 \\ 176.1 & 176.1 & 234 & 0 & 0 & 0 \\ 0 & 0 & 0 & 71.2 & 0 & 0 \\ 0 & 0 & 0 & 0 & 71.2 & 0 \\ 0 & 0 & 0 & 0 & 0 & 71.2 \end{bmatrix}$	$\begin{bmatrix} 218.5 & 151.4 & 151.4 & 0 & 0 & 0 \\ 151.4 & 218.5 & 151.3 & 0 & 0 & 0 \\ 151.4 & 151.3 & 218.5 & 0 & 0 & 0 \\ 0 & 0 & 0 & 80.3 & 0 & 0 \\ 0 & 0 & 0 & 0 & 80.3 & 0 \\ 0 & 0 & 0 & 0 & 0 & 80.3 \end{bmatrix}$

Table 9.12: Palladium FCC: DFT Calculated Properties

9.6.4 FCC Iron

The gamma phase of pure iron does not exist for experimental measurements to be made, and the results here will be used to fit the FCC iron potential in chapter 10.

Ecutwfc (Ry)	71
Ecutrho	430
Smearing (Ry)	0.04
K-points	9 9 9 1 1 1
Nspin	2 (Collinear Spin)
Pseudopotential	Fe.pbe-spn-kjpaw_psl.1.0.0.UPF
etot_conv_thr	0.0001
forc_conv_thr	0.001
conv_thr	1.0D-6
diagonalization	david
mixing_beta	0.1
mixing_mode	plain

Table 9.13: Iron DFT settings

		Iron FCC (this work)					
Structure		Face Centered Cubic			Face Centered Cubic		
a_0 (Angs)	-				4.04 Angstrom		
Nearest Neighbour	-				2.86 Angstrom		
Basis vectors		$\begin{bmatrix} 1.0 & 0.0 & 0.0 \\ 0.0 & 1.0 & 0.0 \\ 0.0 & 0.0 & 1.0 \end{bmatrix}$			$\begin{bmatrix} 1.0 & 0.0 & 0.0 \\ 0.0 & 1.0 & 0.0 \\ 0.0 & 0.0 & 1.0 \end{bmatrix}$		
E_{coh} (eV)	-				-4.26		
Bulk Modulus B_0 (GPA)	-				226.1		
Bulk Modulus $B_{0,r}$ (GPA)	-				217.3		
Bulk Modulus $B_{0,v}$ (GPA)	-				226.6		
Young's modulus E (GPA)	-				356.8		
Shear Modulus G (GPA)	-				144.8		
Poisson Ratio ν	-				0.23		
Elastic Constants (GPA)	-				$\begin{bmatrix} 364.6 & 141.6 & 233.8 & 0 & 0 & 0 \\ 141.6 & 298.7 & 130.4 & 0 & 0 & 0 \\ 233.8 & 130.4 & 364.6 & 0 & 0 & 0 \\ 0 & 0 & 0 & 186.3 & 0 & 0 \\ 0 & 0 & 0 & 0 & 266.8 & 0 \\ 0 & 0 & 0 & 0 & 0 & 186.3 \end{bmatrix}$		

Table 9.14: Iron FCC: DFT Calculated Properties

9.7 QEFORFIT Python Code

9.7.1 Purpose of Code

A code was developed to automate the process of creating.

- the input configuration is relaxed using the vc-relax option in PWscf
- configuration files are created to compute the equation of state, and PWscf is used to calculate the energies of these configurations
- to compute the elastic constants, nine distortions are applied to the relaxed configuration with the required number of steps for each strain applied to each distortion, and these are processed with PWscf
- once all DFT work has completed, the Birch-Murnaghan equation of state is fit to the first set of energies, and the nine orthorhombic elastic constants are fit to the results of the second set of energies

9.7.2 Source Code and Instructions

The source code and instructions on how to use the program are available to download from GitHub.

https://github.com/BenPalmer1983/qe_eos

9.7.3 DFT Configuration Database

The QEFORFIT code was used to create files that contributed towards the DFT configuration database. The broken symmetry cause by the random perturbation of atoms in the configurations caused the calculation to take a long time to complete. The calculations were also very resource intensive, using up to 100GB of RAM and 20-40 cores for up to a day per configuration.

In addition to this, quite often the SCF convergence would either fail, or complete the maximum number of iterations (which was set high, at 100) without converging, which would also be considered a fail.

Due to the large amount of resources required and the regular failure to converge, there were a reduced number of configurations: 1 for Iron, 2 for Iron-Palladium, 10 for Palladium.

The Iron and Palladium PWscf output files produced by the QEEOS code, that were used to calculate the equation of state and elastic constants, were added to the database of files, giving a total of 154 configurations for the Iron-Palladium potential fitting.

Chapter 10

Results: Interatomic Potential Fitting

Chapter Summary

10.1 EAMPA Python Code

10.1.1 Purpose of Code

A code was developed to automate the process of calculating the equation of state and elastic constants of a material using Quantum Espresso. It requires an input file and a starting PWscf input file.

- the input configuration is relaxed using the vc-relax option in PWscf
- configuration files are created to compute the equation of state, and PWscf is used to calculate the energies of these configurations
- to compute the elastic constants, nine distortions are applied to the relaxed configuration with the required number of steps for each strain applied to each distortion, and these are processed with PWscf
- once all DFT work has completed, the Birch-Murnaghan equation of state is fit to the first set of energies, and the nine orthorhombic elastic constants are fit to the results of the second set of energies

10.1.2 Source Code and Instructions

The source code and instructions on how to use the program are available to download from GitHub.

https://github.com/BenPalmer1983/qe_eos

10.1.3 Modular Design

After several iterations, a python-fortran version was developed. The Python code was designed to handle the user input, output and the fitting process. The fortran modules were designed with a separate module for each task: energy/force/stress, bulkproperties, surface energy and defect calculations. The surface energy and defect module is incomplete. More modules may be easily added at a later date.

10.1.4 Energy, Force and Stress Calculations

10.1.5 Bulk Property Calculations

10.1.6 RSS - How Well The Potential Fits Experimental & DFT Data

10.1.7 Analytic Potential Fitting

Overview of the Potential Fitting Process

The input files are read in by python that then loops while changing the potential. The Fortran modules calculate the energy, force and stress of the loaded configurations as well as the bulk property calculations. To save on processing time, the neighbour lists for all configurations, including those used to calculate the equation of state and elastic constants, are only calculated once.

1. Read input files
2. Loads Fortran Module
3. Load potentials into Python and Fortran
4. Load configurations into Python and Fortran
5. Allocate
6. Creates neighbour list for all configurations
7. Creates eos and ec configurations and neighbour lists
8. Begin Fitting Loop in Python
9. Loop starts - loop through potential function parameters
 - (a) Update potential - create tabulated functions and load to fortran modules
 - (b) Loop starts - loop through configurations
 - i. Calculate energy, force, stress using pre-calculated neighbour lists
 - (c) End loop
 - (d) Loop starts - bulk property calculations
 - i. Calculate eos and ec values and bulk properties using pre-calculated neighbour lists
 - (e) End loop
 - (f) Calculate rss between these values and known/dft
 - (g) Save parameters if best
 - (h) Make new parameters
10. End loop
11. Output results

Function Types

All the potential functions in the latest version of the code are analytical. There is a full list in the appendix, and the code may be edited by other users to add more functions.

Most of the functions are basic mathematical functions that use built in operators and functions. Several are more complex, such as the cubic spline, quintic

Cubic and Quintic Splines

Several papers have previously used summed polynomial splines that employ the Heaviside-step function to cut off these functions at the desired points. The cubic term, $(r - r_i)^3$, ensures the function and its first and second derivatives equal zero and are continuous at the cutoff points.

$$V(r) = \sum_i^N a_i (r - r_i)^3 H(r_i - r)$$

where

$$H(x) = \begin{cases} 0 & x < 0 \\ 1 & x \geq 0 \end{cases}$$
(10.1)

This was implemented in Fortran so that an efficient vectorised version could be used by Python.

Listing 10.1: Cubic Spline

```

1 ! Two Band Modelling Fe-Cr Olsson, Wallenius
2 ! CUBIC SPLINE
3 ! sum (ai (r - ri)^3 H(ri - r)
4 ! SCALAR SUBROUTINE
5 SUBROUTINE cubic_spline(r, p, p_fixed, y)
6 !#####
7 ! p coefficients
8 ! pf r cutoffs
9 ! they must be the same size
10 IMPLICIT NONE
11 !#####
12 REAL(kind=DoubleReal), INTENT(IN) :: r
13 REAL(kind=DoubleReal), INTENT(IN) :: p(:)
14 REAL(kind=DoubleReal), INTENT(IN) :: p_fixed(:)
15 REAL(kind=DoubleReal), INTENT(OUT) :: y
16 !#####
17 REAL(kind=DoubleReal) :: H
18 INTEGER(kind=StandardInteger) :: n
19 !#####
20 y = 0.0D0
21 DO n = 1, SIZE(p,1)
22   ! Heaviside((p_fixed(n) - r)
23   IF((p_fixed(n) - r) < 0.0D0)THEN
24     H = 0.0D0
25   ELSE
26     H = 1.0D0
27   END IF
28   y = y + p(n) * (r - p_fixed(n))**3 * H
29 END DO
30 IF(SIZE(p,1) + 1 .EQ. SIZE(p_fixed,1))THEN
31   IF(r .GE. p_fixed(SIZE(p_fixed,1)))THEN
32     y = 0.0D0
33   ELSE
34     y = y * (r - p_fixed(SIZE(p_fixed,1)))**3
35   END IF
36 END IF
37 END SUBROUTINE cubic_spline
38
39 ! VECTOR SUBROUTINE
40 SUBROUTINE cubic_spline_v(r, p, p_fixed, y)

```

```

41 !#####
42 ! CUBIC SPLINE
43 IMPLICIT NONE
44 !#####
45 REAL(kind=DoubleReal), INTENT(IN) :: r(:)
46 REAL(kind=DoubleReal), INTENT(IN) :: p(:)
47 REAL(kind=DoubleReal), INTENT(IN) :: p_fixed(:)
48 REAL(kind=DoubleReal), INTENT(OUT) :: y(1:SIZE(r,1))
49 INTEGER(kind=StandardInteger) :: n
50 !#####
51 ! Loop through all the values in r(:), calculate and store in y(:)
52 DO n = 1, SIZE(r,1)
53   CALL cubic_spline(r(n), p, p_fixed, y(n))
54 END DO
55 END SUBROUTINE cubic_spline_v

```

10.1.8 Knot to Knot Splines

A Fortran module was created for the knot-to-knot spline function. It takes a set of knots and splines between the knots pair at a time using the x , y , y' values to construct a set of linear equations to solve.

$$\begin{aligned}
 c_0 + c_1 x_A + c_2 x_A^2 + c_3 x_A^3 &= f(x_A) \\
 0 + c_1 + 2c_2 x_A + 3c_3 x_A^2 &= f'(x_A) \\
 c_0 + c_1 x_B + c_2 x_B^2 + c_3 x_B^3 &= f(x_B) \\
 0 + c_1 + 2c_2 x_B + 3c_3 x_B^2 &= f'(x_B)
 \end{aligned} \tag{10.2}$$

$$\left[\begin{array}{cccc} 1 & x_A & x_A^2 & x_A^3 \\ 0 & 1 & 2x_A & 3x_A^2 \\ 1 & x_B & x_B^2 & x_B^3 \\ 0 & 1 & 2x_B & 3x_B^2 \end{array} \right] \left[\begin{array}{c} c_0 \\ c_1 \\ c_2 \\ c_3 \end{array} \right] = \left[\begin{array}{c} f(x_A) \\ f'(x_A) \\ f(x_B) \\ f'(x_B) \end{array} \right] \tag{10.3}$$

A subroutine was created to calculate the coefficients for the 3rd order or 5th order polynomial to spline the two knots. This subroutine and the remainder of the module are available to view and download on the github repository for this project.

Listing 10.2: Cubic Spline

```

1 SUBROUTINE spline_ab(spline_type, node_a, node_b, coeffs)
2 !#####
3 IMPLICIT NONE
4 !#####
5 INTEGER(kind=StandardInteger), INTENT(IN) :: spline_type
6 REAL(kind=DoubleReal), INTENT(IN) :: node_a(:)
7 REAL(kind=DoubleReal), INTENT(IN) :: node_b(:)
8 REAL(kind=DoubleReal), INTENT(OUT) :: coeffs(1:SIZE(node_a) + SIZE(node_b) - 2)
9 !#####
10 coeffs = 0.0D0
11 ! POLY3
12 ! Each node/knot has 3 values x, y, y'
13 IF(spline_type .EQ. 1 .AND. SIZE(node_a) .EQ. 3 .AND. SIZE(node_b) .EQ. 3) THEN

```

```

14     CALL spline_ab_poly(node_a, node_b, coeffs)
15 END IF
16 ! POLY5
17 ! Each node/knot has 4 values x, y, y', y'''
18 IF(spline_type .EQ. 2 .AND. SIZE(node_a) .EQ. 4 .AND. SIZE(node_b) .EQ. 4)THEN
19     CALL spline_ab_poly(node_a, node_b, coeffs)
20 END IF
21 END SUBROUTINE spline_ab
22
23
24 SUBROUTINE spline_ab_poly(node_a, node_b, coeffs)
25 !#####
26 IMPLICIT NONE
27 !#####
28 REAL(kind=DoubleReal), DIMENSION(:, INTENT(IN) :: node_a
29 REAL(kind=DoubleReal), DIMENSION(:, INTENT(IN) :: node_b
30 REAL(kind=DoubleReal), INTENT(OUT) :: coeffs(1:SIZE(node_a) + SIZE(node_b) - 2)
31 !#####
32 INTEGER(kind=StandardInteger) :: m_size = 0
33 INTEGER(kind=StandardInteger) :: m_half_size = 0
34 INTEGER(kind=StandardInteger) :: d_loop = 0
35 REAL(kind=DoubleReal) :: x(1:SIZE(node_a,1)+SIZE(node_b,1)-2,1:SIZE(node_a,1)+SIZE(node_b,1)-2)
36 REAL(kind=DoubleReal) :: y(1:SIZE(node_a,1)+SIZE(node_b,1)-2)
37 REAL(kind=DoubleReal) :: x_coeffs(1:(SIZE(node_a,1)+SIZE(node_b,1)-2))
38 REAL(kind=DoubleReal) :: x_exponent(1:(SIZE(node_a,1)+SIZE(node_b,1)-2))
39 INTEGER(kind=StandardInteger) :: n, row, col, a_row
40 !#####
41 ! Sizes
42 m_size = SIZE(node_a) + SIZE(node_b) - 2
43 m_half_size = m_size / 2
44 ! x exponents
45 DO n = 1, m_size
46     x_exponent(n) = 1.0D0 * (n - 1)
47 END DO
48 ! Coeffs
49 x_coeffs = 1
50 ! Make y matrix
51 row = 1
52 DO n=2,SIZE(node_a)
53     y(row) = node_a(n)
54     row = row + 1
55     y(row) = node_b(n)
56     row = row + 1
57 END DO
58 ! Make x matrix
59 row = 0
60 DO a_row = 1, m_half_size
61     ! NODE A
62     row = row + 1
63     DO col = 1, a_row -1
64         x(row, col) = 0.0D0
65     END DO
66     DO col = a_row, m_size
67         x(row, col) = x_coeffs(col) * node_a(1)**x_exponent(col)
68     END DO
69     ! NODE B
70     row = row + 1
71     DO col = 1, a_row -1
72         x(row, col) = 0.0D0
73     END DO
74     DO col = a_row, m_size

```

```

75      x(row, col) = x_coeffs(col) * node_b(1)**x_exponent(col)
76  END DO
77 ! UPDATE COEFFS AND EXPONENT
78 DO col = 1, m_size
79     x_coeffs(col) = x_coeffs(col) * x_exponent(col)
80 END DO
81 DO col = 1, m_size
82     IF(x_exponent(col) .GT. 0.0D0)THEN
83         x_exponent(col) = x_exponent(col) - 1.0D0
84     END IF
85 END DO
86 ! SOLVE
87 CALL sls_solve(x, y, coeffs)
88 !#####
89 !#####
90 END SUBROUTINE spline_ab_poly

```

Simulated Annealing

The simulated annealing algorithm was considered for optimisation, and may be implemented as an option in the future. The algorithm looks for better solutions and replaces the current value with better values. It also accepts worse solutions with a probability that decreases with "temperature". Similar to the way atoms find optimal positions by annealing and cooling, the algorithm starts with a high temperature and it reduces this temperature over time. It is used to search for the global optimum.

Listing 10.3: Simulated Annealing Algorithm

```

1 import numpy
2 class simanneal:
3
4     pbest = None
5     pnew = None
6     rssnew = None
7     pbest = None
8     rssbest = None
9
10    @staticmethod
11    def opt(p, p_var, frss, loops=1000, pfact=0.99, t_start=100, t_end=1.0, t_steps=10):
12
13        simanneal.pnew = p
14        simanneal.pbest = simanneal.pnew
15        simanneal.rssbest = frss(p)
16        simanneal.rssnew = simanneal.rssbest
17
18        t = t_start
19        tn = 0
20        while(t >= t_end):
21            for n in range(loops):
22                p = simanneal.pnew + pfact**(tn) * p_var * (0.5 - numpy.random.rand(len(simanneal.pnew)))
23                rss = frss(p)
24
25                # If better, accept
26                if(rss < simanneal.rssnew):
27                    simanneal.pnew = p
28                    simanneal.rssnew = rss
29                else:
30                    # If worse, accept only with a small probability
31                    if(numpy.random.uniform() < numpy.exp((simanneal.rssnew - rss) / t)):
32                        simanneal.pnew = p
33                        simanneal.rssnew = rss

```

```

34     if(rss < simanneal.rssbest):
35         simanneal.pbest = p
36         simanneal.rssbest = rss
37
38     # Update new with best
39     simanneal.pnew = simanneal.pbest
40     simanneal.rssnew = simanneal.rssbest
41
42     # Cool
43     t = t - (t_start - t_end) / (t_steps-1)
44     tn = tn + 1
45
46 return simanneal.pbest

```

Genetic Algorithm

The global optimum is searched for using a genetic algorithm. The user chooses the parameters for the algorithm and it attempts to search for the optimum parameters to fit the potential functions to the data.

The user settings include the following:

- population size
- "fresh" population size
- number of generations
- number of extinction events
- how often the top solutions are enhanced with linesearch and gradient descent

The subroutine that breeds the parameters randomly combines the parameters from two parent solutions to create two children parameter sets. The parents are either both from the existing pool of solutions, or a parent from the existing pool with a parent from a fresh set that changes with each generation.

Mixing Parameters from Parents to make Children

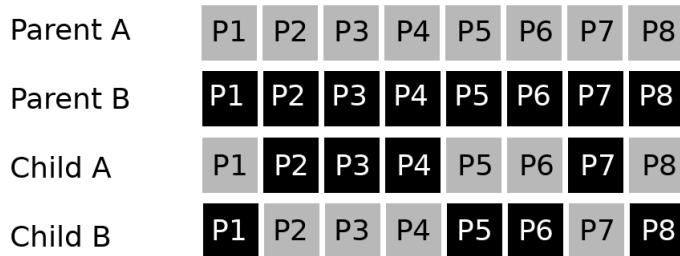


Figure 10.1: Parameter Breed Event

There is a chance that the child solutions will change slightly from the two parent combination, and this chance is set by the user.

Listing 10.4: Genetic Algorithm - parameter breeding

```

1 class pf_parameters:
2
3     def breed_event(p, c, opt):

```

```

4     pc = g.pfdata['params']['count']
5
6     pa = pf_generation.parents[p]
7     if(opt == 'p+p'):
8         pb = pf_generation.parents[p + pf_generation.pop_size_half]
9         pb_array = 'pop'
10    if(opt == 'p+f'):
11        pb = p
12        pb_array = 'fresh'
13
14    ca = c
15    cb = c + 1
16
17    # Breed
18    state = pf_generation.get_state()
19    for i in range(pc):
20        state = pf_generation.get_state(state)
21        if(state):
22            g.pfdata['params']['children'][ca, i] = g.pfdata['params']['pop'][pa, i]
23            g.pfdata['params']['children'][cb, i] = g.pfdata['params'][pb_array][pb, i]
24        else:
25            g.pfdata['params']['children'][cb, i] = g.pfdata['params']['pop'][pa, i]
26            g.pfdata['params']['children'][ca, i] = g.pfdata['params'][pb_array][pb, i]
27
28    # Mutate
29    g.pfdata['params']['children'][ca, :] = pf_generation.mutate(g.pfdata['params']['children'][ca, :], g.fit['mutate_chance'])
30    g.pfdata['params']['children'][cb, :] = pf_generation.mutate(g.pfdata['params']['children'][cb, :], g.fit['mutate_chance'])
31
32    # No clones - Child A
33    for pn in range(pf_generation.pop_size):
34        if((g.pfdata['params']['children'][ca, :] == g.pfdata['params']['pop'][pn,:]).all()):
35            r = numpy.random.rand(pc)
36            g.pfdata['params']['children'][ca, :] = g.pfdata['params']['children'][ca, :] * g.fit['no_clone_var'] * (0.5 - r)
37            break
38
39    # No clones - Child B
40    for pn in range(pf_generation.pop_size):
41        if((g.pfdata['params']['children'][cb, :] == g.pfdata['params']['pop'][pn,:]).all()):
42            r = numpy.random.rand(pc)
43            g.pfdata['params']['children'][cb, :] = g.pfdata['params']['children'][cb, :] * g.fit['no_clone_var'] * (0.5 - r)
44            break
45
46    loop = False
47
48    pf_potential.update(g.pfdata['params']['children'][ca, :])
49    rss = pf.get_rss()
50    if(rss is None):
51        loop = True
52
53    pf_potential.update(g.pfdata['params']['children'][cb, :])
54    rss = pf.get_rss()
55    if(rss is None):
56        loop = True
57
58    if(loop == False):
59        g.pfdata['params']['children_rss'][ca] = rss
60        g.pfdata['params']['children_rss'][cb] = rss

```

```

61
62     # Loop again or not
63     return loop
```

During the breed event, the program will check whether or not the two children already exist within the solution pool. If they are allowed to also enter the pool, the solutions may quickly all breed to become clones. The breed event forces any clones to be varied so they become a different and distinct solution.

Every now and then, an extinction event will occur that kills off any poor solutions and replaces them with variations of the better solutions. The user controls the frequency of the extinction event and the probability that a solution will be removed.

Top solutions are enhanced, at a frequency picked by the user, using a local optimisation algorithm: line-search and gradient descent. It does require many calculations and adds a substantial amount of time to the procedure, but it uses the gradient of solutions to find more optimal solutions more efficiently than randomly generated or bred solutions.

Line-Search and Gradient Descent

Listing 10.5: Gradient Descent

```

1  class gd:
2
3      precision = 1.0e-6
4      max_iterations = 10
5      h = 1.0e-8
6      momentum = 0.1
7      p_in = None
8      rss_in = None
9      p_out = None
10     rss_out = None
11     fix = None
12
13     # Central Difference
14     def df(p):
15         d = numpy.zeros((len(p),))
16         for i in range(len(p)):
17             if(gd.fix[i] == 0.0):
18                 d[i] = 0.0
19             else:
20                 p_f = numpy.copy(p)
21                 p_f[i] = p_f[i] + gd.h
22                 p_b = numpy.copy(p)
23                 p_b[i] = p_b[i] - gd.h
24                 d[i] = (gd.rss(p_f) - gd.rss(p_b)) / (2 * gd.h)
25         return d
26
27     def line_search(p, dp):
28         # Back track
29         best_rss = gd.rss_in
30         best_gamma = 0.0
31         gamma = 20.0 * gd.last_gamma
32         if(gamma < 1.0e-8):
33             gamma < 1.0e-6
34         loop = True
35         n = 0
36         while(gamma > 1.0e-12):
37             n = n + 1
38             p_test = p - gamma * dp
```

```

39     rss = gd.rss(p_test)
40     gamma = 0.5 * gamma
41     if(best_rss is None or rss < best_rss):
42         p_best = p_test
43         best_rss = rss
44         best_gamma = gamma
45     gd.last_gamma = best_gamma
46     return best_rss, best_gamma * dp
47
48
49 # Gradient Descent
50 def opt(f_rss, p0, fix=None):
51     gd.rss = f_rss
52     gd.last_gamma = 1.0
53     p = p0
54
55     if(type(fix) == numpy.ndarray):
56         if(len(p) == len(fix)):
57             for i in range(len(fix)):
58                 if(fix[i] != 0.0):
59                     fix[i] = 1.0
60             else:
61                 fix = numpy.zeros((len(p0),))
62                 fix[:] = 1.0
63         else:
64             fix = numpy.zeros((len(p0),))
65             fix[:] = 1.0
66     gd.fix = fix
67     gd.p_in = numpy.copy(p)
68     gd.rss_in = gd.rss(p)
69
70     best_p = numpy.copy(gd.p_in)
71     best_rss = gd.rss_in
72
73     n = 0
74     last_dp = 0.0
75     while(n < gd.max_iterations):
76         df = gd.df(p)
77         rss, dp = gd.line_search(p, df)
78         if(rss > best_rss):
79             n = gd.max_iterations
80         else:
81             p = p - (gd.momentum * last_dp + dp)
82             last_dp = dp
83             rss = gd.rss(p)
84             best_p = numpy.copy(p)
85             best_rss = rss
86             n = n + 1
87
88     gd.p_out = numpy.copy(best_p)
89     gd.rss_out = best_rss
90
91     return p

```

Newton Gauss Optimisation

The Newton Gauss optimisation estimates the second order derivatives matrix, the Hessian, by multiplying the Jacobian and its transpose. It is not yet implemented in the code, but may be in the future. The algorithm is used to find a local optimum, but it can be unstable or return errors.

$$\begin{aligned}\vec{J}^T \vec{J} \vec{p} &= \vec{J}^T \vec{r} \\ \vec{r} &= \vec{y} - f(\vec{x}, \vec{p})\end{aligned}\tag{10.4}$$

Levenberg-Marquardt Optimisation

The Levenberg-Marquardt algorithm is similar to the Newton-Gauss method, but the estimate for the Hessian does not need to be calculated for each loop of the algorithm. The parameter, λ , is changed and the "Hessian" is updated periodically. This is not implemented in the code, but may be added in the future.

$$\begin{aligned}\left(\vec{J}^T \vec{W} \vec{J} + \lambda \text{diag} \left(\vec{J}^T \vec{W} \vec{J} \right) \right) \vec{p} &= \left(\vec{J}^T \vec{W} \vec{r} \right) \\ \vec{r} &= \vec{y} - f(\vec{x}, \vec{p})\end{aligned}\tag{10.5}$$

10.1.9 Minimising Runtime

Python, Fortran & OpenMP

When fitting a potential, for each trial set of functions, the program must calculate the energy, forces and stress for each of the configurations and tens or hundreds of energy only calculations for the equation of state and elastic constant calculations. Python is a well used programming language in many areas of Science and it has the benefits of many modern languages. For certain specific tasks, it is slow in comparison to Fortran.

The energy, stress and force calculations were programmed in Fortran and use OpenMP to calculate the many configurations in parallel. On a single thread, the fitting process could take weeks, but on a 20 core node the calculation time was reduced to under a couple of days.

Function Caching

The first version of the code used tabulated potentials, and this removed the varying complexity of difference functions and replaced it with a quick search and interpolation of tabulated data. However, the final version was developed to use and fit with analytic potentials. Several of the optional functions, including the knot-to-knot spline function, drastically increased the computational time.

The bulk property module in particular uses configurations with atoms in regular places, with their being a small set of unique atom separation values. Rather than generate tabulated data for each function, to then interpolate, the input and output values are cached. Once the value is calculated, it is quickly retrieved for subsequent calculations with the same input value.

10.1.10 Interpolation for Non Analytic Potentials

Where an analytic potential isn't provided, it is possible to use only a tabulated function. The values are discrete points that are evenly spaced, and Lagrange polynomials are used to interpolate values.

$$D = \{(x_0, y_0) (x_1, y_1) (x_2, y_2) \dots (x_n, y_n)\}\tag{10.6}$$

$$p_n(x) = \sum_{i=0}^n y_i L(i, x) \text{ where } l(i, x) = \prod_{j=0, j \neq i}^n (x - x_j) \quad (10.7)$$

Listing 10.6: Add two numbers function

```

1 // lagrange polynomial interpolation
2 function lpinterp_y(x, data)
3   // x - point value being interpolated at to calc f(x)
4   // data - four x,y data points (n by 2 array)
5   n = len(data, 1)
6   y = 0.0
7   for i = 1,n
8     l = 1.0
9     for j = 1,n
10      if (i != j) then
11        l = l * (x - data[j][1]) / (data[i][1] - data[j][1])
12      end if
13    next j
14    y = y + l * data[i][2]
15  next i
16 end function lpinterp_y

```

There are typically hundreds or thousands of data points, so to interpolate, the closest few points are used. Throughout the computer code four point interpolation was the preferred method, to balance computational speed with a well fitting polynomial.

The gradient of the potential functions are also computed using lagrange polynomials.

$$q_n(x) = \sum_{i=0}^{i=n} y_i g(i, x) \quad (10.8)$$

where $g(i, x) = \left(\prod_{j=0, j \neq i}^{j=n} \frac{1}{x_i - x_j} \right) \times \left(\sum_{k=0, k \neq i}^{k=n} \prod_{j=0, j \neq k, j \neq i}^{j=n} (x - x_j) \right)$

```

1 // lagrange polynomial interpolation
2 function lpinterp_dydx(x, data)
3   // x - point value being interpolated at fo calc f'(x)
4   // data - four x,y data points (n by 2 array)
5   n = len(data, 1)
6   dydx = 0.0
7   for i = 1,n
8     fx = 1.0
9     gx = 1.0
10    for j = 1, n
11      if (i != j) then
12        fx = fx / (data[i][1] - data[j][1])
13        psum = 1.0
14        for k = 1, n
15          if (i != k and j != k) then
16            psum = psum * (x - data[k][1])
17          end if
18        next k
19        gx = gx + psum
20      end if
21    next j
22    dydx = dydx + fx * gx * data[i][2]

```

```

23     next i
24 end function lpinterp_dydx

```

10.2 Iron-Palladium Potentials: Data and Initial Potential Parameters

The developed code was used to fit a potential for Iron, Palladium and their alloy. A number of problems and limitations were encountered, and these will be discussed in more detail throughout this chapter and the conclusion.

10.2.1 Bulk Properties for Fitting

The fitting procedure is for both Iron and Palladium, but the FCC allotrope of Iron that is used in this work does not exist, at room temperature and pressure. Experimental data was available for Palladium, but DFT generated data was required for Iron.

Palladium [FCC]

The input parameters for Palladium were

Element	PD
Structure	Face Centered Cubic
a_0	3.89 Angstrom [97]
Nearest Neighbour	2.75 Angstrom [97]
Basis vectors	$\begin{bmatrix} 1.0 & 0.0 & 0.0 \\ 0.0 & 1.0 & 0.0 \\ 0.0 & 0.0 & 1.0 \end{bmatrix}$
E_{coh}	3.91 eV [106]
B_0	195.5 GPA [106]
Elastic Constants	$\begin{bmatrix} 234.1 & 176.1 & 176.1 & 0 & 0 & 0 \\ 176.1 & 234.1 & 176.1 & 0 & 0 & 0 \\ 176.1 & 176.1 & 234.1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 71.2 & 0 & 0 \\ 0 & 0 & 0 & 0 & 71.2 & 0 \\ 0 & 0 & 0 & 0 & 0 & 71.2 \end{bmatrix}$

Table 10.1

Iron [FCC]

The input parameters were calculated by DFT, with an FCT structure rather than FCC, and the calculated bulk modulus, elastic constants and other related properties are below.

Element	Fe
Structure	Face Centered Cubic
a_0	3.59 Angstrom
Nearest Neighbour	1.85 Angstrom
Basis vectors	$\begin{bmatrix} 0.96 & 0.0 & 0.0 \\ 0.0 & 1.00 & 0.0 \\ 0.0 & 0.0 & 0.96 \end{bmatrix}$
E_{coh}	-4.32 eV
B_0 (GPA)	226.1
E (GPA)	356.8
G (GPA)	144.8
Poisson Ratio η	0.23
Elastic Constants	$\begin{bmatrix} 364.6 & 141.6 & 233.8 & 0 & 0 & 0 \\ 141.6 & 298.7 & 130.4 & 0 & 0 & 0 \\ 233.8 & 130.4 & 364.6 & 0 & 0 & 0 \\ 0 & 0 & 0 & 186.3 & 0 & 0 \\ 0 & 0 & 0 & 0 & 266.8 & 0 \\ 0 & 0 & 0 & 0 & 0 & 186.3 \end{bmatrix}$

Table 10.2

10.2.2 Reference Database

A reference database of 72 configurations for iron, 80 configurations for palladium and 152 combined for iron-palladium was created using the DFT code PWscf. There are a

10.2.3 Potential Functions & Functionals

There are a large number of functions and functional used in the literature, and the potentials provided by H. Sheng are numeric. In this work, it was decided to use a knot-to-knot spline for the pair potential that is attached to a ZBL potential for small separations, with a fixed node at the cutoff, with a zero value and gradient. The radial electron density was calculated using DIRAC and, from this, a knot-to-knot spline was fitted with a forced cutoff to $\rho(r_{cut}) = 0, \rho'(r_{cut}) = 0$. Finally, the embedding functionals take the form $f(x) = p(1) \times \sqrt{r} + p(2) * r^2$ as used by Mendelev[105][107], Ackland[107], Olsson[72] and Wallenius[72].

DIRAC Density Plots

Whilst the potentials in this work are matched to DFT calculated forces, energies and bulk properties of the materials, using the radial electron density gave a starting point for each element. For an alloy, the density functions cannot have an arbitrary scale, as the density of multiple elements will be summed to calculate the energy of an atom embedded in that density. For this reason, it was also reasonable to start with density distributions computed by DIRAC, each relative in scale to the other.

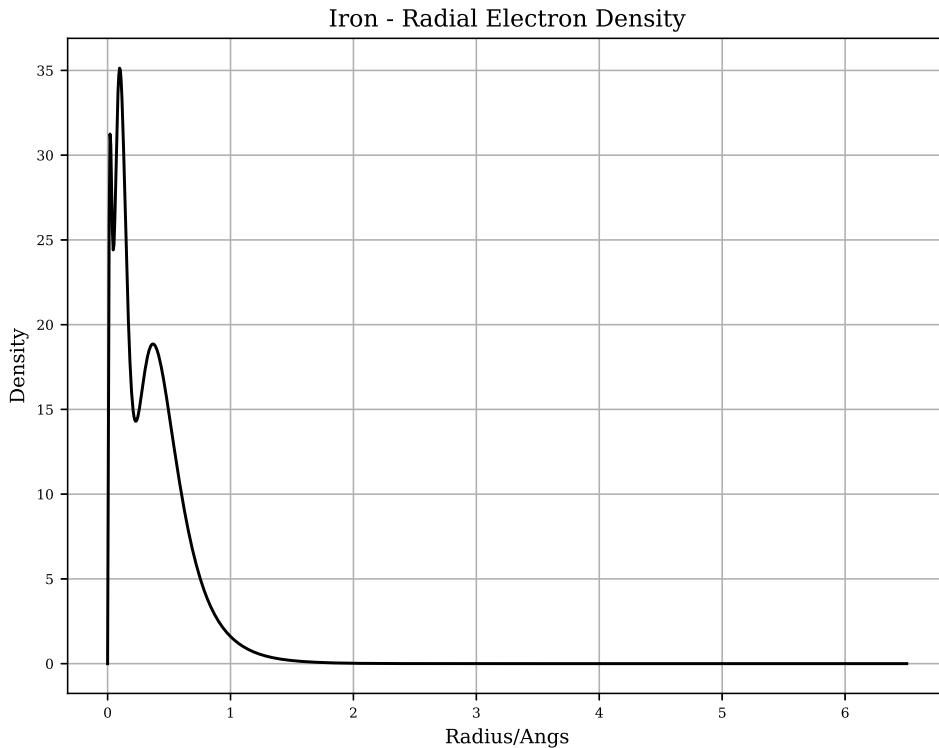


Figure 10.2: DIRAC calculated radial electron density for Iron

Unfortunately, as may be expected, there were a number of issues in using the density distribution of a single atom. The atomic separation for both iron and palladium, for a relaxed crystal, begins at approximately 2-2.5 angstrom, but most of the electrons, from all shells, are distributed much closer. The plot does not take into consideration the valence electrons shared in the crystal, bonding the metal atoms together.

From the point of view of the potential, the pair potential will dominate at small separations due to the ZBL function. The density function will be important at separations of 2 angstrom up to the cutoff radius (5-6 angstrom).

Sheng Potentials

The Sheng potentials available are numeric and so the functions used in this work were first fit to the tabulated data??. This gave a starting point for both iron and palladium.

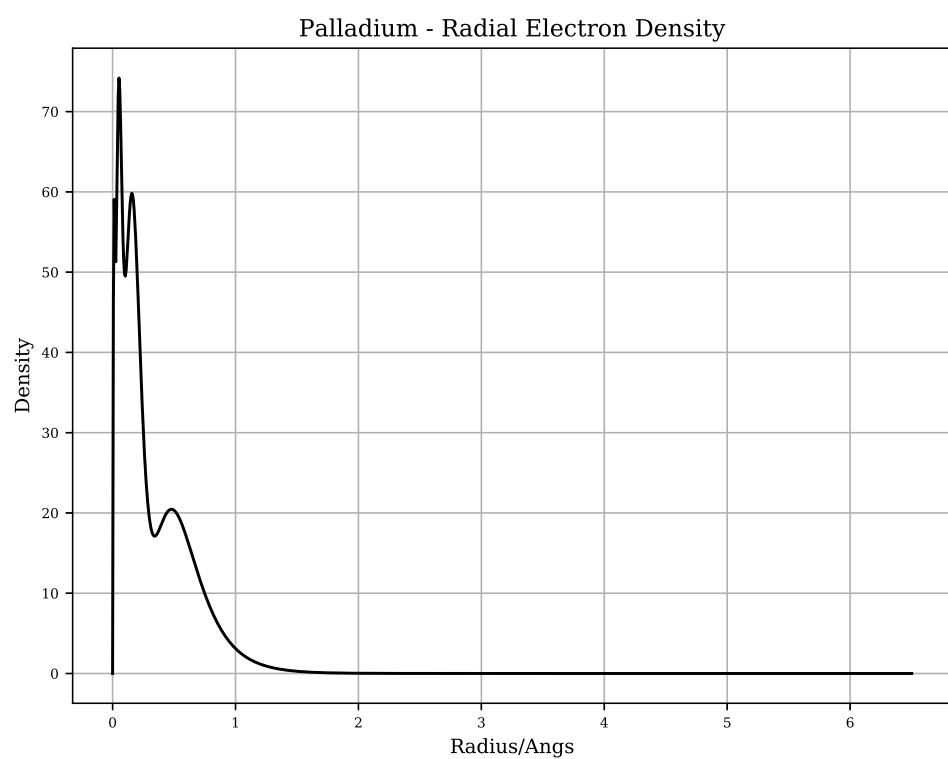


Figure 10.3: DIRAC calculated radial electron density for Palladium

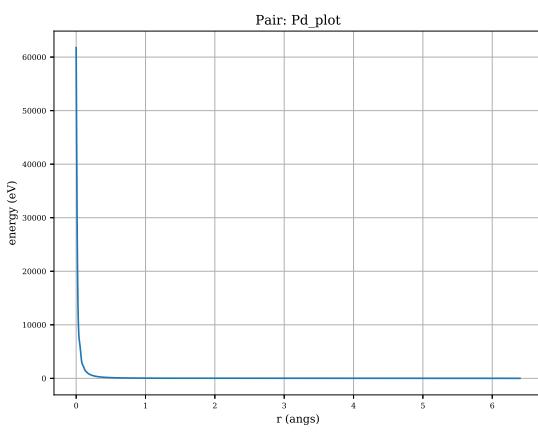


Figure 10.4: Sheng palladium pair function

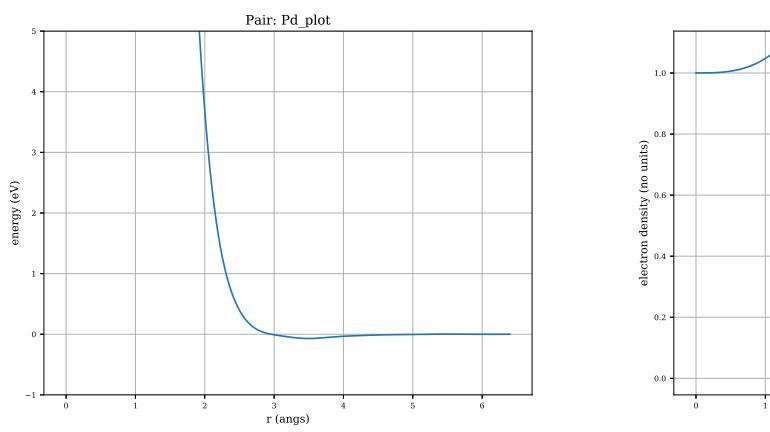


Figure 10.5: Sheng palladium pair function (zoomed in)

Figure 10.6

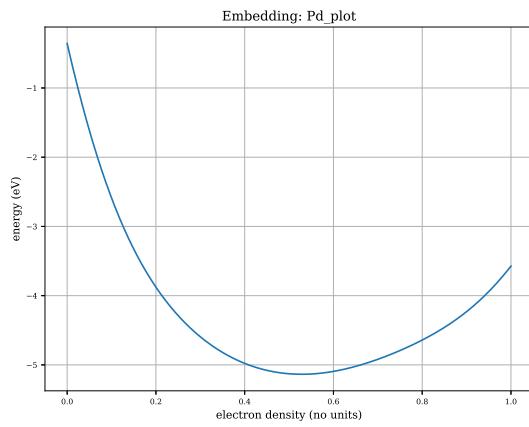


Figure 10.7: Sheng palladium embedding function

Preliminary Choices for the Potential Functions

The form of each function was decided by the function type, past forms from the literature and testing. The density function for all potentials was a knot-to-knot spline that was itself fit to the radial density distributions calculated by DIRAC.

The pair potentials were a cubic knot-to-knot spline that was originally fit to the Sheng potential for Palladium. The spline originally started with 30-40 knots, but several of those at small separations were removed, as varying them had little effect or no effect on the calculation of bulk properties or the energy, forces and stresses of the prepare atom configurations. The spline is forces to end with a value and derivative equal to zero at the cutoff radius.

Originally, the density function was a cubic knot-to-knot spline of the DIRAC produced density plots, but most of the density data points were at a small separation and again had little to no effect on the potential. The Sheng Palladium density was used as a starting point, and the density close to 0 was also reduced so the ZBL pair would dominate at small separations.

The embedding functional was selected as a type used by Mendelev and Ackland for Iron, with a Finnis-Sinclair negative square root, a positive or negative squared term and a positive quartic term to ensure a low or negative energy while at small electron densities, but a positive energy as the electron density continues to increase.

$$F(\rho) = a \times \sqrt{\rho} + b\rho^2 + \rho^4 \quad (10.9)$$

Both the iron and palladium potentials were derived separately as pure elements, and then again as an alloy.

10.3 Iron-Palladium Potentials: First Potential

The iron potential consisted of a cubic knot-to-knot spline for the pair and density functions, and the Mendelev and Ackland type embedding functional as described in the previous subsection. These functions were fit to the Sheng palladium plot to give the potential a starting point, and to ensure the density plot was a similar magnitude to the Palladium potential that is also being derived, to give a reasonable starting point for the alloy fitting.

Similarly to the iron potential, the palladium potential consisted of a cubic knot-to-knot spline were used for the pair and density with a 2 parameter embedding functional similar to that used by Mendelev and Ackland. The Sheng Palladium potentials were used as a starting point. The parameters of the three functions were fit to the Sheng plots before then being fit to the DFT data and experimental bulk properties.

A total of 72 configurations were used in the fitting of iron, but unfortunately only one of these had slightly randomised atomic positions. This was due to an increase in SCF cycles and a number of convergence failures with the slightly perturbed atom positions. The palladium potential was fit using the known experimental parameters and 80 DFT configurations, including 10 with randomised atom locations, slightly perturbed from the perfect crystal locations.

Finally, the two already derived potentials (above) were combined. A seventh potential was required to finish the potential, this being the iron-palladium pair potential. The iron-iron pair potential was used as a starting point. All seven potentials were then fit to the iron, palladium and iron-palladium DFT configurations as well as both the iron and palladium FCC bulk properties.

Throughout, the simulated annealing and genetic algorithms were use in combination to optimise the potential parameters.

Fe-Pd Potential Properties

The properties are calculated throughout the fitting process, but the final properties for both iron and palladium are presented in tables ?? and ??.

Parameter	Experimental/DFT	This Potential
a_0	3.42	3.58
e_0	-4.27	-4.27
B_0	222.0	238.8
C_{11}	365.6	356.8
C_{22}	298.7	296.1
C_{33}	364.0	356.8
C_{44}	186.3	180.9
C_{55}	266.8	263.2
C_{66}	186.3	180.9
C_{12}	141.6	139.1
C_{13}	233.8	241.7
C_{23}	130.4	127.3

Table 10.3

Parameter	Experimental/DFT	This Potential
a_0	3.89	3.95
e_0	-3.91	-3.91
B_0	180.0	171.3
C_{11}	234.0	248.9
C_{12}	176.0	167.2
C_{44}	71.2	67.6

Table 10.4

Fe-Pd Property Plots

The equation of state and elastic constants were used to fit the potential, and for both iron and palladium the fit reasonable well. The iron EOS does begin to drop off whereas it should continue to curve up, although the palladium EOS is better (appendix L.3).

To test the cohesive energy a short python code was written to generate 4x4x4 FCC supercells with a lattice parameter ranging 9.58 angs to 36.9 angs for iron and 10.9 angs to 42.0 angs for palladium. An energy calculation was performed for each element, using the derived potential, and the results were plotted (appendix L.4).

The cohesive energy plots for both iron and palladium take on a recognisable shape and, over all they do give the predicted cohesive energies. Neither is very smooth and to improve the reproduction of the correct energy at varying lattice parameters the energies should be computed using DFT and used during the potential fit.

The surface energies were also calculated. A small python program was created to generate a set of 4x4x4 FCC atoms. The position and separation of the atoms was fixed relative to one another, and anchored to the 0,0,0 coordinate, whilst the unit vector in the z axis was increased. This represents the transition from a bulk material to a slab with two surfaces in the xy-plane.

$$E_s = \frac{E_{total} - N_a \times E_{bulk}}{2A_{surface}}$$

where E_s = surface energy
 and E_{total} = total energy of slab
 and N_a = atoms in bulk
 and E_{bulk} = atom energy in bulk
 and $A_{surface}$ = surface area of slab in xy-plane

(10.10)

The calculated surface energies were $0.205\text{eV}/\text{ang}^2$ ($3,290\text{mJ}/\text{m}^2$) for iron and for $0.106\text{eV}/\text{ang}^2$ ($1690\text{mJ}/\text{m}^2$) palladium. The plots for these calculations are in appendix ???. Whilst the value for palladium is close to the experimental and Sheng values, ($2,000\text{mJ}/\text{m}^2$ and $1,747\text{mJ}/\text{m}^2$ respectively [22]), the calculations were performed without relaxing the atoms at each stage. This will be discussed further in chapter 12.

10.4 Iron-Palladium Potentials: Second Potential

A second attempt was made to derive an iron-palladium and several changes were made. First, the neighbour separations were examined for both palladium and iron, as shown in 10.8 and . Knot-to-knot splines offer a high degree of flexibility, but there is a problem. If the movement of one, or more, knots does not have a direct effect on any of the bulk property or energy, stress, force calculations it will move during optimisation resulting in what looks like a very badly behaved function. It also unnecessarily increases the parameter space for optimisation.

The number of knots for the pair functions was reduced to 7, including the start and end knot. This would help reduce the number of superfluous knots, decreasing the parameter space, with the aim of also ending with a smoother function. A major change was also made to the cubic knot function. The entire function would be the zbl plus a cubic knot spline, to give a function with a continuous value and derivative (eq. 10.11).

$$v(r) = zbl(r, z_a, z_b) + spline(r, \vec{p}) \quad (10.11)$$

Another major change implemented was to the way the cubic knot spline was implemented. Originally, the x positions for the knots were set by the user and the y positions were varied by the fitting program. The derivative at each point, needed to calculate each knot to knot spline, was interpolated. In the latest version the user specifies x, y(x) and y'(x) for the end knot (in this work, 6.5 0.0 and 0.0) and the x value of the start knot (in this work, 0.0). The remaining x, y(x) and y'(x) are all parameters that are varied by the fitting code.

The spline used previously for the density function was replaced with a 2 parameter Slater type function. The embedding functional remained the same. The starting potentials for both iron and palladium were fit to the Sheng potential for palladium and the same bulk properties and dft configurations were used as for the first potential.

The fitting code was modified to use the scipy optimization library. As a result, the potentials were fit using the simulated annealing and genetic algorithm for global optimisation, then Nelder-Mead and Broyden Fletcher

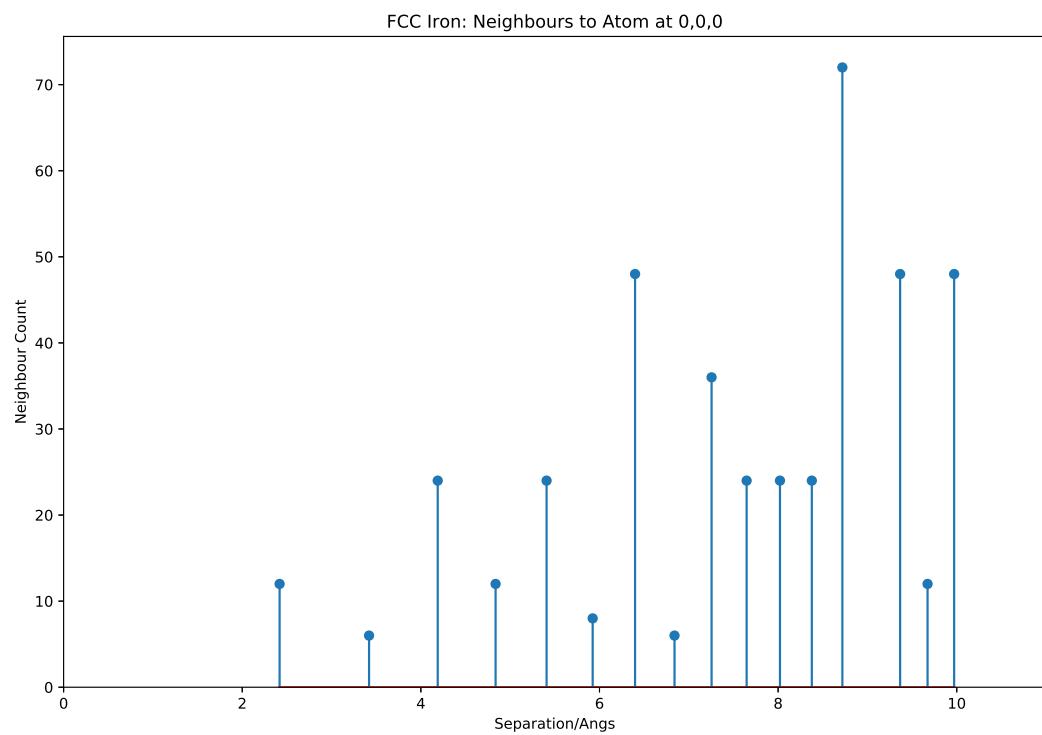


Figure 10.8: FCC Iron Neighbour Separation

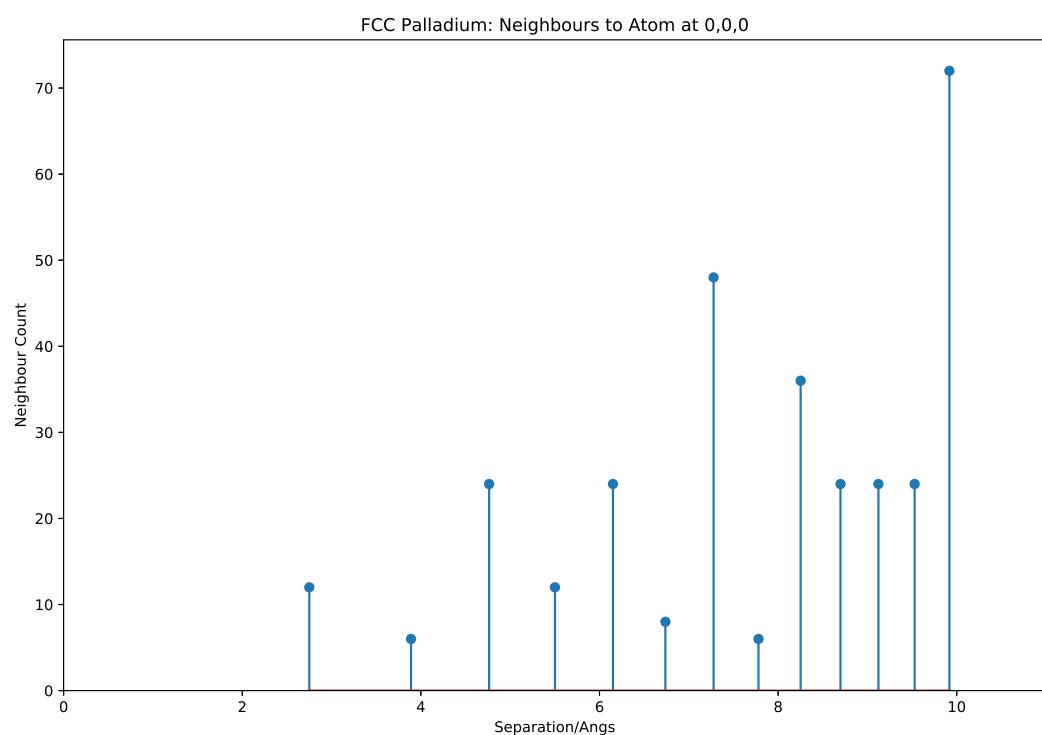


Figure 10.9: FCC Palladium Neighbour Separation

Goldfarb Shanno locally.

Fe-Pd Calculated Properties

The properties are calculated throughout the fitting process, but the final properties for both iron and palladium are presented in tables ?? and ??.

Parameter	Experimental/DFT	This Potential
a_0	3.42	3.40
e_0	-4.27	-4.29
B_0	222.0	229.1
C_{11}	365.6	365.9
C_{22}	298.7	298.6
C_{33}	364.0	365.9
C_{44}	186.3	187.6
C_{55}	266.8	258.4
C_{66}	186.3	187.6
C_{12}	141.6	141.1
C_{13}	233.8	238.1
C_{23}	130.4	124.8

Table 10.5

Parameter	Experimental/DFT	This Potential
a_0	3.89	3.87
e_0	-3.91	-3.91
B_0	180.0	171.3
C_{11}	234.0	232.9
C_{12}	176.0	176.7
C_{44}	71.2	68.8

Table 10.6

Fe-Pd Property Plots

The equation of state and elastic constants were used to fit the potential, and for both iron and palladium the fit reasonable well. The iron EOS does begin to drop off whereas it should continue to curve up, although the palladium EOS is better.

The cohesive energies were calculated similarly to the first version, using the prepared configurations to expand the cell until the atoms are isolated from their neighbours. The energies were plotted at various values of lattice parameter (see appendix M.4). Both the palladium and iron plots follow the expected trend and terminate with a zero energy, as would be expected. Neither plot is a smooth as may be expected, and the iron plot in particular seems quite poor. This supports using DFT to calculate similar plots, and using this data in the fitting procedure.

The calculated surface energies were $0.506\text{eV}/\text{ang}^2$ ($8,110\text{mJ/m}^2$) for iron and for $0.058\text{eV}/\text{ang}^2$ (932mJ/m^2) palladium. The plots for these calculations are in appendix ??.

Chapter 11

Conclusions

It is known that Chromim provides protection to stainless steel through a passive protective layer, but this becomes ineffective to protect the steel under irradiation, as chromium is depleted from the grain boundary. Platinum Group Metals also provide a mechanism to protect steel from corrosion, and experimentaion or simulations may be used to investigate this. If either of the two chosen PGMs, Palladium or Ruthenium, are depleted at the grain boundary under irradiation as well as depletion of Chromium, the steel will become susceptible to corrosion.

Experimental testing is cheaper and easy if an ion beam is used to replicate neutron damage. Either source will cause the target material to become radioactive. It is important to be able to calculate how radioactive, and how long after irradiation (to the desired damage, in DPA) it will be until the material is safe to be handled.

Computer simulations to simulate radiation damage will be much too large and complex for first-principles calculations. They will need either classical molecular dynamics or kinetic monte carlo simulations to encapsulate the much larger simulation volumes, to contain the atoms effected by the damage cascade, and to represent the grain boundary. This requires derivation of an appropriate interatomic potential using experimental data and first-principles data where it is not available, or is either impossible or impractical to measure.

11.1 Inter Granular Stress Corrosion Cracking

Austenitic stainless steel is particularly suscesptible to inter granular stress corrosion cracking. It has good properties, including a good resistance to corrosion due to its high chromium and nickel content. Under irradiation, the chromium at grain boundaries is depleted with the formation of Chromium carbides at the boundaries. Three requirements for intergranular stress corrosion cracking to occur are:

- a susceptible material (austenitic stainless steel)
- stress, welds, swelling, pressure, any applied stress (high pressure in reactor environment)
- a corrosive environment (radiolysis of water)

The passive chromium layer, under normal conditions, protects the steel from corrosion. The steel is made up from small grains, and the surface of these grains of crystalline metal are where the protective layer covers. During irradiation, chromium is depleted, forming chromium carbides between the grain boundaries. As the percentage of chromium drops, the surface becomes prone to corrosion.

Small quantities of platinum group metals may be added to a steel to also increase the corrosion resistance of the steel. Platinum group metals are rare and much more costly than Iron, Nickel, Chromium. Where irradiation is not present, it has been shown experimentally that the benefit due to Palladium is lost due to the formation of PdMn nanoparticles, where Manganese is present in the steel, although no such particles are present when Palladium was replaced by Ruthenium. It is expected that Palladium will not form such nano particles in steel without Manganese, but what about under irradiation? Are the grain boundaries enriched with Platinum group metals, depleted or remain unchanged?

11.2 Activity Code

Testing a material inside a nuclear reactor is an expensive experiment. There are many more ion sources around the world than high flux neutron sources, and ion beams are easier to direct due to the charge of the ion versus the neutral charge of the neutron. The ion energies required to create similar sized damage cascades in the material are high enough to also have a chance of transmuting the target nuclei. These transmuted nuclei are most likely going to be radioactive. To reach damage dose levels comparable to a component in a GenIV reactor by the end of its life (up to 200 DPA), the target material will become dangerously radioactive.

Bateman's equation may be used to calculate the amount of an isotope in a decay chain of several isotopes, and thus how radioactive an isotope would be after a certain time. Major modifications were required to be made to the equation to also include branching factors and source rates for isotopes generated by an ion beam.

11.3 Modified Equation and Activity Code

Whilst the equation was first tackled by solving the differential equations head on, it became obvious that using Laplace transforms would be the logical choice; this was also the route Bateman followed when deriving the original equations. Once transformed, a numerical algorithm (Gaver-Stehfest) was programmed and used to solve, with some success. It isn't an exact solution, and the errors incurred often resulted in negative amounts of an isotope, and this was unacceptable. More progress was made towards solving the problem analytically, and by using partial fractions, obviating the need for a numerical method.

The modified equation as derived is an original contribution to knowledge.

The Activity was created to automate the process of calculating the activity of an ion irradiated target. The program:

- reads in simulation details (target composition, thickness, duration etc)
- reads in ion cross section data from the TENDL database
- processes SRIM exyz data points
- calculates the amount of each isotope by the end of the simulation
- calculates the activity of each isotope and the dose an average human would be exposed to

A number of simplifications are made to the model, but predicted the radioactivity of certain peaks reasonably well with a sample of proton irradiated pure iron. Given more time, the integration of the ion paths and cross section data would be revised. A range of alloys would also be irradiated by a proton beam and, once their activity is measured at several time intervals after irradiation, they would be compared to the activity predicted by the Activity code.

11.4 Interatomic Potential: Iron-Palladium

It would be too big a task to develop an interatomic potential that includes Iron, Palladium, Chromium, Nickel and any other elements that make up a typical sample of austenitic stainless steel; Iron and Palladium are the starting point, with the possibility of adding more elements to the potential in the future.

Palladium exists in the FCC state at normal conditions, but Iron is BCC. The relatively high percentage of Nickel in the steel stabilises the austenitic FCC structure. The bulk properties, such as the lattice parameter, bulk modulus, elastic constants and so forth, are experimentally known for Palladium. As iron doesn't exist in the gamma phase at normal conditions, DFT calculations were used to find the bulk properties of gamma iron.

11.5 Collinear Spin DFT

With most models, there will be a trade off between simplicity, accuracy and the computational cost of the model. The parameters that had the largest impact on how long a DFT calculation will take are the planewave energy cutoff and number of k-point used to integrate the Brillouin zone. These should be reduced as much as possible, while keeping the energy and force results within a tolerance determined by the user.

A choice in the complexity of the calculation also had to be decided upon.

Pure Iron under normal conditions has a BCC structure and energetically favours a ferromagnetic configuration. Chromium also has a BCC structure under normal conditions, but it's optimum structure magnetically is antiferromagnetic. Gamma phase Iron (FCC) is also antiferromagnetic, and so it will be important to include magnetism in the DFT calculations.

The options in increasing complexity, and computational time, are:

- non-magnetic
- collinear - suitable for high symmetry, ferromagnetic/antiferromagnetic
- non-collinear - spin may not be aligned in the same direction

For the Iron and Palladium calculations, collinear calculations were used and the starting configurations of the atoms were set up in accordance with the known configurations i.e. BCC Iron was configured such that the spins were aligned in the same direction, and FCC were set up in an antiferromagnetic setting.

A Python code was developed to automatically converge the aforesaid parameters. Whilst it worked well for the planewave cutoff parameters, the choice of smearing type, amount and number k-points felt more like an art than a science. It was more useful to select the number of k-points with a reasonable smearing width, within the convergence threshold set, and then to use those settings to calculate properties for the material and compare to the known values.

During testing, a much larger number of k-points were needed for the predicted properties of Aluminium to replicate the known properties reasonably well. Unfortunately, the number of k-points used for Iron and Palladium was constrained by the amount of time and the RAM available on the compute nodes. Ideally, more k-points would have been used, but this was not possible.

11.6 Bulk Properties of Orthorhombic FCC Iron

As pure Iron does not exist in its gamma phase at room temperature, DFT calculations were used in place of experimental data. As eluded to in the previous section, collinear spin DFT was selected. Although a smaller number of k-points were used, the calculation lasted for approximately 30 days to complete.

Rather than calculate the input files manually, a program was created to automate this process (QE_EOS). It was also designed to cache input and output files for the DFT program, PWscf, such that it would load an output file from the cache if it had already successfully run, to reduce the calculation time.

The relaxed, energy minimised, shape of the antiferromagnetic FCC iron was not cubic, but orthorhombic. The equation of state calculation is specifically for a cubic crystal, but the elastic constants were calculated for the orthorhombic crystal. From the 9 calculated elastic constants the melting point, bulk modulus, shear modulus and Young's modulus were calculated. These data are an original contribution to science.

11.7 Palladium-Iron Configurations as Fitting Data

To be able to fit the iron-palladium potentials, the bulk properties of both pure FCC iron and FCC palladium were either calculated or taken from experimental values. The potentials would be trained to fit this data. As well as this, a number of configurations of iron-palladium were generated with their atoms slightly displaced from their lattice positions. This would give a wider range of atom separations and force values for the fitting program to use.

The force data were entirely from first-principles calculations, and ideally there would have been more configurations, and they would have been computed with a higher number of k-points. In addition, a wider range of vacancies, interstitials and mixing concentrations would have been desirable. The latter point is limited due to the configuration sizes being only 32 atoms. The atomic percentage on Palladium or Ruthenium doped is half a percent which would be better represented with a one PGM atom in a 4x4x4 FCC 256 atom supercell.

11.8 Iron-Palladium Potential

The derived Iron-Palladium potential

The collinear DFT calculations showed that Iron FCC, where magnetism is taken into account, is slightly tetragonal with the atoms arranged in a FCT structure. The aligned spin of valence electrons in one direction

11.9 Two-Band EAM Contribution to DL-Poly

During the early stages of this work it was clear that the Embedded Atom Method was a good candidate for the type of potential to be derived as it was well suited to modelling metals. Work by Prof. Ackland at the University of Edinburgh considered using a separate density function and embedding functional to represent the S-Band and D-Band of transition elements such as Caesium. Work by P. Olsson and J. Wallenius also used this two band version to model Iron-Chromium.

After meeting with Prof. Todorov at Daresbury Laboratory, the source code for DL-Poly was edited to include new keywords, additional arrays to store the two-band density and embedding data for the second band and modifications to the energy and force subroutines. This was then released with version 4.05 of DL-POLY in July 2013.

Chapter 12

Future Work

Chapter Summary

12.1 Activity Code

12.1.1 Experimental Activity Readings for Ion Irradiated Targets

The activity code was developed using data, and generated data, from the TENDL data file. To test the validity of the code, an Iron sample was irradiated and its activity was measured several days after irradiation.

To validate the code further, a range of pure targets and alloys would be irradiated by a cyclotron, with targets of varying thickness and beams of varying fluences and energies. This would generate data to be used to compare to the results predicted by the code.

12.1.2 Code Improvements

Ideally, a built in ion transport code similar to SRIM would be developed. Failing this, an improvement in the reaction rate calculation and a rewrite of the code in Python with F2PY to improve ease of developing in the future.

12.2 Potential Fitting and MD Simulations

The DFT calculations would be processed on a larger computer with a similar k-point count, but larger supercell size (4x4x4) to increase the accuracy of the calculations.

12.3 Improved Potential: Taking Into Account Non-Isotropic Structure of "FCC" Iron

12.4 Improved Fitting: Surface Energy, Cohesive Energy and Defects

The cohesive energy and surface energy plots for the derived iron and palladium potentials were recognisable, but they were not smooth and did have many small bumps in the plots. Having plotted the same for Aluminium,

the DFT results were much smoother.

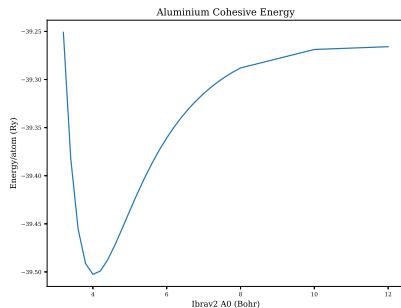


Figure 12.1: DFT Aluminium Cohesive Energy

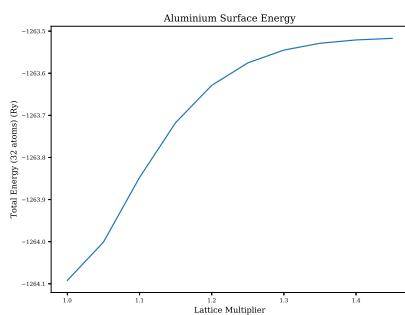


Figure 12.2: DFT Aluminium Surface Energy

Rather than just fit the data to the energy surface, it would be useful to create DFT configurations for the cohesive energy and surface plots for both Iron and Palladium. These configurations, and the energies, forces and stresses computed by DFT would be used to improve the fitting process.

Acronyms

2BEAM Two Band Embedded-Atom method. 110

ABWR Advanced Boiling Water Reactor. 4

AGR Advanced Gas Reactor. 5, 8, 36

BCA Binary Collision Approximation. 47

BCC Body Centered Cubic. 111, 134, 136

BFGS Broyden Fletcher Goldfarb Shanno. 163

BWR Boiling Water Reactor. 10, 35

BZ Brillouin Zone. xiv, 84

CCGT Combined Cycle Gas Turbine. 6, 9, 10

DFT Density Functional Theory. vi, 16, 50, 70, 78, 87, 90–92, 110, 111, 114, 129, 159, 160, 164, 168–170

DPA Displacements Per Atom. 127, 289, 300

EAM Embedded-Atom method. 15, 64, 110

ECIS Equations Couples Itrations Squentielle. 44

ECP Electro Chemical Potential. 35

EDF lectricit de France. 5

ELSY European Lead Fast Reactor. 9

ENDF Evaluated Nuclear Data File. 45, 46

EOS Equation of State. 160, 163

EPR European Pressurised Reactor. 5–7

FBR Pebble-Bed Reactor. 10

FCC Face Centered Cubic. 28, 33, 134, 159–161, 168

FCT Face Centered Tetragonal. 168

FS Finnis-Sinclair. 110

GenIV Generation IV. 42

- GFR** Gas-Cooled Fast Reactor. 36
- GGA** Generalized Gradient Approximation. 89–91, 129, 134
- HFIR** High Flux Isotope Reactor. xiv, 38, 39, 42
- IGSCC** Inter Granular Stress Corrosion Cracking. 1, 26, 33, 35
- JEFF** Joint Evaluated Fission and Fusion File. iv, 46
- KS** Kohn-Sham. 88
- LDA** Local Density Approximation. 89, 90, 129, 134
- LFR** Lead-Cooled Fast Reactor. 8, 10
- LINAC** Linear Accelerator. 40
- LMA** Levenberg-Marquardt Algorithm. 99
- LSDA** Local Spin Density Approximation. 89, 90, 129
- LWR** Light Water Reactor. 35
- MD** Molecular Dynamics. 92, 110
- MSR** Molten Salt Reactor. 11
- NDS** Nuclear Data Services. 43
- NEA** Nuclear Energy Agency. 46
- NG** Newton-Gauss. vi, 99
- ORNL** Oak Ridge National Laboratory. 40
- PBE** Perdew-Burke-Ernzerhof. 90
- PBESOL** Perdew-Burke-Ernzerhof functional revised for solids. 90
- PKA** Primary Knock-On Atom. 12, 14, 15, 24, 92
- PWR** Pressurised Water Reactor. 4–6, 10, 11, 35, 36
- PZ** Perdew-Zunger. 90
- RIS** Radiation Induced Segregation. 36
- SCC** Stress Corrosion Cracking. 35
- SCWR** Supercritical-Water-Cooled Reactor. 9–11
- SFR** Sodium-Cooled Fast Reactor. 9–11, 36
- SLAC** Stanford Linear Accelerator Center. 37
- SNS** Spallation Neutron Source. 40, 42
- SRIM** Stopping Range In Matter. 43, 47, 100, 106, 126

TENDL TALYS-based Evaluated Nuclear Data Library. vii, 45, 46, 119, 120, 122, 124

TGSCC Trans Granular Stress Corrosion Cracking. 35

TISE Time Independent Schrödinger Equation. v, 78, 80, 82, 87, 91

TRISO Tristructural Isotropic. 11

TRIUMF TRI-University Meson Facility. 37

VHTR Very-High-Temperature Reactor. 9, 10

VPI Vacancies Per Ion. 127, 300

XC Exchange-Correlation. 87–89, 91

ZBL Ziegler-Biersack-Littmark. 15, 68

Glossary

304SS Austenitic stainless steel: 18-20%Cr, 8-11%Ni. 36

316SS Austenitic stainless steel: 16.5-18.5%Cr, 10-13%Ni, 2.0-2.5%Mo. 36

antiferromagnetic neighbouring electrons orientate in opposite directions, cancelling out overall magnetism. 66

Bloch theorem electron wave functions may be written as a sum of plane waves $\psi_{\vec{k},n} \vec{r} = \sum c_{\vec{k}+\vec{G},n} \exp(i(\vec{k} + \vec{G})\vec{r})$. 83

Brillouin Zone the Wigner-Seitz cell in reciprocal space. 83

Cauchy pressure $C_{12} - C_{44}$. 66

enthalpy sum of internal energy plus pressure times volume $H = E_{internal} + PV$. 65

jellium a homogeneous electron gas. 85, 89

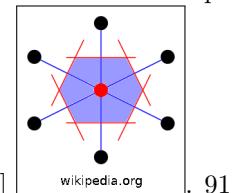
Neel temperature temperature at which an antiferromagnetic becomes a paramagnet. 66

orthorhombic $a, b, c, \alpha = \beta = \gamma = 90^\circ$. 50

plane wave a wave of parallel planes of constant frequency that are normal to the direction of propagation. 83

wave function contains all there is to know about a system and its meaning is given by a probability $|\Psi|^2 = 1$. 78

Wigner-Seitz cell given a lattice point, the set of all points in space which are closer to that lattice point



that any other lattice point constitute the Wigner-Seitz cell [solidstatebasicswcs]. 91

Bibliography

- [1] H. Khartabil. *Gen-4 Reactors*. 2013. URL: <http://www.gen-4.org/GIF/About/documents/25-Session2-3-Khartabil.pdf>.
- [2] Financial Times. *Cost of new Sizewell C Nuclear Plant*. 2020. URL: <https://www.ft.com/content/77c209f7-6d18-4609-ac3c-77d1b5b82b34>.
- [3] Energy for Generations. *ESB opens 820m Euro Carrington Power Station in Manchester*. 2020. URL: <https://www.esb.ie/tns/press-centre/2017/2017/03/13/esb-opens-820m-carrington-power-station-in-manchester>.
- [4] Yuchen Dou, Hong Luo, and Jing Zhang. “Elastic Properties of FeCr20 Ni8Xn (X = Mo, Nb, Ta, Ti, V, W and Zr) Austenitic Stainless Steels: A First Principles Study”. In: *Metals* 9 (145 2019).
- [5] Philippe F Weck et al. *Mechanical properties of zirconium alloys and zirconium hydrides predicted from density functional perturbation theory*. 2020. URL: <https://www.osti.gov/pages/servlets/purl/1259850>.
- [6] MIT Open Courseware. *Engineering of Nuclear Systems - Lecture 6A*. 2013. URL: <http://ocw.mit.edu/courses/nuclear-engineering/22-06-engineering-of-nuclear-systems-fall-2010/lectures-and-readings/MIT22\%5F06F10\%5Flec06a.pdf>.
- [7] GenIV International Forum. *Introduction to Generation IV Nuclear Energy Systems and the International Forum*. 2013. URL: <https://www.iea.org/topics/world-energy-outlook>.
- [8] Wikipedia. *Carnot's theorem (thermodynamics)*. 2020. URL: [https://en.wikipedia.org/wiki/Carnot\%27s\%5Ftheorem\%5F\(thermodynamics\)](https://en.wikipedia.org/wiki/Carnot\%27s\%5Ftheorem\%5F(thermodynamics)).
- [9] S. E. Jensen and E. Nonbol. *Description of the Magnox Type of Gas Cooled Reactor (MAGNOX)*. 1998. URL: <https://inis.iaea.org/collection/NCLCollectionStore/\%5FPublic/30/052/30052480.pdf>.
- [10] Wikipedia. *Combined cycle power plant*. 2020. URL: <https://en.wikipedia.org/wiki/Combined\%5Fcycle\%5Fpower\%5Fplant>.
- [11] World Nuclear. *Uranium Enrichment*. 2020. URL: <https://www.world-nuclear.org/information-library/nuclear-fuel-cycle/conversion-enrichment-and-fabrication/uranium-enrichment.aspx>.
- [12] Los Alamos National Laboratory. *Los Alamos National Laboratory Tour*. 2013. URL: <http://t2.lanl.gov/nis/tour/sch007.html>.
- [13] GenIV International Forum. *Very-High-Temperature Reactor (VHTR)*. 2020. URL: <https://www.gen-4.org/gif/jcms/c\%5F9362/vhtr>.
- [14] Sangeeta Deokattey et al. “Hydrogen production using high temperature reactors: an overview”. In: *Advances in Energy Research* 1 (1 2013), pp. 013–033.
- [15] M Müller and P. U. Foscolo. *Advanced Biomass Gasification - New Concepts for Efficiency Increase and Product Flexibility*. 2016.

- [16] MIT Open Courseware. *Engineering of Nuclear Systems - Lecture 6B*. 2013. URL: <http://ocw.mit.edu/courses/nuclear-engineering/22-06-engineering-of-nuclear-systems-fall-2010/lectures-and-readings/MIT22\%5F06F10\%5Flec06b.pdf>.
- [17] NEA. 2020. URL: <https://www-nds.iaea.org/exfor/>.
- [18] J. Fikar and R. Schäublin Ecole. “Molecular dynamics simulation of radiation damage in bcc tungsten”. In: *Journal of Nuclear Materials* (101 2009), pp. 97–101.
- [19] Andrew F. Calder et al. “Computer simulation of cascade damage in a-iron with carbon in solution”. In: *Journal of Nuclear Materials* (382 2008), pp. 91–95.
- [20] N J Carron. *An Introduction to the Passage of Energetic Particles through Matter*. 2007.
- [21] J. Rodenas and G. Verdu. “Analysis of nuclear reactions to determine the radionuclides generated and its activity in various devices”. In: *Radiation Physics and Chemistry* 167 (2020), pp. 1–6.
- [22] H. Sheng. *EAM Potentials*. 2012. URL: <https://sites.google.com/site/eampotentials>.
- [23] G. S. Was. *Fundamentals of Radiation Materials Science*. 2007.
- [24] S. Choudhury a et al. “Ab-initio based modelling of diffusion in dilute bcc FeNi and FeCr alloys and implications for radiation induced segregation”. In: *Journal of Materials* 411 (2011), pp. 1–14.
- [25] J. L. Seran and M. Le Flem. *Irradiation-resistant Austenitic Steels as Core Materials for Generation IV Nuclear Reactors*. 2017, pp. 285–328.
- [26] T.R. Allen and J.T. Busby. “Radiation damage concerns for extended light water Reactor service”. In: *Journal of Materials* 61 (2009), pp. 29–34.
- [27] A. Janotti and R.C. Reed. “Diffusion rates of 3d transition metal solutes in nickel by first-principles calculations”. In: *Acta Materialia* 53 (2005), pp. 2369–2376.
- [28] IAEA. 2019. URL: <https://inis.iaea.org/collection/NCLCollectionStore/\%5FPublic/29/010/29010110.pdf>.
- [29] Jeremy L. Gilber. *Metals - Basic Principles*.
- [30] S. Hoffmann and H. Hofsass. “Compositional Changes of Passive Films Due To Different Transport Rates and Preferential Dissolution”. In: *Corrosion Science* 31 (1990), pp. 573–578.
- [31] H. Knote and U. Stolz. “The Passivity of Iron-Chromium Alloys”. In: *Corrosion Science* 29 (7 1989), pp. 899–917.
- [32] P. J. Maziasz and M. Le Flem. *Properties of Austenitic Steels for Nuclear Reactor Applications*. Vol. 2. 2012, pp. 303–318.
- [33] Rolled Alloys. *What is alloy sensitization?* 2020. URL: <https://www.rolledalloys.com/dotAsset/50dc766c-08f5-43d9-bbda-5d9a9a2fa6d5.jpg>.
- [34] Speciality Steel Industry of North America. *Intergranular Corrosion*. 2020. URL: <https://www.ssina.com/education/corrosion/intergranular-corrosion/>.
- [35] R.W. Staehle. *Historical views on stress corrosion cracking of nickel-based alloys: the Coriou effect*. 2016.
- [36] Gary S. Was and Shigeharu Ukai. *Austenitic Stainless Steels*. 2019.
- [37] United States Nuclear Regulatory Commission. *AP1000 Design Control Document*. 2020. URL: <https://www.nrc.gov/docs/ML1117/ML11171A447.pdf>.
- [38] P. N. Standing. *The Long Term Storage of Advanced Gas-Cooled Reactor Fuel*. URL: <https://inis.iaea.org/collection/NCLCollectionStore/\%5FPublic/30/040/30040095.pdf>.
- [39] Guy O.H. Whillock et al. “Investigation of thermally sensitised stainless steels as analogues for spent AGR fuel cladding to test a corrosion inhibitor for intergranular stress corrosion cracking”. In: *Journal of Nuclear Materials journal* (498 2018), pp. 187–198.

- [40] IAEA. *Status report 81 - Advanced Passive PWR (AP 1000)*. 2020. URL: <https://aris.iaea.org/PDF/AP1000.pdf>.
- [41] Inc Framatome ANP. *EPR Design Description*. 2020. URL: <https://www.nrc.gov/docs/ML0522/ML052280170.pdf>.
- [42] F. Dalle et al. *Conventional austenitic steels as out-of-core materials for Generation IV nuclear reactors*.
- [43] IAEA Department of Nuclear Sciences and Applications. *Directory of Cyclotrons used for Radionuclide Production in Member States*. 2006. URL: <http://www-naweb.iaea.org/napc/iachem/cyclotrons/PDF/DCRP.pdf>.
- [44] Diamond. *About Synchrotrons*. 2020. URL: <https://www.diamond.ac.uk/Home/About/FAQs/About-Synchrotrons.html>.
- [45] World Nuclear. 2018. URL: <http://www.world-nuclear.org/information-library/non-power-nuclear-applications/radioisotopes-research/research-reactors.aspx>.
- [46] IAEA. 2018. URL: <https://nucleus.iaea.org/RRDB/RR/ReactorSearch.aspx>.
- [47] ORNL. 2018. URL: <https://neutrons.ornl.gov/sites/default/files/HighFluxIsotopeReactorUserGuide2.0.pdf>.
- [48] ORNL. 2018. URL: <https://neutrons.ornl.gov/hfir/parameters>.
- [49] ORNL. *High Flux Isotope Reactor (HFIR) USER GUIDE*. 2015. URL: <https://neutrons.ornl.gov/sites/default/files/High%20Flux%20Isotope%20Reactor%20User%20Guide%202.0.pdf>.
- [50] E. B. Iverson. *Neutron Moderation*. 2016. URL: <https://conference.sns.gov/event/56/attachments/64/99/Lecture%5F2a%5F-%5FNeutron%5FModeration%5F-%5FErik%5FIverson.pdf>.
- [51] M. N. H. Comsan. *Spallation Neutron Sources for Science and Technology*. 2011. URL: <https://inis.iaea.org/collection/NCLCollectionStore/%5FPublic/43/099/43099436.pdf>.
- [52] R. B. Leachman. *Peaceful Uses of Atomic Energy*. Vol. 2. 1956, p. 193.
- [53] P E Hodgson. “The Nuclear Optical Model”. In: *Rep. Prog. Phys.* 34 (1971), pp. 765–819.
- [54] A Koning, S Hilaire, and S Goriely. *TALYS 1.95 A Nuclear Reaction Program*. 2019.
- [55] NEA. *Janis Book*. 2020. URL: <https://tendl.web.psi.ch/tendl%5F2019/other/book-protons.pdf>.
- [56] C. Vaglio and J-F. Vidal. *The JEFF-3.1.1 Nuclear Data Library*. 2009.
- [57] M. D. Ziegler and J. P. Biersack. “SRIM - The stopping and range of ions in matter”. In: *Nuclear Instruments and Methods in Physics Research Section B* 268 (2010), pp. 1818–1823.
- [58] Roger Rousseau, Vassiliki-Alexandra Glezakou, and Annabella Selloni. “Theoretical insights into the surface physics and chemistry of redox-active oxides”. In: *Nature Reviews Materials* (5 2020), pp. 460–475. URL: <https://www.nature.com/articles/s41578-020-0198-9>.
- [59] F. D. Murnaghan. “The Compressibility of Media Under Extreme Pressures”. In: (1944).
- [60] A. Mahmoud ad A. Erba. *Crystal: Equation of State*. 2019. URL: <http://tutorials.crystalsolutions.eu/tutorial.html?td=eos&tf=eos%5Ftut>.
- [61] F. Birch. “Finite elastic strain of cubic crystals”. In: (1947).
- [62] gilgamesh. *Fitting Birch-Murnaghan*. 2013. URL: <http://gilgamesh.cheme.cmu.edu/doc/software/jacapo/appendices/appendix-eos.html>.
- [63] Michael J. Mehl. “Pressure dependence of the elastic moduli in aluminium-rich Al-Li compounds”. In: *Physical Review B* 47 (5 1993), pp. 2493–2500.
- [64] P. A. Korzhavyi and B. Johansson. “Density functional theory for calculation of elastic properties of orthorhombic crystals - applications to TiSi2”. In: *Journal of Applied Physics* 84 (9 1998), pp. 4891–4904.

- [65] B B Karki, G J Ackland, and J Crain. *Elastic instabilities in crystals from ab initio stressstrain relations*. 2013. URL: <http://www.homepages.ed.ac.uk/gja/karki1.pdf>.
- [66] G. Ghosh. “A first-principles study of cementite (Fe₃C) and its alloyed counterparts: Elastic constants, elastic anisotropies, and isotropic elastic moduli”. In: *AIP Advances* 5 (2015).
- [67] L. D. Brown and H. L. Marcus. “Elastic Constants Versus Melting Temperature in Metals”. In: *Scripta Metallurgica* 18 (1984), pp. 951–956.
- [68] A Koning, S Hilaire, and S Goriely. *The Stopping and Range of Ions in Matter*. 2015.
- [69] M. W. Finnis and J. E. Sinclair. “A simple empirical N-body potential for transition metals”. In: *Philosophical Magazine A* 50 (2003).
- [70] M. S. Daw and M. I. Baskes. “Embedded-atom method Derivation and application to impurities, surfaces, and other defects in metals”. In: *Physical Review B* 29 (12 1984), pp. 6443–6453.
- [71] Graeme J. Ackland. “Two-band second moment model for transition metals and alloys”. In: *Condensed Materials* (2005).
- [72] K. Nordlund and L. Målerba. “Two-band modelling of alpha prime phase formation in Fe-Cr”. In: *Physical Review B* 72 (2005).
- [73] G. Bonny et al. “Iron chromium potential to model high-chromium ferritic alloys”. In: *Philosophical Magazine* 91 (April 2011), pp. 1724–1746.
- [74] I.T.Todorov and W. Smith. *The DL_POLY_4 User Manual*. 2016.
- [75] Wikipedia. *Virial Stress*. 2016. URL: <https://en.wikipedia.org/wiki/Virial\%5Fstress>.
- [76] F. Ercolessi and J. B. Adams. “Interatomic Potentials from First-Principles Calculations: the Force-Matching Method”. In: *Europhys. Lett.* 26 (8 1994), pp. 583–588.
- [77] Steven H. Simon. *Solid State Basics*. 2013, p. 213.
- [78] Materials. *Microstructure and Mechanical Properties of Pressure-Quenched SS304 Stainless Steel*. 2019. URL: <https://www.mdpi.com/1996-1944/12/2/290/pdf#:~:text=This%20material%20displays%20a%20typical,size%20and%20irregular%20grain%20shape..>
- [79] Steven H. Simon. *Solid State Basics*. 2013.
- [80] MIT. *Physics for Solid State Applications*. 2004. URL: <http://web.mit.edu/6.730/www/ST04/Lectures/Lecture7.pdf>.
- [81] W. Kohn and L. J. Sham. “Self-Consistent Equations Including Exchange and Correlation Effects”. In: *Physical Review* 140 (1965).
- [82] SISSA. 2020. URL: <https://people.sissa.it/~degironc/ES/lectures/ABCofDFT.pdf>.
- [83] M. C. Payne et al. “Iterative minimization techniques for ab initio total-energy calculations: molecular dynamics and conjugate gradients”. In: *REVIEWS OF MODERN PHYSICS* 64 (1992).
- [84] Annabella Selloni. *The plane wave pseudopotential method (basic aspects)*. 2009. URL: https://th.fhi-berlin.mpg.de/th/Meetings/DFT-workshop-Berlin2009/Talks/OnlinePublication/0624-2_20090623-1_Selloni_FINAL_2perA4_-_Selloni-Berlin2.pdf.
- [85] P. Ziesche, S. Jurth, and J. P. Perdew. “Density functionals from LDA to GGA”. In: *Computational Materials Science* 11 11 (1998), pp. 122–127.
- [86] John P Perdew, Kieron Burke, and Matthias Ernzerhof. “Generalized Gradient Approximation Made Simple”. In: *Physical Review Letters* 77 (18 1996), pp. 3865–3868.
- [87] John P Perdew. “Generalized Gradient Approximations for exchange and correlation: A look backward and forward”. In: *Physica B* 172 (1991), pp. 1–6.

- [88] M R Benam, N Abdoshahi, and M Majidiyan Sarmazdeh. “Ab initio study of the effect of pressure on the structural and electronic properties of cubic LaAl03 by density function theory using GGA, LDA and PBESOL exchange correlation potentials”. In: *Physica B* 446 (2014), pp. 32–38.
- [89] periodictable.com. *Sodium: periodictable.com*. 2020. URL: <https://periodictable.com/Elements/011/data.html>.
- [90] periodictable.com. *Aluminium: periodictable.com*. 2020. URL: <https://periodictable.com/Elements/013/data.html>.
- [91] P Giannozzi et al. “Quantum Espresso”. In: *J Phys Condens Matter* 29 (2017).
- [92] H. Stehfest. “Algorithm 368 Numerical Inversion of Laplace Transform”. In: *Communications of the ACM* 13 (1970), pp. 47–49.
- [93] J. Ziegler. *The Stopping and Range of Ions in Matter*. 2019. URL: <http://www.srim.org/SRIM/SRIMINTRO.htm>.
- [94] webelements.com. *Webelements: Aluminium*. 2020. URL: <https://www.webelements.com/aluminium>.
- [95] webelements.com. *Webelements: Iron*. 2020. URL: <https://www.webelements.com/iron>.
- [96] webelements.com. *Webelements: Nickel*. 2020. URL: <https://www.webelements.com/nickel>.
- [97] webelements.com. *Webelements: Palladium*. 2020. URL: <https://www.webelements.com/palladium>.
- [98] N. Marzari et al. “Thermal Contraction and Disordering of the Al(110) Surface Nicola”. In: *Condensed Matter* (110 2008), pp. 4–7.
- [99] G. Dimitri Ngantso and A. T. Raji. “Adsorption of Br₂ molecule on the Fe W(110) substrate - Energetics, electronic and magnetic properties”. In: *Computational Condensed Matter* 23 (2020).
- [100] A. E. Rezaee and M. Almasi-Kashi. “The influence of point defects on Na diffusion in black phosphorene - first principles study”. In: *Journal Pre-proof* ().
- [101] N. Acharya and S. P. Sanyal. “Structural phase transition, electronic and superconducting properties of ScBi and YBi”. In: *Solid State Communications* 266 (2017), pp. 39–45.
- [102] L. Malakkal and R. K. Siripurapu. “First principles calculations of hydrogen storage on Cu and Pd-decorated graphene”. In: *International Journal of Hydrogen Energy* 41 (2016), pp. 17652–17656.
- [103] P. O. Adebambo and G. A. Adebayo. “Elastic constants and observed ferromagnetism in inverse Heusler alloy Ti₂CoAs using kjpaw pseudopotentials - A first-principles approach”. In: *Journal of Alloys and Compounds* 722 (2017), pp. 207–211.
- [104] D. Rochman and A. J. Koning. “Modern Nuclear Data Evaluation With The TALYS Code System”. In: *Nuclear Data Sheets* 113 (2012).
- [105] G. J. Ackland, D. Y. Sun, and M. Asta. “Development of new interatomic potentials appropriate for crystalline and liquid iron”. In: *Philosophical Magazine* 83 (2003), pp. 3977–3994.
- [106] Jae-Hyeok Shim and M. I. Baskes. “Semiempirical atomic potentials for the fcc metals Cu, Ag, Au, Ni, Pd, Pt, Al, and Pb based on first and second nearest-neighbor modified embedded atom method”. In: *Physical Review B* 68 (2003).
- [107] G J Ackland et al. “Development of an interatomic potential for phosphorus impurities in alpha-iron”. In: *Condensed Matter* 16 (2004).

Chapter 13

Acknowledgements

Thank you to Brian, Mark and Alessandro for your supervision and support, to my wife Rachel, son Zakk and my family, and to the various forums in the scientific community, in particular useful discussions with Dr. Todorov of Daresbury and members of the Quantum Espresso forum.

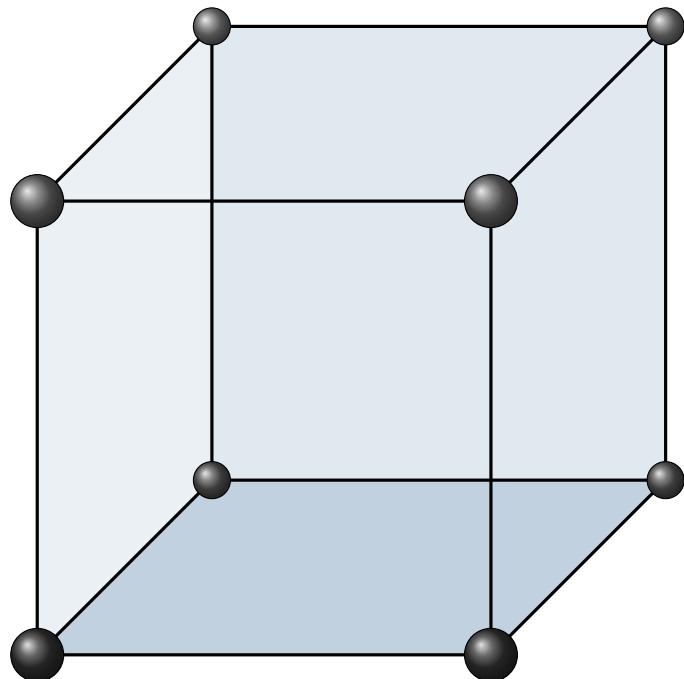
Appendices

Appendix A

Crystal Structures

A.1 Common Cubic Structures

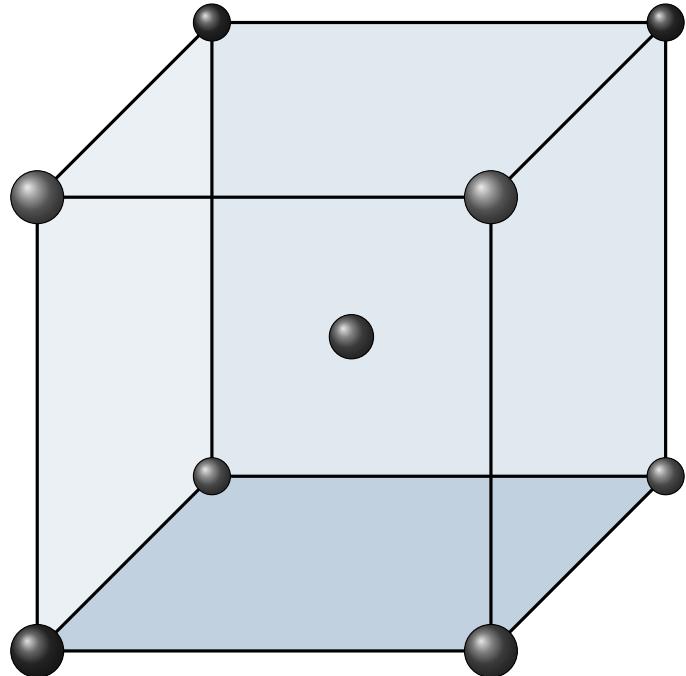
A.1.1 Simple Cubic



Sample Configuration
[0.0 0.0 0.0]

Table A.1

A.1.2 Body Centered Cubic

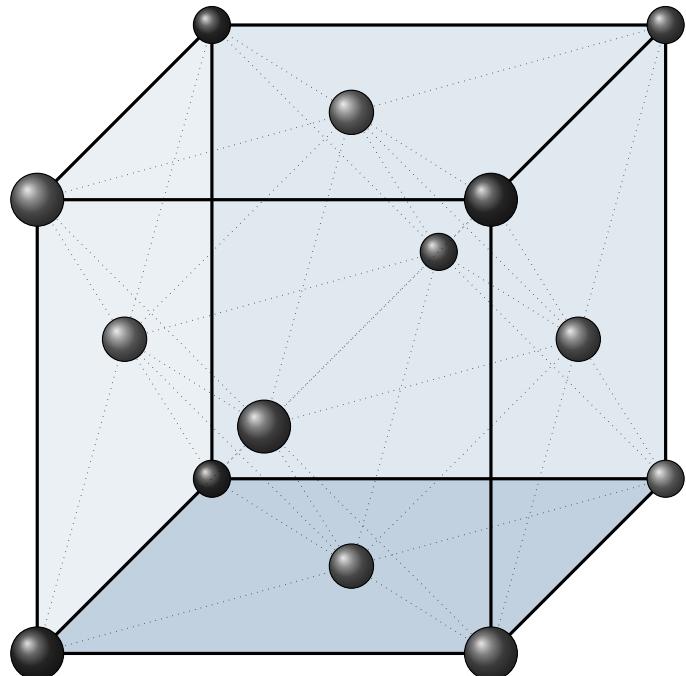


Sample Configuration

$$\begin{array}{ccc} [0.0 & 0.0 & 0.0] \\ [0.5 & 0.5 & 0.5] \end{array}$$

Table A.2

A.1.3 Face Centered Cubic

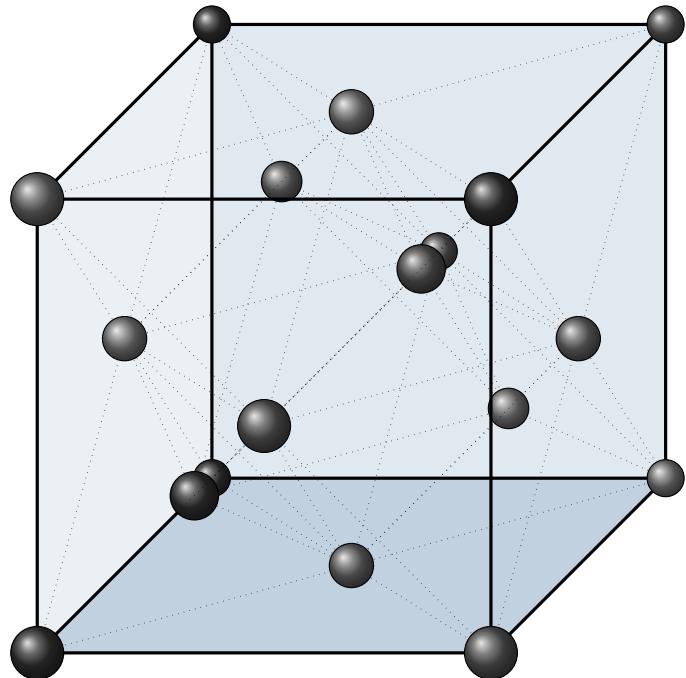


Sample Configuration

[0.0	0.0	0.0]
[0.0	0.5	0.5]
[0.5	0.0	0.5]
[0.5	0.5	0.0]

Table A.3

A.1.4 Zincblende



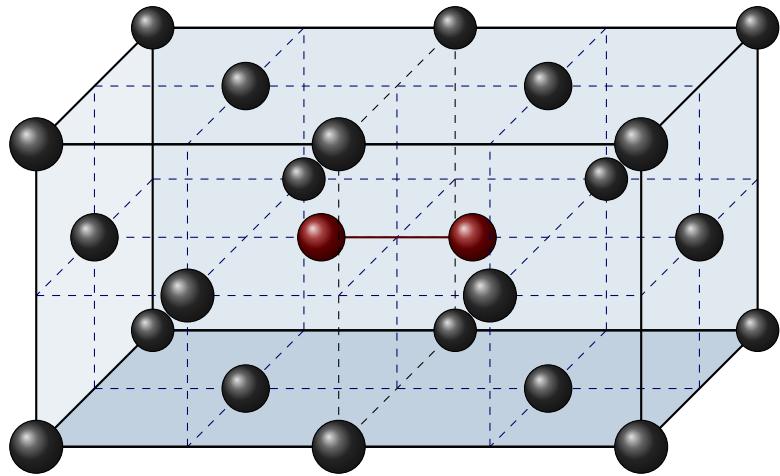
Sample Configuration

[0.0	0.0	0.0]
[0.0	0.5	0.5]
[0.5	0.0	0.5]
[0.5	0.5	0.0]
[0.25	0.25	0.25]
[0.75	0.75	0.25]
[0.25	0.75	0.75]
[0.75	0.25	0.75]

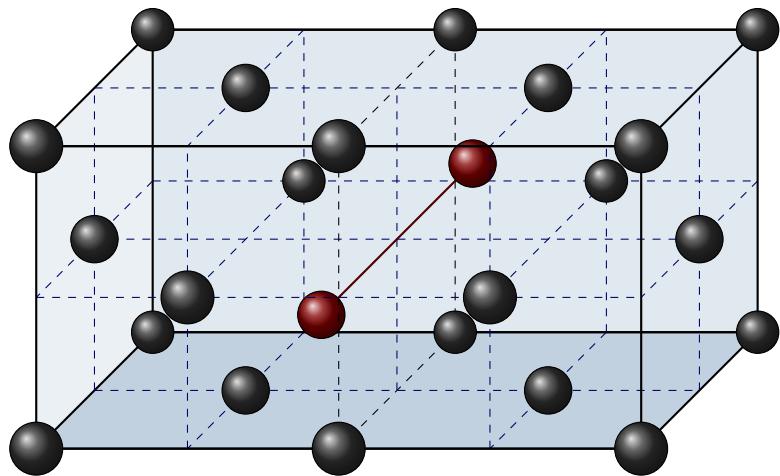
Table A.4

A.2 FCC Defects

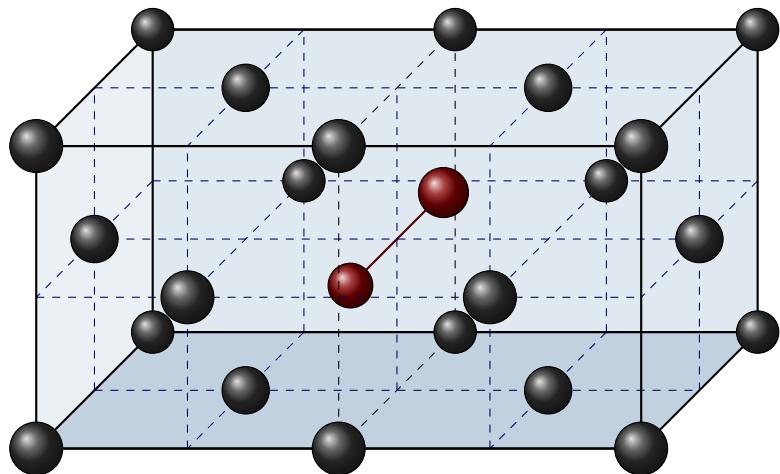
A.2.1 Dumbbell $<100>$



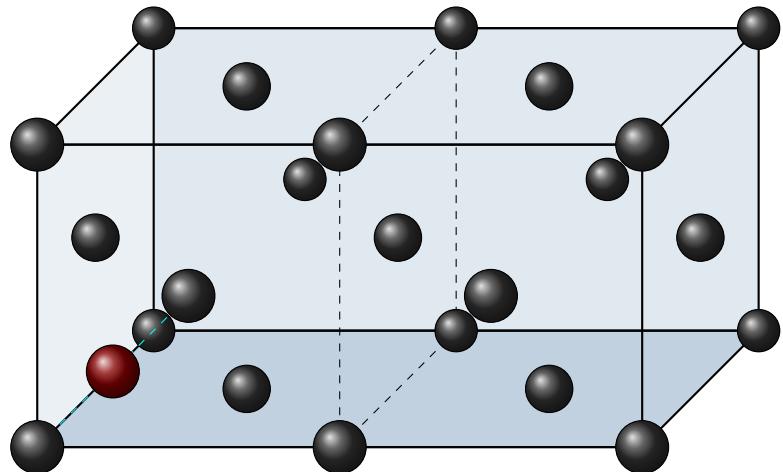
A.2.2 Dumbbell $<110>$



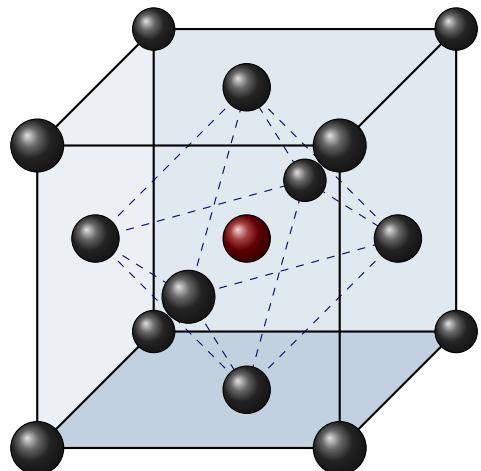
A.2.3 Dumbbell $<111>$



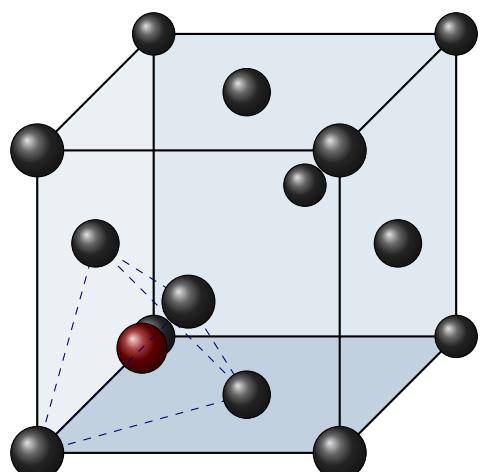
A.2.4 Crowdion



A.2.5 Octahedral Interstitial



A.2.6 Tetrahedral Interstitial



Appendix B

Elastic Constants

Crystal:	Cubic
Sides:	$a = b = c$
Angles:	$\alpha = \beta = \gamma = \frac{\pi}{2}$ radians
Symmetry:	$\frac{\pi}{2}$ radians rotation, 3 mutually orthogonal planes of symmetry
No. Independent Elastic Constants:	3
Elastic Constants:	$\begin{bmatrix} C_{11} & C_{12} & C_{12} & 0 & 0 & 0 \\ C_{12} & C_{11} & C_{12} & 0 & 0 & 0 \\ C_{12} & C_{12} & C_{11} & 0 & 0 & 0 \\ 0 & 0 & 0 & C_{44} & 0 & 0 \\ 0 & 0 & 0 & 0 & C_{44} & 0 \\ 0 & 0 & 0 & 0 & 0 & C_{44} \end{bmatrix}$

Table B.1

Crystal:	Orthorhombic/Orthotropic
Sides:	$a \neq b \neq c$
Angles:	$\alpha = \beta = \gamma = \frac{\pi}{2}$ radians
Symmetry:	3 mutually orthogonal planes of symmetry
No. Independent Elastic Constants:	9
Elastic Constants:	$\begin{bmatrix} C_{11} & C_{12} & C_{13} & 0 & 0 & 0 \\ C_{12} & C_{22} & C_{23} & 0 & 0 & 0 \\ C_{13} & C_{23} & C_{33} & 0 & 0 & 0 \\ 0 & 0 & 0 & C_{44} & 0 & 0 \\ 0 & 0 & 0 & 0 & C_{55} & 0 \\ 0 & 0 & 0 & 0 & 0 & C_{66} \end{bmatrix}$

Table B.2

Crystal:	Monoclinic
Sides:	$a \neq b \neq c$
Angles:	$\alpha < \frac{\pi}{2}$ radians, $\beta = \gamma = \frac{\pi}{2}$ radians
Symmetry:	symmetry in the xy plane
No. Independent Elastic Constants:	13
Elastic Constants:	$\begin{bmatrix} C_{11} & C_{12} & C_{13} & 0 & 0 & C_{16} \\ C_{12} & C_{22} & C_{23} & 0 & 0 & C_{26} \\ C_{13} & C_{23} & C_{33} & 0 & 0 & C_{36} \\ 0 & 0 & 0 & C_{44} & C_{45} & 0 \\ 0 & 0 & 0 & C_{45} & C_{55} & 0 \\ C_{16} & C_{26} & C_{36} & 0 & 0 & C_{66} \end{bmatrix}$

Table B.3

Crystal:	Triclinic
Sides:	$a \neq b \neq c$
Angles:	$\alpha < \frac{\pi}{2}$ radians, $\beta < \frac{\pi}{2}$ radians, $\gamma < \frac{\pi}{2}$ radians
Symmetry:	no symmetry
No. Independent Elastic Constants:	21
Elastic Constants:	$\begin{bmatrix} C_{11} & C_{12} & C_{13} & 0 & 0 & C_{16} \\ C_{12} & C_{22} & C_{23} & 0 & 0 & C_{26} \\ C_{13} & C_{23} & C_{33} & 0 & 0 & C_{36} \\ 0 & 0 & 0 & C_{44} & C_{45} & 0 \\ 0 & 0 & 0 & C_{45} & C_{55} & 0 \\ C_{16} & C_{26} & C_{36} & 0 & 0 & C_{66} \end{bmatrix}$

Table B.4

Appendix C

Activity V1

C.1 Sample Input File

Listing C.1: Activity V1 Input File

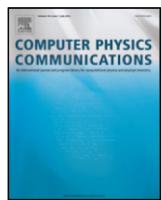
```
1 #elements
2 Fe 100
3 #isotopes
4 "/home/ben/activity/data/isotopes.txt"
5 #decaymodes
6 "/home/ben/activity/data/decaymodes.txt"
7 #gammaenergies
8 "/home/ben/activity/data/gammaenergies.txt"
9 #xsfiles
10 "/home/ben/activity/data/xs"
11 #trajfile
12 "Fe36MeV.exyz"
13 #polyfitorder
14 5
15 #integrationgranularity
16 10
17 #beamflux
18 0.5 uA
19 #beamenergy
20 36 MeV
21 #beadmduration
22 300 s
23 #beamarea
24 100 mm2
25 #amtime
26 260000 s
27 #timestep
28 1000 s
29 #projectile
30 1 1
31 #targetthickness
32 0.5 mm
33 #materialdensity
34 8000 kgm3
35 #vpi
36 60.2
37 #individualisotopeactivity
38 yes
39 #verboseterminal
```

```
40 yes
41 #targetdpa
42 0.0
43 #gammachartresolution
44 200
```

C.2 Output Plots

The Activity V1 code and the above input file generates a plot of predicted gammas at the time specified in the input file.

C.3 Paper published in Computer Physics Communications



Activity computer program for calculating ion irradiation activation[☆]

Ben Palmer ^{*}, Brian Connolly, Mark Read

University of Birmingham, United Kingdom



ARTICLE INFO

Article history:

Received 1 June 2015

Received in revised form 15 February 2017

Accepted 19 February 2017

Available online 29 March 2017

ABSTRACT

A computer program, Activity, was developed to predict the activity and gamma lines of materials irradiated with an ion beam. It uses the TENDL (Koning and Rochman, 2012) [1] proton reaction cross section database, the Stopping and Range of Ions in Matter (SRIM) (Biersack et al., 2010) code, a Nuclear Data Services (NDS) radioactive decay database (Sonzogni, 2006) [2] and an ENDF gamma decay database (Herman and Chadwick, 2006) [3]. An extended version of Bateman's equation is used to calculate the activity at time t, and this equation is solved analytically, with the option to also solve by numeric inverse Laplace Transform as a failsafe. The program outputs the expected activity and gamma lines of the activated material.

Program summary

Program title: Activity

Catalogue identifier: AFBS_v1_0

Program summary URL: http://cpc.cs.qub.ac.uk/summaries/AFBS_v1_0.html

Program obtainable from: CPC Program Library, Queen's University, Belfast, N. Ireland

Licensing provisions: GNU GPL v3

No. of lines in distributed program, including test data, etc.: 688828

No. of bytes in distributed program, including test data, etc.: 71056048

Distribution format: tar.gz

Programming language: Fortran.

Computer: PCs or HPCs.

Operating system: Linux (tested on Debian).

Has the code been vectorized or parallelized?: OpenMPI

RAM: 250MB per process + 200MB overhead

Classification: 2.2, 17.8.

Nature of problem: To calculate the predicted activity of an ion irradiated target. The expected range of ion energies is between 1MeV and 200MeV; this is the range of the available ion cross section data.

Solution method: The program loads cross section data from the TENDL database and trajectory data from a SRIM [1] simulation xyz data file. It uses this data to calculate the production/loss rate of each isotope in the simulated target. Radioactive decay equations are used to calculate the amounts and activity of each radioactive isotope at the set time.

Running time: Typically the Activity program runs each input from seconds to no more than several minutes.

References:

- [1] SRIM — The stopping and range of ions in matter (2010). Ziegler, James F., Ziegler, M.D. and Biersack, J.P. 2010, Nuclear Instruments and Methods in Physics Research Section B, Vol. 268, pp. 1818–1823.

© 2017 Elsevier B.V. All rights reserved.

[☆] This paper and its associated computer program are available via the Computer Physics Communication homepage on ScienceDirect (<http://www.sciencedirect.com/science/journal/00104655>).

* Corresponding author.

E-mail address: benpalmer1983@gmail.com (B. Palmer).

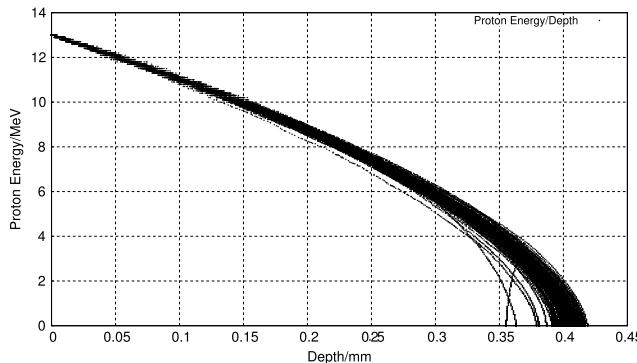


Fig. 1. One hundred simulated 13 MeV proton energy loss curves in Fe simulated with SRIM [4].

1. Background

1.1. Ion irradiation at the University of Birmingham

The Scanditronix MC-40 Cyclotron is used at the University of Birmingham to create a beam of protons or other light ions. The energies of these ions are typically between 10 MeV and 60 MeV with beam currents ranging up to 50 μA (3.1×10^{14} protons per second). Target materials are irradiated by this cyclotron for a number of reasons, including purposely creating radioactive isotopes for the nearby Queen Elizabeth Hospital, investigating ion irradiation damage and emulating neutron irradiation.

The cyclotron is usually used to create radioactive isotopes for medical use, but an additional beam line has been devoted to material science investigations into radiation damage. While the creation of radioactive isotopes is desired in some cases, material being tested for radiation damage should preferably have low levels of radioactivity.

It is expensive to arrange the irradiation of target materials by high energy neutrons sources, whereas it is relatively inexpensive to irradiate using an ion beam on the MC-40 Cyclotron. The energies can be controlled, and a set dose at a single energy, or a range of energies, can be precisely deposited into the target material.

The Activity code discussed here was developed to calculate the activity of a target material irradiated by a proton beam. It has been developed in Fortran and uses data from the TENDL-2013 proton cross section database, SRIM ion transport code and NDS radioactive decay database.

1.2. Simulating ion irradiation with SRIM

A package of ion transport codes, SRIM, is freely available to download and use to investigate the transport of ions through matter. SRIM uses the binary collision approximation (BCA) to simulate the passage of ions in a material. It is an approximate method, and one key restriction is that it does not take into account the structure of the material, and this approximation is therefore also imposed on the Activity code.

One file that SRIM creates is of importance to the Activity code, and that is the trajectory file that contains the energy and x , y , z co-ordinate data points for simulated ions moving through matter. Fig. 1 shows the trajectory of one hundred 13 MeV protons entering and passing through an Iron target, and it is this set of data points (together with the cross section database) that the Activity code uses to calculate the reaction rates for the transmutation of nuclei in the target. At higher energies, the ions slow as they lose energy due to electronic stopping, but as the ion energy drops the mechanism of loss through nuclear collisions becomes important.

The spreading of ion depths at lower energies is a result of the higher momentum transfer during nuclear collisions, as can be seen in Fig. 1.

1.3. Transmutation of nuclei by ion irradiation

Considering a simplified nuclear potential well, energetic protons approaching a nucleus may overcome the Coulomb potential barrier. They are captured by the nucleus and held within the potential well by the strong nuclear force. This process may leave the nucleus in an excited and unstable state, depending on the input energy of the proton and configuration of nucleons. The process is probabilistic, and the average chance of a reaction (the microscopic cross section) may be measured as a function of the projectile, projectile energy and target, either experimentally or by optical model potential calculations. The reaction rate is calculated from the microscopic cross section using the following equation:

$$R = \frac{J}{e} n_t \sigma \cdot 10^{-28} \delta t \quad (1)$$

- R Reaction Rate (reactions per second)
- J Beam current (A)
- n_t Number density of target (atoms per cubic meter)
- σ Microscopic reaction cross section (barns)
- e Elementary charge (1.602177E-19C)
- δt Target thickness (m).

1.4. Radioactive decay

Radioactive decay is the random change in nucleons or energy state of an unstable nucleus. It is impossible to predict when a single nucleus will decay, but the decay of a collection of nuclei is statistical in nature. The radioactivity and number of unstable nuclei at time t can be predicted using the decay constant, λ , for the radioactive isotope. This constant is defined as follows:

$$\lambda = -\frac{N'(t)}{N(t)}. \quad (2)$$

The number of radioactive nuclei $N(t)$ at time t is given by the following equation, where $N(0)$ is the starting number of nuclei:

$$N(t) = N(0) \exp(-t\lambda). \quad (3)$$

The activity $A(t)$ of the radioactive nuclei is predicted at time t by using the following equations, where $N'(t)$ is the change in amount of nuclei with respect to time:

$$A(t) = -N'(t) = \lambda N(t) \quad (4)$$

$$A(t) = \lambda N(0) \exp(-t\lambda). \quad (5)$$

1.5. Bateman equation for radioactive decay

The English mathematician Harry Bateman derived an Eq. (6) to calculate the amount of each isotope in a decay chain, illustrated in Fig. 2, at time t .

$$N_n(t) = \sum_{i=1}^{i=n} \left(\left(\prod_{j=i}^{j=n-1} \lambda_{(ij+1)} \right) \sum_{j=i}^{j=n} \left(\frac{N_{i0} \exp(-\lambda_j t)}{\prod_{p=i, p \neq j}^{p=n} (\lambda_p - \lambda_j)} \right) \right). \quad (6)$$

When a radioactive isotope decays, there may be more than one mode of decay, and this leads to branching factors. Pb-214 only decays via beta decay to Bi-214, giving a branching factor of 1.0, whereas Bi-214 has a 99.979% chance of decaying to Po-214 by beta decay and a 0.021% of emitting an alpha particle and decaying to Tl-210 (branching factors of 0.99979 and 0.00021 respectively) [5].

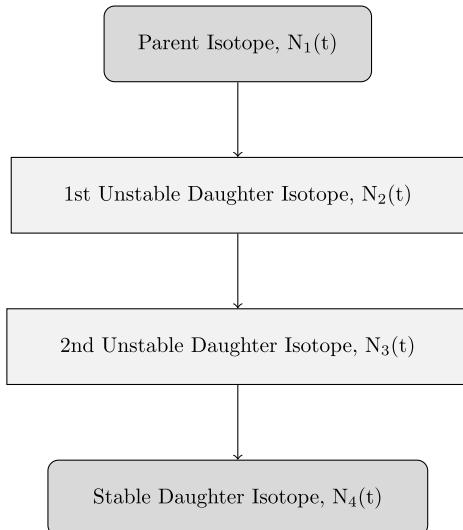


Fig. 2. An example decay chain from an unstable parent isotope, through unstable daughter isotopes ending with a stable daughter isotope.

When a target material is irradiated, there is a source term for transmuted nuclei due to the irradiation. The daughter isotopes of these transmuted isotopes will also be affected by the irradiation and will transmute further, giving a source term for each daughter isotope as a result of the irradiation. Sources for each isotope in the decay chain, and branching factors between a parent isotope and its daughter isotope/s must be accounted for.

The Bateman equation was derived using Laplace transforms, and this same method has been used to develop a modified equation that incorporates branching factors and production rates for each isotope in the decay chain, as illustrated by Fig. 3.

1.6. Laplace transform

Laplace Transforms (7) are a useful mathematical tool, and allow ordinary differential equations to be solved by simple algebraic manipulation in the s domain. Bateman took advantage of Laplace Transforms in deriving his equation, and this is the method that has been taken here as well.

$$F(s) = \int_0^\infty f(t) \exp(-st) dt. \quad (7)$$

1.7. Constructing the differential equations

The first step is to set up differential equations for the parent isotope, unstable daughter isotopes and stable daughter isotope. The parent isotope has a source term, due to production, and a loss term, due to decay. The unstable daughter isotopes have two source terms, from the production by irradiation induced transmutation and the decay of preceding isotopes in the decay chain, and a loss term, due to decay. Finally, the stable daughter that finalizes the decay chain has two source terms (the same as the unstable daughters) but no loss term.

The variables (and vectors) used in these equations are defined as follows:

- $\vec{\lambda}$ vector containing isotope decay constants λ_i
- \vec{b} vector containing isotope to isotope branching factors b_i
- \vec{w} vector containing isotope production rates w_i
- t time at which activity/amount of isotope is measured
- $N_i(0)$ starting amount of the i th isotope

- $N_i(t)$ amount of the i th isotope at time t
- $N'_i(t)$ change in amount of the i th isotope, with respect to time, at time t .

The differential equations for the parent isotope (first isotope), unstable daughter isotopes (i th isotopes) and stable, final, daughter isotope (z th isotope) in the time domain are as follows:

$$N'_1(t) = \omega_1 - \lambda_1 N_1(t) \quad (8)$$

$$N'_i(t) = \omega_i + b_{i-1} \lambda_{i-1} N_{i-1}(t) - \lambda_i N_i(t) \quad (9)$$

$$N'_z(t) = \omega_z + b_{z-1} \lambda_{z-1} N_{z-1}(t). \quad (10)$$

Applying the Laplace Transform to these three differential equations allows them to be manipulated and solved algebraically in the s-domain.

$$N_1(s) = \frac{1}{s + \lambda_1} N_1(0) + \frac{1}{s(s + \lambda_1)} \omega_1 \quad (11)$$

$$N_i(s) = \frac{1}{s(s + \lambda_i)} (\omega_i) + \frac{1}{s + \lambda_i} (b_{i-1} \lambda_{i-1} N_{i-1}(s)) \\ + \frac{1}{s + \lambda_i} N_i(0) \quad (12)$$

$$N_z(s) = \frac{1}{s^2} \omega_z + \frac{1}{s} b_{z-1} \lambda_{z-1} N_{z-1}(s) + \frac{1}{s} N_z(0). \quad (13)$$

1.8. Numerical inversion of the Laplace Transform

The Gaver–Stehfest [6] algorithm was developed in the 1960s and 1970s and is a method of calculating the inverse of a Laplace Transform in the real number domain. It is an easy to implement and reasonably accurate method, although it is an approximation to the real value. A comparison between an analytic and numeric inversion for the unstable isotope Po-218 is discussed at the end of this section (Fig. 4).

$$f(t) \approx f_n(t) = \frac{\ln(2)}{t} \sum_{k=1}^{2n} a_k(n) F(s) \text{ where } n \geq 1, t > 0 \quad (14)$$

$$s = \frac{k \ln(2)}{t} \quad (15)$$

$$a_k(n) = \frac{(-1)^{(n+k)}}{n!} \sum_{j=\text{Floor}(\frac{k+1}{2})} j^{n+1} \binom{n}{j} \binom{2j}{j} \binom{j}{k-j}. \quad (16)$$

The equation for the i th isotope may be calculated by recursively calculating the equations by numeric inversion, starting from the first (parent isotope) and inserting the result into each subsequent recursion until the i th isotope is reached (changing the equations appropriately for the parent, unstable daughter and stable daughter isotopes).

1.9. Analytic solution by partial fraction expansion

The equation for the i th isotope in the s domain can be written in full by substituting the preceding equation until the parent isotope is reached, and this full equation may be rearranged with the production amount of each isotope and starting amount of each isotope in individual terms. Each of these terms is multiplied by a fraction that can be expanded, using partial fractions, and inverted analytically.

This is illustrated with an example unstable isotope, fourth in the decay chain (including the parent isotope):

$$N_4(s) = \frac{1}{(s + \lambda_1)(s + \lambda_2)(s + \lambda_3)(s + \lambda_4)} b_2 b_3 b_4 \lambda_1 \lambda_2 \lambda_3 N_1(0)$$

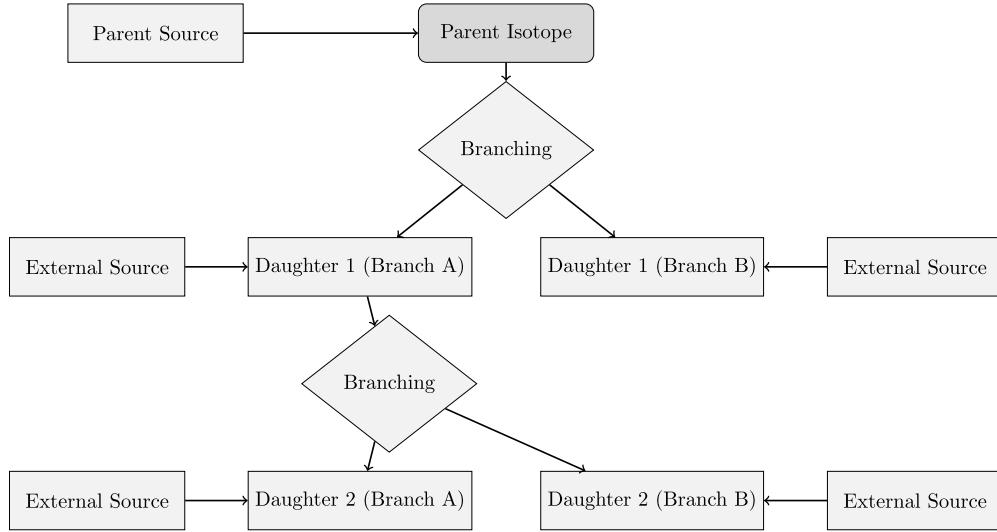


Fig. 3. An example of several decay chains including branching factors and possible external source terms for each isotope on each chain.

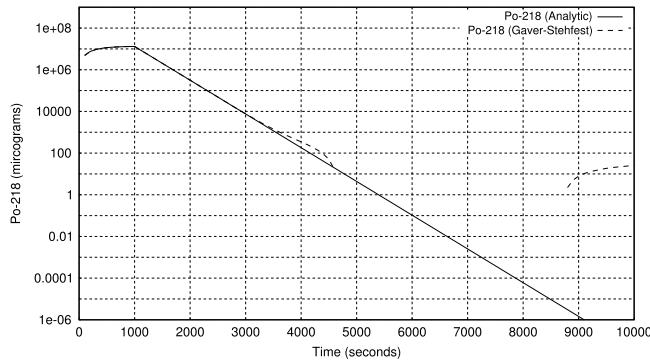


Fig. 4. Decay of Po-218: Analytic and Gaver–Stehfest Calculation [5].

$$\begin{aligned}
 & + \frac{1}{(s + \lambda_2)(s + \lambda_3)(s + \lambda_4)} b_3 b_4 \lambda_2 \lambda_3 N_2(0) \\
 & + \frac{1}{(s + \lambda_3)(s + \lambda_4)} b_4 \lambda_3 N_3(0) + \frac{1}{(s + \lambda_4)} N_4(0) \\
 & + \frac{1}{s(s + \lambda_1)(s + \lambda_2)(s + \lambda_3)(s + \lambda_4)} b_2 b_3 b_4 \lambda_1 \lambda_2 \lambda_3 \omega_1 \\
 & + \frac{1}{s(s + \lambda_2)(s + \lambda_3)(s + \lambda_4)} b_3 b_4 \lambda_2 \lambda_3 \omega_2 \\
 & + \frac{1}{s(s + \lambda_3)(s + \lambda_4)} b_4 \lambda_3 \omega_3 + \frac{1}{s(s + \lambda_4)} \omega_4. \quad (17)
 \end{aligned}$$

An example stable isotope, fourth (last) in the decay chain (including the parent isotope):

$$\begin{aligned}
 N_4(s) = & \frac{1}{s(s + \lambda_1)(s + \lambda_2)(s + \lambda_3)} b_2 b_3 b_4 \lambda_1 \lambda_2 \lambda_3 N_1(0) \\
 & + \frac{1}{s(s + \lambda_2)(s + \lambda_3)} b_3 b_4 \lambda_2 \lambda_3 N_2(0) \\
 & + \frac{1}{s(s + \lambda_3)} b_4 \lambda_3 N_3(0) + N_4(0) \\
 & + \frac{1}{s^2(s + \lambda_1)(s + \lambda_2)(s + \lambda_3)} b_2 b_3 b_4 \lambda_1 \lambda_2 \lambda_3 \omega_1 \\
 & + \frac{1}{s^2(s + \lambda_2)(s + \lambda_3)} b_3 b_4 \lambda_2 \lambda_3 \omega_2
 \end{aligned}$$

$$+ \frac{1}{s^2(s + \lambda_3)} b_4 \lambda_3 \omega_3 + \frac{1}{s^2} \omega_4. \quad (18)$$

By using partial fraction expansion and standard Laplace Transforms, the set of equations below is used to calculate the amount of the m th isotope in the decay chain, providing the m th isotope is unstable.

$$\begin{aligned}
 N_m(t; \vec{\lambda}, \vec{b}, \vec{w}) \\
 = \sum_{k=1,m} r(k; \vec{\lambda}, \vec{b}) [f(t; k, m, \vec{\lambda}) N_k(0) + g(t; k, m, \vec{\lambda}) w_k] \quad (19)
 \end{aligned}$$

$$r(k, m, \vec{\lambda}) = \begin{cases} \prod_{i=k,m-1} (b_{i+1} \lambda_i), & \text{if } k < m \\ 1, & \text{if } k = m \end{cases} \quad (20)$$

$$\begin{aligned}
 f(t; k, m, \vec{\lambda}) \\
 = (-1)^{m-k} \sum_{i=k,m} \left[\exp(-\lambda_i t) \prod_{j=k,m;j \neq i} \left(\frac{1}{\lambda_i - \lambda_j} \right) \right] \quad (21)
 \end{aligned}$$

$$\begin{aligned}
 g(t; k, m, \vec{\lambda}) \\
 = \frac{1}{\prod_{i=k,m} \lambda_i} + (-1)^{m-k+1} \\
 \times \sum_{i=k,m} \left[\frac{1}{\lambda_i} \exp(-\lambda_i t) \prod_{j=k,m;j \neq i} \left(\frac{1}{\lambda_i - \lambda_j} \right) \right]. \quad (22)
 \end{aligned}$$

The set of equations below is used to calculate the amount of the m th isotope in the decay chain, where the m th isotope is stable.

$$\begin{aligned}
 N_m(t; \vec{\lambda}, \vec{b}, \vec{w}) = N_m + w_m t + \sum_{k=1,m-1} r(k; \vec{\lambda}, \vec{b}) \\
 \times [f(t; k, m-1, \vec{\lambda}) N_k(0) + g(t; k, m, \vec{\lambda}) w_k] \quad (23)
 \end{aligned}$$

$$r(k, m, \vec{\lambda}) = \begin{cases} \prod_{i=k,m-1} (b_{i+1} \lambda_i), & \text{if } k < m \\ 1, & \text{if } k = m \end{cases} \quad (24)$$

$$\begin{aligned}
 f(t; k, m, \vec{\lambda}) \\
 = \frac{1}{\prod_{i=k,m} \lambda_i} + (-1)^{m-k+1} \\
 \times \sum_{i=k,m} \left[\frac{1}{\lambda_i} \exp(-\lambda_i t) \prod_{j=k,m;j \neq i} \left(\frac{1}{\lambda_i - \lambda_j} \right) \right] \quad (25)
 \end{aligned}$$

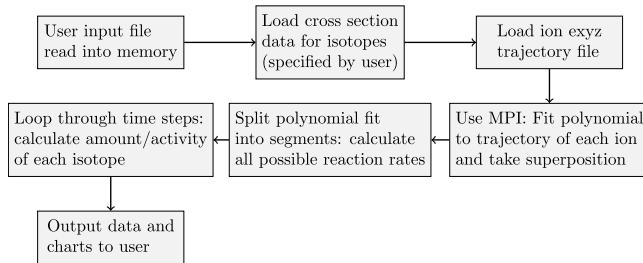


Fig. 5. Flow chart of major processes in the Activity code.

$$g(t; k, m, \vec{\lambda}) = \frac{1}{\prod_{i=k,m} \lambda_i} t + \frac{\sum_{i=k,m} \left[\prod_{j=k,m;j \neq i} \lambda_j \right]}{\prod_{i=k,m} \lambda_i^2} + (-1)^{m-k+1} \sum_{i=k,m} \left[\frac{1}{\lambda_i^2} \exp(-\lambda_i t) \prod_{j=k,m;j \neq i} \left(\frac{1}{\lambda_i - \lambda_j} \right) \right]. \quad (26)$$

1.10. Preference: analytic over numeric

The numeric solution only requires the equation to be solved in the s-domain; the Gaver–Stehfest algorithm performs the inversion. It is worth the extra effort to derive and implement an analytic solution, as the numeric is only an approximation. Examples of the pitfalls of the numeric solution are that it can give negative amounts of an isotope and the difference between the numeric and analytic calculated amounts can become quite large when the isotope decays away to a very small value. Fig. 4 shows the predicted decay of a sample of Po-218 irradiated for 1000 s, and sampled until 10,000 s. In the region between 4000 s and 9000 s the amount from the numeric calculation drops below zero, whereas the analytic calculation remains above zero, as would be expected.

2. Computational methods

The Activity program has been developed in Fortran and takes advantage of MPI (Message Passing Interface) to speed up calculation times by allowing the use of multiple processes in parallel. It has a self contained maths library, although this could be improved in the future by using optimized maths libraries for certain functions (e.g. matrix operations).

The code was developed on a Debian based distribution of Linux, but it should be supported on other variants of Linux and Unix, and does not require any specialist hardware.

The user is required to prepare an input file that contains the instructions required to perform a calculation. In addition to the input file, the user must provide an EXYZ ion trajectory file output by SRIM. Activity will read in the user input file, and the SRIM and data files listed within, before performing the calculation. Fig. 5 shows a flowchart of the major steps the code performs.

There are various settings in the user input file, but the main ones relating to the simulated experiment are:

- Element composition of target (percentage by mass).
- Beam flux (current), energy, duration and area on target.
- Activity measurement time (end of the “experiment”).
- Material density.
- Target thickness.

Several data files are generated by Activity and, if the user has matplotlib [7], charts will be created too. The most relevant to the user are:

- gammaLines.dat – tally into discrete bins of predicted gamma counts.
- ionTraj.dat – the averaged ion trajectory used in the calculation.
- isotopeActivityFileG.dat – a large data file detailing the activity of every predicted radioactive isotope in the target at user specified times following irradiation.

The charts include:

- activityTop5.png – activity of the top 5 active isotopes as a function of time after irradiation starts.
- gammaLines.png – predicted gamma spectrum expected at the “experiment end time”.

The Activity code uses the equations derived above to calculate the amount and activity of each isotope in the calculation. One problem with the original Bateman equation that also exists in the set of modified Bateman equations is that two different isotopes with the same decay constant will cause a singularity and a halt in the calculation. The activity code loops through all the decay constants in use before it attempts to run the calculation. If any isotope decay constants match they are varied by a small amount relative to the decay constant. It repeats this process until all decay constants are unique before proceeding.

3. Approximations

The accuracy of the Activity code is dependent on the input files provided by the user and the method used to calculate the reaction rates and resulting activity. The TENDL proton database consists of experimental measured cross sections as well as values calculated using the optical model potential. Using the latest database is recommended.

SRIM uses the binary collision approximation to simulate ion transport. It is a well tested code that has been used for many years. One limitation is that the structure of the material is not taken into account. This would have an impact on a user of the Activity program if they were trying to calculate, for example, whether a FCC (face centered cubic) steel would be irradiated differently when compared to a BCC (body centered cubic) steel. The Activity code would determine the activity of the steel as a function of the ion current, ion type and the density, thickness and composition of the steel, not its structure.

This version of the Activity code averages the path of all the SRIM simulated ions, rather than treating each ion differently. This may or may not have an impact on the results. If a new version of the code is developed there would be an option to calculate reaction rates for each individual simulated ion, and a comparison could then be made to the calculations using the averaged path of a set of ions.

The final approximation would be to use the numeric solution to the activity equations, although the analytic solution is forced within the code unless it returns a failed result.

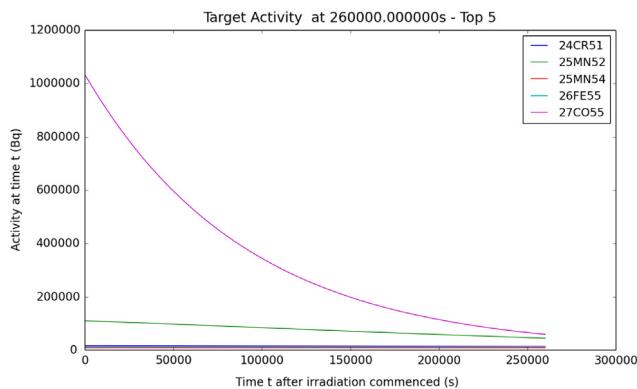
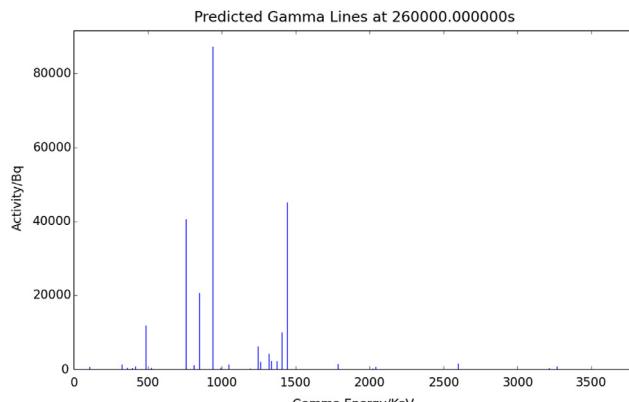
4. Results

A target of high purity Iron was irradiated with 36 MeV protons by the University of Birmingham Scanditronix MC-40 Cyclotron. The target was 0.5 mm thick and was irradiated at a current of 0.5 μ A for 300 s, irradiating approximately 0.25 g of Iron. A high purity Germanium detector was used to measure the gamma peaks three days after irradiation.

The peak that dominated the readings was the 931 keV Cobalt 55 line. After calibrating the detector and adjusting the readings, this peak was measured at 44,300 Bq+/−2000 Bq. The activity of this peak as predicted by the Activity code was 44,565 Bq.

Table 1

Gamma energy (keV)	Predicted activity (Bq)	Experimental activity (Bq)
766	4.45E6	5.11E5+/-2.5E4
778	6.14E6	1.36E6+/-6.8E4
812	5.04E6	1.15E6+/-5.8E4
850	6.00E6	1.39E6+/-7.0E4
126	9.33E5	2.10E5+/-1.1E4

**Fig. 6.** Sample Activity code output chart for the top five most active isotopes for Iron irradiated by 36 MeV protons.**Fig. 7.** Sample Activity code output chart for the expected gamma lines to be measured for Iron irradiated by 36 MeV protons.

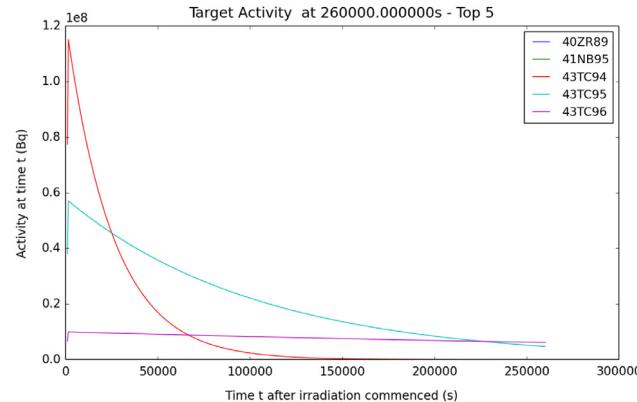
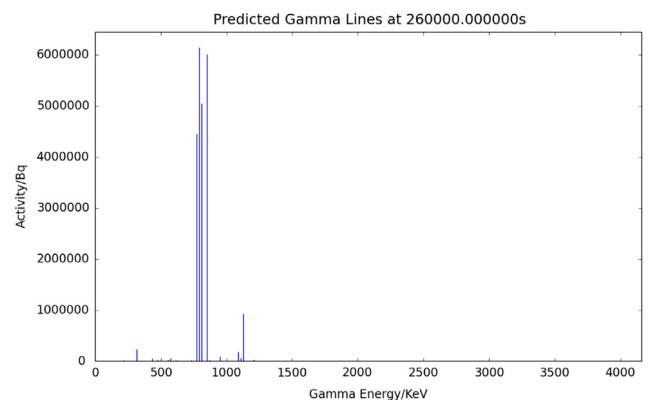
A target of high purity Molybdenum was irradiated with 13 MeV protons by the University of Birmingham Scanditronix MC-40 Cyclotron. The target was 0.5 mm thick and was irradiated at a current of 5 μ A for 1500 s, irradiating approximately 0.3 g of Molybdenum. A high purity Germanium detector was used to measure the gamma peaks three days after irradiation.

Five peaks were of interest, and these are listed in Table 1.

There was the possibility of the high purity Germanium detector introducing errors due to detector dead time, where by a source that is too active floods the device with gammas. The samples were safe to handle, and did not appear to flood the detector in any way, but there was still the possibility of counts being missed due to dead time. The probabilistic nature of radioactive decay also introduced inherent errors to the experimental activity measurements (see Figs. 6–9).

5. Conclusions

The Activity program is an easy to use Fortran compiled executable that can be built and run, for free, on a Linux computer. It

**Fig. 8.** Sample Activity code output chart for the top five most active isotopes for Molybdenum irradiated by 13 MeV protons.**Fig. 9.** Sample Activity code output chart for the expected gamma lines to be measured for Molybdenum irradiated by 13 MeV protons.

takes advantage of multi-core processors, and typical calculations take from seconds to minutes.

Using the SRIM code and TENDL database, the code has been used to predict the activity of Iron and Molybdenum targets that have been irradiated with a proton beam. The prediction of the 931 (keV) Cobalt 55 gamma activity of irradiated Iron was very close to the measured value. The five predicted gamma activities for ion irradiated Molybdenum were up to a factor of 5–10 away from the measured activities.

There are a number of improvements that would be considered in a future version of the Activity code. These improvements would include the following:

- A more readable output with less information printed, as too much may confuse the user.
- A file containing the top five radioactive isotopes with their top five gamma lines.
- Individual ion trajectories used to calculate the reaction rates, rather than the average path.
- Experimental ion activation data for a wider range of elements to test the Activity code against.
- Expand to include deuterons (this would require the TENDL deuteron cross section database).

Acknowledgments

We would like to thank and acknowledge the input and advice of the following:

- Dr Chris Cooper and John Hewett for irradiation activation data points.

- The University of Birmingham for providing the funding for this project.

References

- [1] A.J. Koning, D. Rochman, *Nucl. Data Sheets* 113 (2012).
- [2] A.A. Sonzogni, Nuclear Data Services. www-nds.iaea.org. 2006. URL: <http://www-nds.iaea.org/ndspub/download-endf/ENDF-B-VII.0/decay-index.htm> (visited on 08/14/2015).
- [3] M. Herman, M.B. Chadwick, *Nucl. Data Sheets* 107 (2006) 2931.
- [4] J.P. Biersack, J.F. Ziegler, M.D. Ziegler, *Nuclear Instrum. Methods Phys. Res. B* 268 (2010) 1818–1823.
- [5] P. Blaise, M. Coste, A. Courcelle, T.D. Huynh, C. Jouanne, P. Leconte, O. Litaize, S. Mengelle, G. Nogure, J.-M. Ruggiri, O. Srot, J. Tommasi, C. Vaglio, J.-F. Vidal, A. Santamarina, D. Bernard, The JEFF-3.1.1 Nuclear Data Library. 2009. ISBN: 978-92-64-99074-6.
- [6] H. Stehfest, *Commun. ACM* 13 (1970) 47–49.
- [7] J.D. Hunter, *Comput. Sci. Eng.* 9 (2007) 90–95.

C.4 Accompanying Manual

UNIVERSITY OF BIRMINGHAM



Department of Metallurgy & Materials

Activity Manual

Contents

1	Background	4
1.1	Ion Irradiation	4
1.1.1	Ion Irradiation at the University of Birmingham	4
1.1.2	Simulating Ion Irradiation with SRIM	4
1.1.3	Transmutation of Nuclei by Ion Irradiation	5
1.2	Decay and Activity Equations	6
1.2.1	Radioactive Decay	6
1.2.2	Bateman Equation for Radioactive Decay	6
1.2.3	Laplace Transform	7
1.2.4	Constructing the Differential Equations	7
1.2.5	Numerical Inversion of the Laplace Transform	8
1.2.6	Analytic Solution by Partial Fraction Expansion	9
1.2.7	Preference: Analytic over Numeric	11
1.3	Computational Methods	12
1.3.1	Activity Code	12
1.3.2	Approximations	13
1.3.3	Results	13
2	Installation and Using the Activity Code	17
2.1	Getting Started	17
2.1.1	Prerequisites	17
2.2	Installing Activity	17
2.2.1	Download Source Code	17
2.2.2	Compile Source Code	18
2.3	Input File	18
2.4	Acknowledgements	21
Appendices		22

A Example Input File	23
A.1 Iron 36MeV Proton Beam	23
B Fortran 90 Code	25
B.1 Fortran 90 Implementation of Analytic Method	25

List of Figures

1.1	Proton energy loss in Fe simulated with SRIM [2]	5
1.2	An example decay chain from an unstable parent isotope, through unstable daughter isotopes ending with a stable daughter isotope.	6
1.3	An example of several decay chains including branching factors and possible external source terms for each isotope on each chain.	7
1.4	Decay of Po-218: Analytic and Gaver-Stehfest Calculations [5]	11
1.5	Flow chart of major processes in the Activity code	12
1.6	Sample Activity code output chart for the top five most active isotopes for Iron irradiated by 36MeV protons.	14
1.7	Sample Activity code output chart for the expected gamma lines to be measured for Iron irradiated by 36MeV protons.	15
1.8	Sample Activity code output chart for the top five most active isotopes for Molybdenum irradiated by 13MeV protons.	15
1.9	Sample Activity code output chart for the expected gamma lines to be measured for Molybdenum irradiated by 13MeV protons.	16

Chapter 1

Background

1.1 Ion Irradiation

A computer program, Activity, was developed to predict the activity and gamma lines of materials irradiated with an ion beam. It uses the TENDL[1] proton reaction cross section database, the Stopping and Range of Ions in Matter (SRIM)[2] code, a Nuclear Data Services (NDS) radioactive decay database [3] and an ENDF gamma decay database [4]. An extended version of Bateman's equation is used to calculate the activity at time t, and this equation is solved analytically, with the option to also solve by numeric inverse Laplace transform as a failsafe. The program outputs the expected activity and gamma lines of the activated material.

1.1.1 Ion Irradiation at the University of Birmingham

The Scanditronix MC-40 Cyclotron is used at the University of Birmingham to create a beam of protons or other light ions. The energies of these ions are typically between 10 MeV and 60 MeV with beam currents ranging up to 50 microamps (3.1×10^{14} protons per second). Target materials are irradiated by this cyclotron for a number of reasons, including purposely creating radioactive isotopes for the nearby Queen Elizabeth Hospital, investigating ion irradiation damage and emulating neutron irradiation.

The Cyclotron is usually used to create radioactive isotopes for medical use, but an additional beam line has been devoted to material science investigations into radiation damage. While the creation of radioactive isotopes is desired in some cases, material being tested for radiation damage should preferably have low levels of radioactivity.

It is expensive to arrange the irradiation of target materials by high energy neutrons sources, whereas it is relatively inexpensive to irradiate using an ion beam on the MC-40 Cyclotron. The energies can be controlled, and a set dose at a single energy, or a range of energies, can be precisely deposited in the target material.

The Activity code discussed here was developed to calculate the activity of a target material irradiated by a proton beam. It has been developed in Fortran and uses data from the TENDL-2013 proton cross section database, SRIM ion transport code and NDS radioactive decay database.

1.1.2 Simulating Ion Irradiation with SRIM

A package of ion transport codes, SRIM, is freely available to download and use to investigate the transport of ions through matter. SRIM uses the binary collision approximation (BCA) to simulate the passage of ions in a material. It is an approximate method, and one key restriction is that it does not take into account the structure of the material, and this approximation is therefore also imposed on the Activity code.

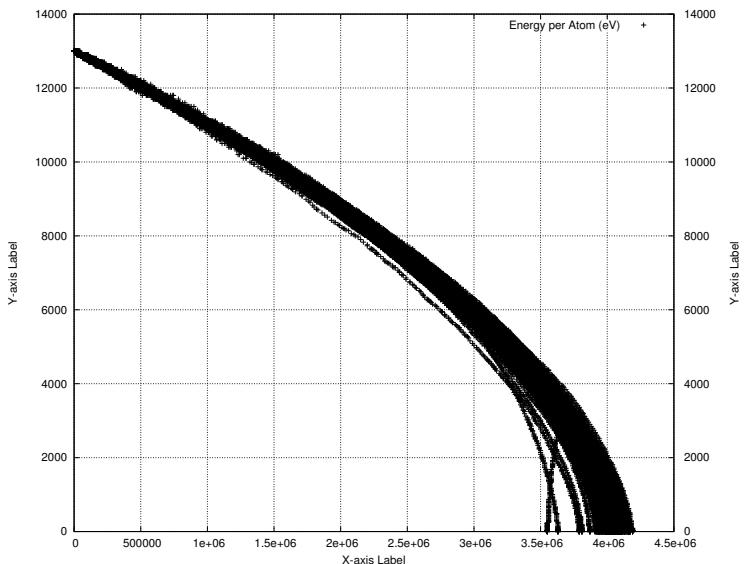


Figure 1.1: Proton energy loss in Fe simulated with SRIM [2]

One file that SRIM creates is of importance to the Activity code, and that is the trajectory file that contains the energy and x,y,z co-ordinate data points for simulated ions moving through matter. Figure 1.1 shows the trajectory of protons passing through an Iron target, and it is this set of data points (together with the cross section database) that the Activity code uses to calculate the reaction rates for the transmutation of nuclei in the target. At higher energies, the ions slow as they lose energy due to electronic stopping, but as the ion energy drops the mechanism of loss through nuclear collisions becomes important. The spreading of ion depths at lower energies is a result of the higher momentum transfer during nuclear collisions, as can be seen in Figure 1.1.

1.1.3 Transmutation of Nuclei by Ion Irradiation

Considering a simplified nuclear potential well, energetic protons approaching a nucleus may overcome the coulomb potential barrier. They are captured by the nucleus and held within the potential well by the strong nuclear force. This process may leave the nucleus in an excited and unstable state, depending on the input energy of the proton and configuration of nucleons. The process is probabilistic, and the average chance of a reaction (the microscopic cross section) may be measured as a function of the projectile, projectile energy and target, either experimentally or by optical model potential calculations. The reaction rate is calculated from the microscopic cross section using the following equation:

$$R = \frac{J}{e} n_t \sigma \cdot 10^{28} \delta t \quad (1.1)$$

- R Reaction Rate (reactions per second)
- J Beam current (A)
- n_t Number density of target (atoms per cubic metre)
- σ Microscopic reaction cross section (barns)
- e Elementary charge (1.602177E-19C)
- δT Target thickness (m)

1.2 Decay and Activity Equations

1.2.1 Radioactive Decay

Radioactive decay is the random change in nucleons or energy state of an unstable nucleus. It is impossible to predict when a nucleus will decay, but the decay of a collection of nuclei is statistical in nature. The radioactivity and number of unstable nuclei at time t can be predicted using the decay constant λ for the radioactive isotope. This constant is defined as follows:

$$\lambda = -\frac{N'(t)}{N(t)} \quad (1.2)$$

The number of radioactive nuclei $N(t)$ at time t is given by the following equation, where $N(0)$ is the starting number of nuclei:

$$N(t) = N(0) \exp(-t\lambda) \quad (1.3)$$

The activity $A(t)$ of the radioactive nuclei is predicted at time t by using the following equations, where $N'(t)$ is the change in amount of nuclei with respect to time:

$$A(t) = -N'(t) = \lambda N(t) \quad (1.4)$$

$$A(t) = \lambda N(0) \exp(-t\lambda) \quad (1.5)$$

1.2.2 Bateman Equation for Radioactive Decay

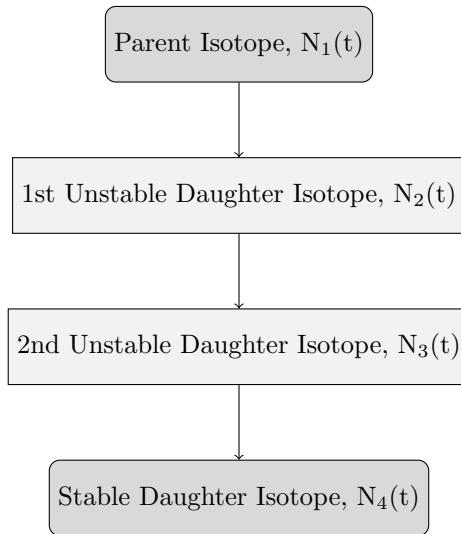


Figure 1.2: An example decay chain from an unstable parent isotope, through unstable daughter isotopes ending with a stable daughter isotope.

The English mathematician Harry Bateman derived an equation to calculate the amount of each isotope in a decay chain, illustrated in Figure 1.2, at time t .

$$N_n(t) = \sum_{i=1}^{i=n} \left(\left(\prod_{j=i}^{j=n-1} \lambda_{(ij+1)} \right) \sum_{j=i}^{j=n} \left(\frac{N_{i0} \exp(-\lambda_j t)}{\prod_{p=i, p \neq j}^{p=n} (\lambda_p - \lambda_j)} \right) \right) \quad (1.6)$$

When a radioactive isotope decays, there may be more than one mode of decay, and this leads to branching factors. Pb-214 only decays via beta decay to Bi-214, giving a branching factor of 1.0, whereas Bi-214 has a 99.979% chance of decaying to Po-214 by beta decay and a 0.021% of emitting an alpha particle and decaying to Tl-210 (branching factors of 0.99979 and 0.00021 respectively) [5].

When a target material is irradiated, there is a source term for transmuted nuclei due to the irradiation. The daughter isotopes of these transmuted isotopes will also be affected by the irradiation and will transmute further, giving a source term for each daughter isotope as a result of the irradiation. Sources for each isotope in the decay chain, and branching factors between a parent isotope and its daughter isotope/s must be accounted for.

The Bateman equation was derived using Laplace transforms, and this same method has been used to develop a modified equation that incorporates branching factors and production rates for each isotope in the decay chain, as illustrated by Figure 1.3.

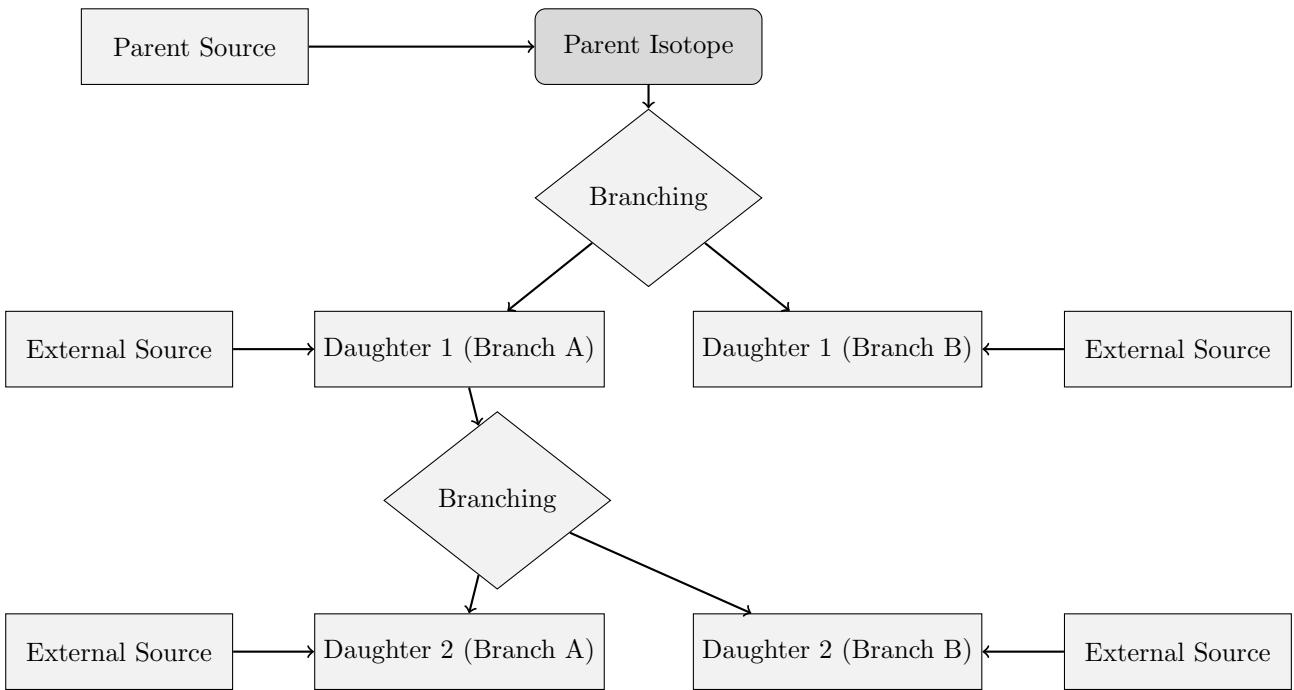


Figure 1.3: An example of several decay chains including branching factors and possible external source terms for each isotope on each chain.

1.2.3 Laplace Transform

Laplace Transforms are a useful mathematical tool, and allow ordinary differential equations to be solved by simple algebraic manipulation in the s domain. Bateman took advantage of Laplace Transforms in deriving his equation, and this is the method that has been taken here as well.

$$F(s) = \int_0^{\infty} f(t) \exp(-st) dt \quad (1.7)$$

1.2.4 Constructing the Differential Equations

The first step is to set up differential equations for the parent isotope, unstable daughter isotopes and stable daughter isotope. The parent isotope has a source term, due to production, and a loss term, due to decay. The unstable daughter isotopes have two source terms, from the production by irradiation induced transmutation

and the decay of preceding isotopes in the decay chain, and a loss term, due to decay. Finally, the stable daughter that finalizes the decay chain has two source terms (the same as the unstable daughters) but no loss term.

The variables (and vectors) used in these equations are defined as follows:

- $\vec{\lambda}$ vector containing isotope decay constants λ_i
- \vec{b} vector containing isotope to isotope branching factors b_i
- \vec{w} vector containing isotope production rates w_i
- t time at which activity/amount of isotope is measured
- $N_i(0)$ starting amount of the i th isotope
- $N_i(t)$ amount of the i th isotope at time t
- $N'_i(t)$ change in amount of the i th isotope, with respect to time, at time t

The differential equations for the parent isotope (first isotope), unstable daughter isotopes (i th isotopes) and stable, final, daughter isotope (z th isotope) in the time domain are as follows:

$$N'_1(t) = \omega_1 - \lambda_1 N_1(t) \quad (1.8)$$

$$N'_i(t) = \omega_i + b_{i-1} \lambda_{i-1} N_{i-1}(t) - \lambda_i N_i(t) \quad (1.9)$$

$$N'_z(t) = \omega_z + b_{z-1} \lambda_{z-1} N_{z-1}(t) \quad (1.10)$$

Applying the Laplace Transform to these three differential equations allows them to be manipulated and solved algebraically in the s -domain.

$$N_1(s) = \frac{1}{s + \lambda_1} N_1(0) + \frac{1}{s(s + \lambda_1)} \omega_1 \quad (1.11)$$

$$N_i(s) = \frac{1}{s(s + \lambda_i)} (\omega_i) + \frac{1}{s + \lambda_i} (b_{i-1} \lambda_{i-1} N_{i-1}(s)) + \frac{1}{s + \lambda_i} N_i(0) \quad (1.12)$$

$$N_z(s) = \frac{1}{s^2} \omega_z + \frac{1}{s} b_{z-1} \lambda_{z-1} N_{z-1}(s) + \frac{1}{s} N_z(0) \quad (1.13)$$

1.2.5 Numerical Inversion of the Laplace Transform

The Gaver-Stehfest[6] algorithm was developed in the 1960s and 1970s and is a method of calculating the inverse of a Laplace Transform in the real number domain. It is an easy to implement and reasonably accurate method, although it is an approximation to the real value. A comparison between an analytic and numeric inversion for the unstable isotope Po-218 is discussed at the end of this section.

$$f(t) \approx f_n(t) = \frac{\ln(2)}{t} \sum_{k=1}^{2n} a_k(n) F(s) \text{ where } n \geq 1, t > 0 \quad (1.14)$$

$$s = \frac{k \ln(2)}{t} \quad (1.15)$$

$$a_k(n) = \frac{(-1)^{(n+k)}}{n!} \sum_{j=\text{Floor}(\frac{k+1}{2})} j^{n+1} \binom{n}{j} \binom{2j}{j} \binom{j}{k-j} \quad (1.16)$$

The equation for the i th isotope may be calculated by recursively calculating the equations by numeric inversion, starting from the first (parent isotope) and inserting the result into each subsequent recursion until the i th isotope is reached (changing the equations appropriately for the parent, unstable daughter and stable daughter isotopes).

1.2.6 Analytic Solution by Partial Fraction Expansion

The equation for the i th isotope in the s domain can be written in full by substituting the preceding equation until the parent isotope is reached, and this full eqaution may be rearranged with the production amount of each isotope and starting amount of each isotope in individual terms. Each of these terms is multiplied by a fraction that can be expanded, using partial fractions, and inverted analytically.

This is illustrated with an example unstable isotope, fourth in the decay chain (including the parent isotope):

$$\begin{aligned} N_4(s) = & \frac{1}{(s + \lambda_1)(s + \lambda_2)(s + \lambda_3)(s + \lambda_4)} b_2 b_3 b_4 \lambda_1 \lambda_2 \lambda_3 N_1(0) \\ & + \frac{1}{(s + \lambda_2)(s + \lambda_3)(s + \lambda_4)} b_3 b_4 \lambda_2 \lambda_3 N_2(0) \\ & + \frac{1}{(s + \lambda_3)(s + \lambda_4)} b_4 \lambda_3 N_3(0) \\ & + \frac{1}{(s + \lambda_4)} N_4(0) \\ & + \frac{1}{s(s + \lambda_1)(s + \lambda_2)(s + \lambda_3)(s + \lambda_4)} b_2 b_3 b_4 \lambda_1 \lambda_2 \lambda_3 \omega_1 \\ & + \frac{1}{s(s + \lambda_2)(s + \lambda_3)(s + \lambda_4)} b_3 b_4 \lambda_2 \lambda_3 \omega_2 \\ & + \frac{1}{s(s + \lambda_3)(s + \lambda_4)} b_4 \lambda_3 \omega_3 \\ & + \frac{1}{s(s + \lambda_4)} \omega_4 \end{aligned} \quad (1.17)$$

An example stable isotope, fourth (last) in the decay chain (including the parent isotope):

$$\begin{aligned}
N_4(s) = & \frac{1}{s(s+\lambda_1)(s+\lambda_2)(s+\lambda_3)} b_2 b_3 b_4 \lambda_1 \lambda_2 \lambda_3 N_1(0) \\
& + \frac{1}{s(s+\lambda_2)(s+\lambda_3)} b_3 b_4 \lambda_2 \lambda_3 N_2(0) \\
& + \frac{1}{s(s+\lambda_3)} b_4 \lambda_3 N_3(0) \\
& + N_4(0) \\
& + \frac{1}{s^2(s+\lambda_1)(s+\lambda_2)(s+\lambda_3)} b_2 b_3 b_4 \lambda_1 \lambda_2 \lambda_3 \omega_1 \\
& + \frac{1}{s^2(s+\lambda_2)(s+\lambda_3)} b_3 b_4 \lambda_2 \lambda_3 \omega_2 \\
& + \frac{1}{s^2(s+\lambda_3)} b_4 \lambda_3 \omega_3 \\
& + \frac{1}{s^2} \omega_4
\end{aligned} \tag{1.18}$$

By using partial fraction expansion and standard Laplace Transforms, the set of equations below is used to calculate the amount of the m th isotope in the decay chain, providing the m th isotope is unstable.

$$N_m(t; \vec{\lambda}, \vec{b}, \vec{w}) = \sum_{k=1,m} r(k; \vec{\lambda}, \vec{b}) [f(t; k, m, \vec{\lambda}) N_k(0) + g(t; k, m, \vec{\lambda}) w_k] \tag{1.19}$$

$$r(k, m, \vec{\lambda}) = \begin{cases} \prod_{i=k, m-1} (b_{i+1} \lambda_i), & \text{if } k < m \\ 1, & \text{if } k = m \end{cases} \tag{1.20}$$

$$f(t; k, m, \vec{\lambda}) = (-1)^{m-k} \sum_{i=k, m} \left[\exp(-\lambda_i t) \prod_{j=k, m; j \neq i} \left(\frac{1}{\lambda_i - \lambda_j} \right) \right] \tag{1.21}$$

$$g(t; k, m, \vec{\lambda}) = \frac{1}{\prod_{i=k, m} \lambda_i} + (-1)^{m-k+1} \sum_{i=k, m} \left[\frac{1}{\lambda_i} \exp(-\lambda_i t) \prod_{j=k, m; j \neq i} \left(\frac{1}{\lambda_i - \lambda_j} \right) \right] \tag{1.22}$$

The set of equations below is used to calculate the amount of the m th isotope in the decay chain, where the m th isotope is stable.

$$N_m(t; \vec{\lambda}, \vec{b}, \vec{w}) = N_m + w_m t + \sum_{k=1, m-1} r(k; \vec{\lambda}, \vec{b}) [f(t; k, m-1, \vec{\lambda}) N_k(0) + g(t; k, m, \vec{\lambda}) w_k] \tag{1.23}$$

$$r(k, m, \vec{\lambda}) = \begin{cases} \prod_{i=k, m-1} (b_{i+1} \lambda_i), & \text{if } k < m \\ 1, & \text{if } k = m \end{cases} \tag{1.24}$$

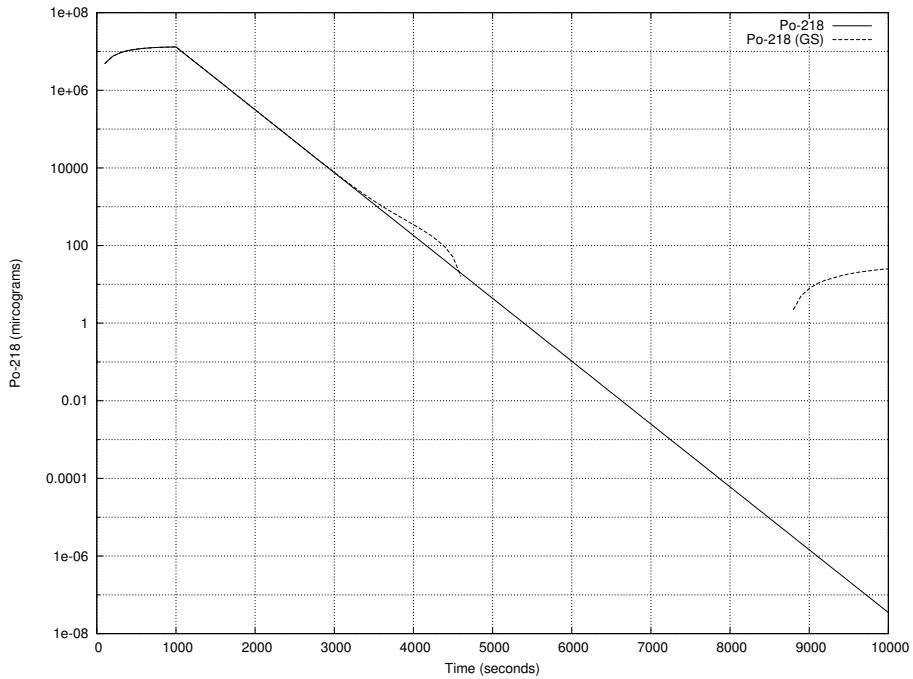


Figure 1.4: Decay of Po-218: Analytic and Gaver-Stehfest Calculations [5]

$$f(t; k, m, \vec{\lambda}) = \frac{1}{\prod_{i=k,m} \lambda_i} + (-1)^{m-k+1} \sum_{i=k,m} \left[\frac{1}{\lambda_i} \exp(-\lambda_i t) \prod_{j=k,m; j \neq i} \left(\frac{1}{\lambda_i - \lambda_j} \right) \right] \quad (1.25)$$

$$g(t; k, m, \vec{\lambda}) = \frac{1}{\prod_{i=k,m} \lambda_i} t + \frac{\sum_{i=k,m} \left[\prod_{j=k,m; j \neq i} \lambda_j \right]}{\prod_{i=k,m} \lambda_i^2} + (-1)^{m-k+1} \sum_{i=k,m} \left[\frac{1}{\lambda_i^2} \exp(-\lambda_i t) \prod_{j=k,m; j \neq i} \left(\frac{1}{\lambda_i - \lambda_j} \right) \right] \quad (1.26)$$

1.2.7 Preference: Analytic over Numeric

The numeric solution only requires the equation to be solved in the s-domain; the Gaver-Stehfest algorithm performs the inversion. It is worth the extra effort to derive and implement an analytic solution, as the numeric is only an approximation. Examples of the pitfalls of the numeric solution are that it can give negative amounts of an isotope and the difference between the numeric and analytic calculated amounts can become quite large when the isotope decays away to a very small value. Figure 1.4 shows the predicted decay of a sample of Po-218 irradiated for 1,000s, and sampled until 10,000s. In the region between 4,000s and 9,000s the amount from the numeric calculation drops below zero, whereas the analytic calculation remains above zero, as would be expected.

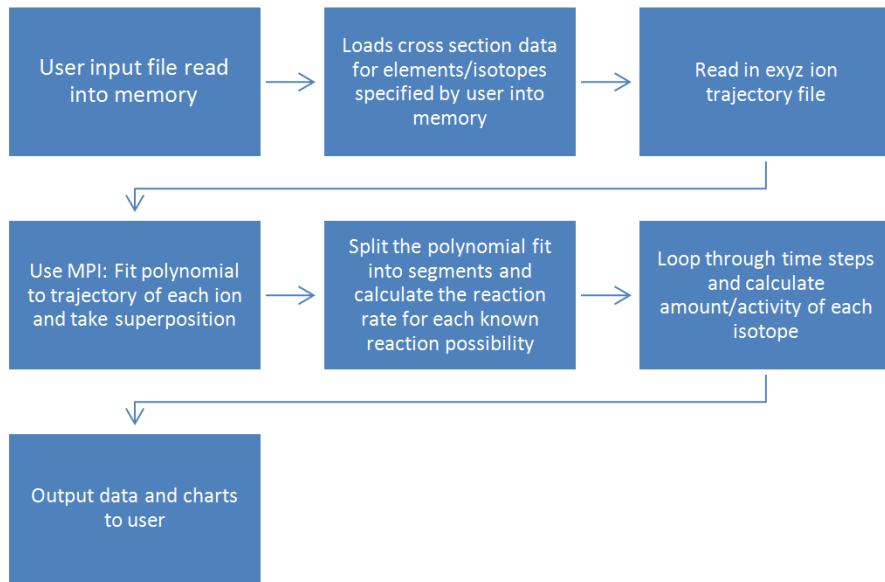


Figure 1.5: Flow chart of major processes in the Activity code

1.3 Computational Methods

1.3.1 Activity Code

The Activity program has been developed in Fortran and takes advantage of MPI (Message Parsing Interface) to speed up calculation times by allowing the use of multiple processes in parallel. It has a self contained maths library, although this could be improved in the future by using optimised maths libraries for certain functions (e.g. matrix operations).

The code was developed on a Debian version of Linux, but it should be supported on other variants of Linux and Unix, and does not require any specialist hardware.

The user is required to prepare an input file that contains the instructions required to perform a calculation. In addition to the input file, the user must provide an EXYZ ion trajectory file output by SRIM. Activity will read in the user input file, and the SRIM and data files listed within, before performing the calculation. Figure 1.5 shows a flowchart of the major steps the code performs.

There are various settings in the user input file, but the main ones relating to the simulated experiment are:

- element composition of target (percentage by mass)
- beam flux (current), energy, duration and area on target
- activity measurement time (end of the “experiment”)
- material density
- target thickness

Several data files are generated by Activity and, if the user has matplotlib [7], charts will be created too. The most relevant to the user are:

- gammaLines.dat - tally into discrete bins of predicted gamma counts

- ionTraj.dat - the averaged ion trajectory used in the calculation
- isotopeActivityFileG.dat - a large data file detailing the activity of every predicted radioactive isotope in the target at user specified times following irradiation

The charts include:

- activityTop5.png - activity of the top 5 active isotopes as a function of time after irradiation starts
- gammaLines.png - predicted gamma spectrum expected at the “experiment end time”

The Activity code uses the equations derived above to calculate the amount and activity of each isotope in the calculation. One problem with the original Bateman equation that also exists in the set of modified Bateman equations is that two different isotopes with the same decay constant will cause a singularity and a halt in the calculation. The activity code loops through all the decay constants in use before it attempts to run the calculation. If any isotope decay constants match they are varied by a small amount relative to the decay constant. It repeats this process until all decay constants are unique before proceeding.

1.3.2 Approximations

The accuracy of the Activity code is dependant on the input files provided by the user and the method used to calculate the reaction rates and resulting activity. The TENDL proton database consists of experimental measured cross sections as well as values calculated using the optical model potential. Using the latest database is recommended.

SRIM uses the binary collision approximation to simulate ion transport. It is a well tested code that has been used for many years. One limitation is that the structure of the material is not taken into account. This would have an impact on a user of the Activity program if they were trying to calculate, for example, whether a FCC (face centered cubic) steel would be irradiated differently when compared to a BCC (body centered cubic) steel. The Activity code would determine the activity of the steel as a function of the ion current, ion type and the density, thickness and composition of the steel, not its structure.

This version of the Activity code averages the path of all the SRIM simulated ions, rather than treating each ion differently. This may or may not have an impact on the results. If a new version of the code is developed there would be an option to calculate reaction rates for each individual simulated ion, and a comparison could then be made to the calculations using the averaged path of a set of ions.

The final approximation would be to use the numeric solution to the activity equations, although the analytic solution is forced within the code unless it returns a failed result.

1.3.3 Results

A target of high purity Iron was irradiated with 36 MeV protons by the University of Birmingham Scanditronix MC-40 Cyclotron. The target was 0.5mm thick and was irradiated at a current of 0.5 micro Amps for 300 seconds, irradiating approximately 0.25g of Iron. A high purity Germanium detector was used to measure the gamma peaks three days after irradiation.

The peak that dominated the readings was the 931 keV Cobalt 55 line. After calibrating the detector and adjusting the readings, this peak was measured at 44,300Bq +/- 2,000Bq. The activity of this peak as predicted by the Activity code was 44,565Bq.

A target of high purity Molybdenum was irradiated with 13 MeV protons by the University of Birmingham Scanditronix MC-40 Cyclotron. The target was 0.5mm thick and was irradiated at a current of 5 micro Amps

Table 1.1: Gamma peaks predicted and measured for a 13 MeV ion irradiated sample of Mo

Gamma Energy (keV)	Predicted Activity (Bq)	Experimental Activity (Bq)
766	4.45E6	5.11E5 +/- 2.5E4
778	6.14E6	1.36E6 +/- 6.8E4
812	5.04E6	1.15E6 +/- 5.8E4
850	6.00E6	1.39E6 +/- 7.0E4
126	9.33E5	2.10E5 +/- 1.1E4

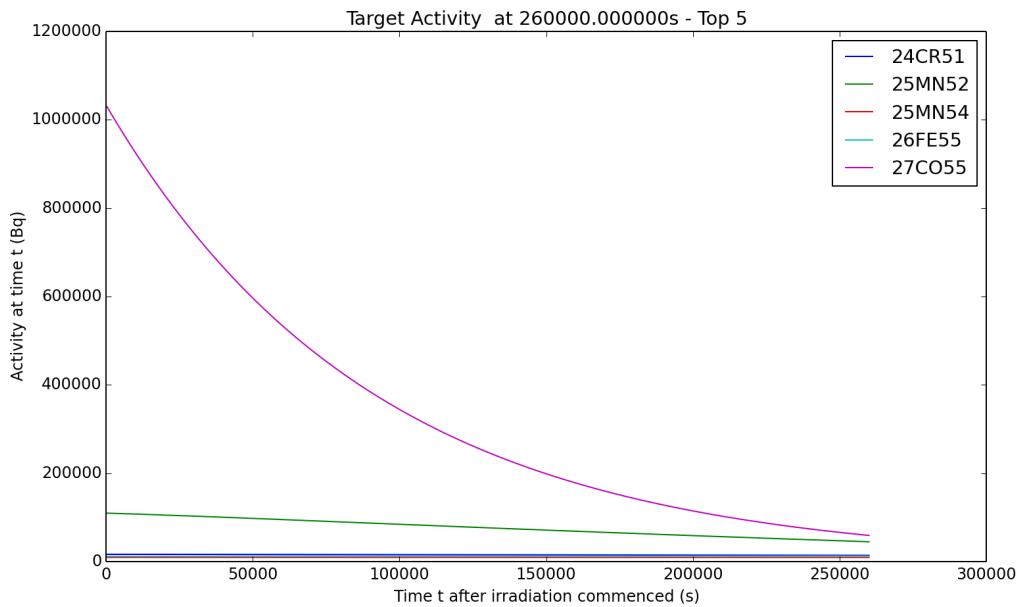


Figure 1.6: Sample Activity code output chart for the top five most active isotopes for Iron irradiated by 36MeV protons.

for 1500 seconds, irradiating approximately 0.3g of Molybdenum. A high purity Germanium detector was used to measure the gamma peaks three days after irradiation.

Five peaks were of interest, and these are listed in Table 1.1.

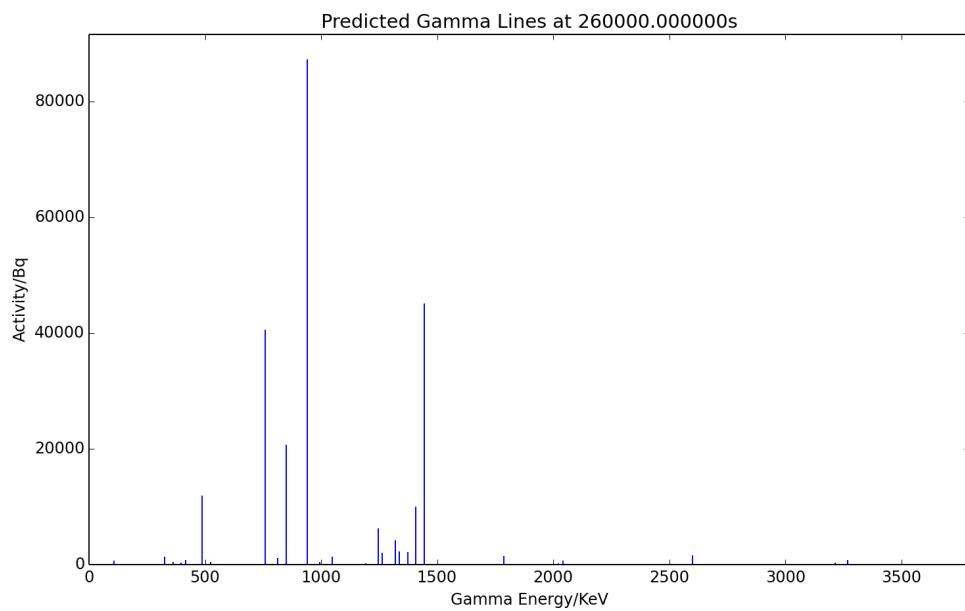


Figure 1.7: Sample Activity code output chart for the expected gamma lines to be measured for Iron irradiated by 36MeV protons.

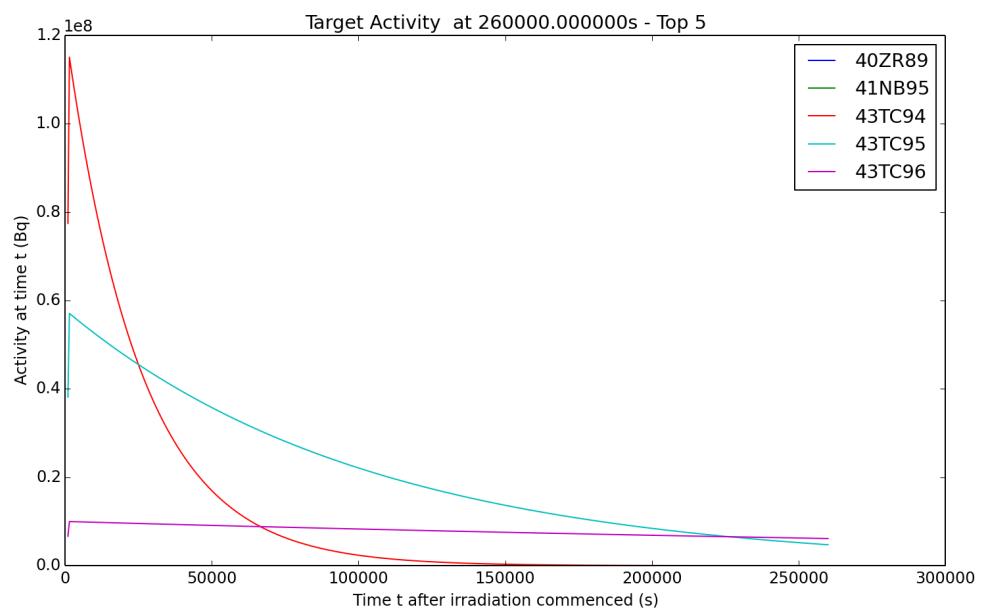


Figure 1.8: Sample Activity code output chart for the top five most active isotopes for Molybdenum irradiated by 13MeV protons.

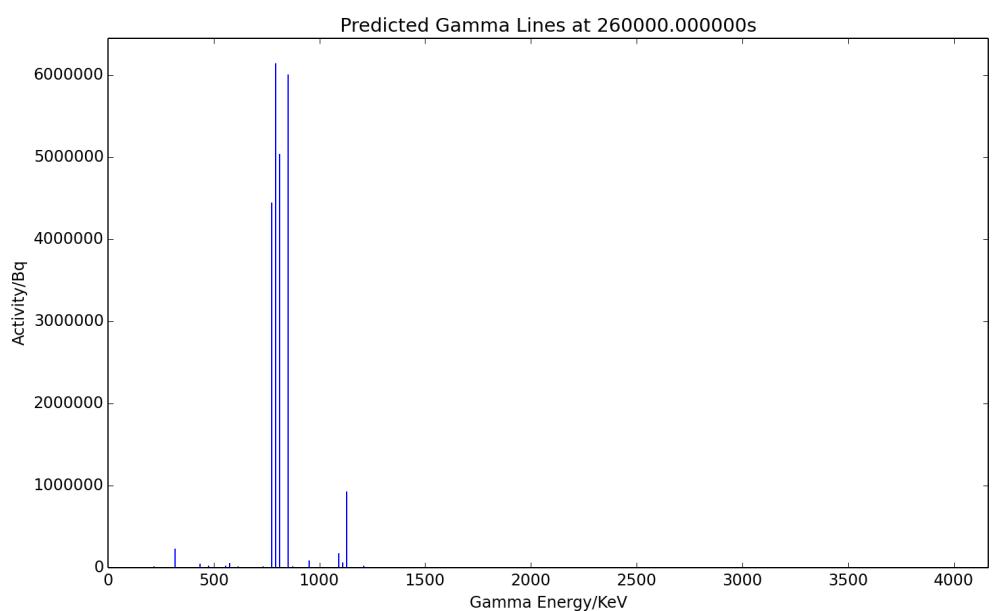


Figure 1.9: Sample Activity code output chart for the expected gamma lines to be measured for Molybdenum irradiated by 13MeV protons.

Chapter 2

Installation and Using the Activity Code

2.1 Getting Started

2.1.1 Prerequisites

This code was designed to run on Linux and has been developed and tested on the Debian based distribution Mint. It has not been tested on any other versions of Linux or Unix, nor has it been tested on cygwin.

The minimal requirements of the code are:

- 250MB per process + 200MB overhead
- 1GB Hard Drive space
- Ideally a modern multicore processor
- Fortran90
- OpenMPI

Optional:

- GNUPlot
- Python & MatPlotLib

2.2 Installing Activity

2.2.1 Download Source Code

The most recent source code is available for download from github. It may be downloaded and extracted using the terminal and the commands in listing 1.

Listing 2.1: bash

```
1 cd ~
2 mkdir -p ~/activity/tar
3 cd ~/activity/tar
4 wget https://github.com/BenPalmer1983/activity/raw/master/activity.tar.gz
5 tar -xzvf activity.tar.gz
```

2.2.2 Compile Source Code

Once the source files have been downloaded and extracted, run the install script then remove the downloaded files (if you need to free up space). The terminal commands are given in listing 2. Providing the prerequisites are available, the process will only take a few minutes (with a reasonable Internet connection).

Listing 2.2: bash

```
1 cd ~/activity/tar
2 ./install.sh #choose installation directory when prompted; default is ~/activity
3 rm -R ~/activity/tar
```

The activity code is now installed and the activity.x binary may be executed on the terminal. If the terminal fails to find the binary, the .bashrc or .bash_profile file must be edited, or a symlink created from the bin file to a valid and accessible bin directory.

2.3 Input File

The input file sets the target starting composition, the beam and simulated experiment settings as well as the paths to the required data files.

#elements

The elements keyword instructs the code to read in the composition of the target material. The elements and their percentage by weight are listed under the keyword.

Listing 2.3: Input File

```
1 #elements
2 Fe 72
3 Cr 18
4 Ni 8
5 Mg 2
```

#isotopes #decaymodes #gammaenergies #xsfiles

Four sets of data files are distributed with the code, and the paths to these files are defined underneath each keyword. N.B. the path for the xsfiles is to the directory that holds all the cross section data files, rather than each individual file.

Listing 2.4: Input File

```
1 #isotopes
2 "/home/ben/activity/data/isotopes.txt"
3 #decaymodes
4 "/home/ben/activity/data/decaymodes.txt"
```

```
5 #gammaenergies
6 "/home/ben/activity/data/gammaenergies.txt"
7 #xsfiles
8 "/home/ben/activity/data/xs"
```

```
#trajfile
```

The SRIM exyz file used for the calculation is pointed to under this keyword.

Listing 2.5: Input File

```
1 #trajfile
2 "/home/ben/activity/examples/Fe36MeV.exyz"
```

```
#polyfitorder #integrationgranularity
```

The code calculates the reaction rate of each projectile and target nucleus by first fitting a polynomial to the energy/depth data file. This polynomial gives an average path for the projectiles travelling through the target, allowing E(x) to be calculated quickly. The order of the polynomial is selected here, and a 5th order shold give a reasonable fit. The integration granularity keyword determines how many sections the polynomial is split up into. The energy, cross section for each section and the beam settings are used to calculate the reaction rate for each section, and these are all summed up to give the overall reaction rate.

Listing 2.6: Input File

```
1 #polyfitorder
2 5
3 #integrationgranularity
4 10
```

```
#beamflux #beamenergy #beamduration #beamarea
```

These keywords are self explanatory and control the beam settings used in the calculation.

Listing 2.7: Input File

```
1 #beamflux
2 0.5 uA
3 #beamenergy
4 36 MeV
5 #beamduration
6 300 s
7 #beamarea
8 100 mm2
```

```
#amtime #timestep
```

The time at which the activity of the sample is measured is set by the #amtime keyword. The Activity code calculates the activity of each radioactive isotope in the calculation from the time the beam starts until the activity measured time; and the intervals at which these calculations are made is determined by the #timestep keyword.

Listing 2.8: Input File

```
1 #amtime
2 260000 s
```

```
3 #timestep
4 1000 s
```

#projectile

The current version only supports protons as the projectile, therefore the only possible atomic and mass number combination for a projectile is 1 1.

Listing 2.9: Input File

```
1 #projectile
2 1 1
```

#targetthickness #materialdensity

Both the target thickness and density of the material it is made from are input with these keywords.

Listing 2.10: Input File

```
1 #targetthickness
2 0.5 mm
3 #materialdensity
4 8000 kgm3
```

#vpi

This keyword is only being used for a feature currently under testing, so the keyword can be omitted from the input file.

Listing 2.11: Input File

```
1 #vpi
2 60.2
```

#individualisotopeactivity #verboseterminal

If the #individualisotopeactivity keyword is followed by yes; additional data files are output containing the activity of each individual isotope in the calculation. #verboseterminal followed by yes will output more verbosely to the terminal.

Listing 2.12: Input File

```
1 #individualisotopeactivity
2 yes
3 #verboseterminal
4 yes
```

#targetdpa

This keyword is only being used for a feature currently under testing, so the keyword can be omitted from the input file.

Listing 2.13: Input File

```
1 #targetdpa
2 0.0
```

```
#gammachartresolution
```

The gamma chart is created by tallying the gamma values into bins, and this figure specifies the resolution(number of bins) used to create the output chart.

Listing 2.14: Input File

```
1 #gammachartresolution  
2 200
```

2.4 Acknowledgements

We would like to thank and acknowledge the input and advice of the following:

- Dr Chris Cooper and John Hewett for irradiation activation data points.
- The University of Birmingham for providing the funding for this project.

Appendices

Appendix A

Example Input File

A.1 Iron 36MeV Proton Beam

Listing A.1: Fe36MeV.in

```
1 #elements
2 Fe 100
3 #isotopes
4 "/home/ben/activity/data/isotopes.txt"
5 #decaymodes
6 "/home/ben/activity/data/decaymodes.txt"
7 #gammaenergies
8 "/home/ben/activity/data/gammaenergies.txt"
9 #xsfiles
10 "/home/ben/activity/data/xs"
11 #trajfile
12 "/home/ben/activity/examples/Fe36MeV.exyz"
13 #polyfitorder
14 5
15 #integrationgranularity
16 10
17 #beamflux
18 0.5 uA
19 #beamenergy
20 36 MeV
21 #beanduration
22 300 s
23 #beamarea
24 100 mm2
25 #amtime
26 260000 s
27 #timestep
28 1000 s
29 #projectile
30 1 1
31 #targetthickness
32 0.5 mm
33 #materialdensity
34 8000 kgm3
```

```
35 #vpi
36 60.2
37 #individualisotopeactivity
38 yes
39 #verboseterminal
40 yes
41 #targetdpa
42 0.0
43 #gammachartresolution
44 200
```

Appendix B

Fortran 90 Code

B.1 Fortran 90 Implementation of Analytic Method

Listing B.1: Fortran 90

```
1 Type :: decayChainObj
2   Real(kind=DoubleReal) :: time = 0.0D0
3   !Real(kind=DoubleReal) :: productionRate = 0.0D0
4   Integer(kind=StandardInteger) :: isotopes
5   Character(len=16), Dimension(1:100) :: label
6   Real(kind=DoubleReal), Dimension(1:100) :: productionRate = 0.0D0
7   Real(kind=DoubleReal), Dimension(1:100) :: branchFactor = 1.0D0 ! from isotope parent
8   Real(kind=DoubleReal), Dimension(1:100) :: decayConstant = -1.0D0 ! negative for stable
9   Real(kind=DoubleReal), Dimension(1:100) :: halfLife = -1.0D0 ! negative for stable
10  Real(kind=DoubleReal), Dimension(1:100) :: amountStart = 0.0D0
11  Real(kind=DoubleReal), Dimension(1:100) :: amountEnd = 0.0D0
12 End Type
13
14 Subroutine CalcIsotopeChain(decayChain)
15 ! Uses inverse laplace transform to calculate isotope amounts at time t (after time = 0)
16 ! t time in seconds after t=0
17 ! w production rate of parent isotope
18 ! isotope chain data
19   Implicit None ! Force declaration of all variables
20 ! Vars In/Out
21   Type(decayChainObj) :: decayChain
22 ! Vars Private
23   Integer(kind=StandardInteger) :: i
24   Real(kind=DoubleReal) :: t, nEnd
25   Real(kind=DoubleReal), Dimension(1:100) :: W ! Production Rate
26   Real(kind=DoubleReal), Dimension(1:100) :: L ! Lambda
27   Real(kind=DoubleReal), Dimension(1:100) :: N ! Starting number of atoms
28   Real(kind=DoubleReal), Dimension(1:100) :: B ! Exp
29 ! Complete decay chain data
30   Call decayChainComplete(decayChain)
31 ! store input in shortned name arrays to make equations clearer
32   t = decayChain%time
33   Do i=1,decayChain%isotopes
34     W(i) = decayChain%productionRate(i)
```

```

35     L(i) = decayChain%decayConstant(i)
36     B(i) = decayChain%branchFactor(i)
37     N(i) = decayChain%amountStart(i)
38   End Do
39 ! Break infinities
40   Call DecayBreakInfinities(L,decayChain%isotopes)
41 ! Run analytic calculations
42 ! Loop through isotopes
43 ! Using L-1(1/(q+ps) = (1/p)*exp(-1*(q/p)*t) and partial fractions
44   Do i=1,decayChain%isotopes
45     nEnd = CalcIsotopeChainCalc(t,i,W,L,N,B)
46     decayChain%amountEnd(i) = nEnd
47   End Do
48 End Subroutine CalcIsotopeChain
49
50 Subroutine decayChainComplete(decayChain)
51 ! Completes the decay chain object
52 ! Implicit None ! Force declaration of all variables
53 ! Vars In/Out
54 Type(decayChainObj) :: decayChain
55 ! Vars Private
56 Integer(kind=StandardInteger) :: i, n
57 n = 0
58 Do i=1,100
59   n = n + 1
60   If(decayChain%decayConstant(i).eq.-1.0D0.and.decayChain%halfLife(i).gt.0.0D0)Then
61 ! complete decay constant from half life
62     decayChain%decayConstant(i) = lnTwo/decayChain%halfLife(i)
63   End If
64   If(decayChain%halfLife(i).le.0.0D0.and.decayChain%decayConstant(i).gt.0.0D0)Then
65 ! complete decay constant from half life
66     decayChain%halfLife(i) = lnTwo/decayChain%decayConstant(i)
67   End If
68 ! Adjust for stable isotope
69   If(decayChain%decayConstant(i).lt.0.0D0)Then
70     decayChain%halfLife(i) = -1.0D0
71     decayChain%decayConstant(i) = 0.0D0
72   End If
73   If(decayChain%halfLife(i).lt.0.0D0)Then
74     decayChain%halfLife(i) = -1.0D0
75     decayChain%decayConstant(i) = 0.0D0
76   End If
77 ! Break out if stable
78   If(decayChain%decayConstant(i).eq.0.0D0)Then
79     Exit
80   End If
81   End Do
82   decayChain%isotopes = n
83 End Subroutine decayChainComplete
84
85 Subroutine DecayBreakInfinities(L,n)
86 !
87 ! Implicit None ! Force declaration of all variables

```

```

88 ! Vars In/Out
89   Real(kind=DoubleReal), Dimension(:) :: L ! Lambda
90   Integer(kind=StandardInteger) :: n
91 ! Vars Private
92   Integer(kind=StandardInteger) :: i,j
93   Logical :: breaking
94 ! Loop and alter matching decay constants slightly
95   breaking = .true.
96   Do While(breaking)
97     breaking = .false.
98     Do i=1,n-1
99       Do j=i+1,n
100         If(L(i).eq.L(j))Then
101           breaking = .true.
102           L(i) = L(i)*1.0000001D0 ! Vary by 0.00001%
103         End If
104       End Do
105     End Do
106   End Do
107 End Subroutine DecayBreakInfinities
108
109 Function CalcIsotopeChainCalc(t,m,W,L,N,B) Result (nEnd)
110   Implicit None ! Force declaration of all variables
111 ! Vars In
112   Real(kind=DoubleReal) :: t
113   Integer(kind=StandardInteger) :: m
114   Real(kind=DoubleReal), Dimension(1:100) :: W ! Production Rate
115   Real(kind=DoubleReal), Dimension(1:100) :: L ! Lambda
116   Real(kind=DoubleReal), Dimension(1:100) :: N ! Starting number of atoms
117   Real(kind=DoubleReal), Dimension(1:100) :: B ! Branch factor
118 ! Vars Out
119   Real(kind=DoubleReal) :: nEnd
120 ! Vars Private
121   Integer(kind=StandardInteger) :: i, j, k, z
122   Real(kind=DoubleReal) :: mult, multR
123   Real(kind=DoubleReal) :: nChange
124 ! Init
125   nEnd = 0.0D0
126 ! -----
127 ! UNSTABLE Isotopes
128 ! -----
129   If(L(m).gt.0.0D0)Then
130 ! Loop through terms
131   Do k=1,m
132     multR = CalcIsotopeChainMultR(k,m,L,B)
133     mult = multR * N(k)
134   ! decay of starting matter
135     nChange = CalcIsotopeChainF_Unstable(k,m,t,mult,L)
136     nEnd = nEnd + nChange
137   ! production term
138     mult = multR * W(k)
139     nChange = CalcIsotopeChainG_Unstable(k,m,t,mult,L)
140     nEnd = nEnd + nChange

```

```

141      print *,k,nEnd
142      End Do
143      Else
144 ! -----
145 ! STABLE Isotopes
146 ! -----
147 ! Loop through terms
148     nEnd = N(m)+t*W(m)
149     Do k=1,m-1
150         multR = CalcIsotopeChainMultR(k,m,L,B)
151         mult = multR * N(k)
152 ! decay of starting matter
153         nChange = CalcIsotopeChainF_Stable(k,m,t,mult,L)
154         nEnd = nEnd + nChange
155 ! production term
156         mult = multR * W(k)
157         nChange = CalcIsotopeChainG_Stable(k,m,t,mult,L)
158         nEnd = nEnd + nChange
159         print *,k,nEnd
160     End Do
161     End If
162 End Function CalcIsotopeChainCalc
163
164 Function CalcIsotopeChainMultR(k,m,L,B) Result (multR)
165 ! Vars In
166     Integer(kind=StandardInteger) :: k, m
167     Real(kind=DoubleReal), Dimension(1:100) :: L ! Lambda
168     Real(kind=DoubleReal), Dimension(1:100) :: B ! Branching Factor
169 ! Vars Out
170     Real(kind=DoubleReal) :: multR
171 ! Private
172     Integer(kind=StandardInteger) :: i
173 ! Result
174     multR = 1.0D0
175     If(k.lt.m)Then
176         Do i=k,m-1
177             multR = multR * B(i+1) * L(i)
178         End Do
179     End If
180 End Function CalcIsotopeChainMultR
181
182 ! -----
183 ! Unstable Isotope Functions
184 ! -----
185
186 Function CalcIsotopeChainF_Unstable(k,m,t,mult,L) Result (nChange)
187 ! Vars In
188     Integer(kind=StandardInteger) :: k, m
189     Real(kind=DoubleReal) :: t, mult
190     Real(kind=DoubleReal), Dimension(1:100) :: L ! Lambda
191 ! Vars Out
192     Real(kind=DoubleReal) :: nChange
193 ! Private

```

```

194     Real(kind=DoubleReal) :: multP
195     Real(kind=DoubleReal) :: p, q, r, s
196     Integer(kind=StandardInteger) :: i, j
197 ! Calculate isotope amount change
198     nChange = 0.0D0
199     multP = (-1.0D0)**(m-k)
200 ! Loop through pfrac
201     Do i=k,m
202         r = 1.0D0
203         Do j=k,m
204             If(j.ne.i)Then
205                 r = r * (L(i)-L(j))
206             End If
207         End Do
208         nChange = nChange + (1.0D0/r)*exp(-1.0D0*L(i)*t)*multP*mult
209     End Do
210 End Function CalcIsotopeChainF_Unstable
211
212 Function CalcIsotopeChainG_Unstable(k,m,t,mult,L) Result (nChange)
213 ! Vars In
214     Integer(kind=StandardInteger) :: k, m
215     Real(kind=DoubleReal) :: t, mult
216     Real(kind=DoubleReal), Dimension(1:100) :: L ! Lambda
217 ! Vars Out
218     Real(kind=DoubleReal) :: nChange
219 ! Private
220     Real(kind=DoubleReal) :: multP
221     Real(kind=DoubleReal) :: p, q, r, s
222     Integer(kind=StandardInteger) :: i, j
223 ! Calculate isotope amount change
224     nChange = 0.0D0
225     multP = (-1.0D0)**(m-k+1)
226 ! term A
227     r = 1.0D0
228     Do i=k,m
229         r = r * L(i)
230     End Do
231     nChange = nChange + (1.0D0/r)*mult
232 ! Loop through pfrac
233     Do i=k,m
234         r = 1.0D0*L(i)
235         Do j=k,m
236             If(j.ne.i)Then
237                 r = r * (L(i)-L(j))
238             End If
239         End Do
240         nChange = nChange + (1.0D0/r)*exp(-1.0D0*L(i)*t)*multP*mult
241     End Do
242 End Function CalcIsotopeChainG_Unstable
243
244 ! -----
245 ! Stable Isotope Functions
246 ! -----

```

```

247
248 Function CalcIsotopeChainF_Stable(k,mIn,t,mult,L) Result (nChange)
249 ! Vars In
250   Integer(kind=StandardInteger) :: k, mIn
251   Real(kind=DoubleReal) :: t, mult
252   Real(kind=DoubleReal), Dimension(1:100) :: L ! Lambda
253 ! Vars Out
254   Real(kind=DoubleReal) :: nChange
255 ! Private
256   Integer(kind=StandardInteger) :: m
257   Real(kind=DoubleReal) :: multP
258   Real(kind=DoubleReal) :: p, q, r, s
259   Integer(kind=StandardInteger) :: i, j
260 ! In
261   m = mIn-1
262 ! Calculate isotope amount change
263   nChange = 0.0D0
264   multP = (-1.0D0)**(m-k+1)
265 ! term A
266   r = 1.0D0
267   Do i=k,m
268     r = r * L(i)
269   End Do
270   nChange = nChange + (1.0D0/r)*mult
271 ! Loop through pfrac
272   Do i=k,m
273     r = 1.0D0*L(i)
274     Do j=k,m
275       If(j.ne.i)Then
276         r = r * (L(i)-L(j))
277       End If
278     End Do
279     nChange = nChange + (1.0D0/r)*exp(-1.0D0*L(i)*t)*multP*mult
280   End Do
281 End Function CalcIsotopeChainF_Stable
282
283
284 Function CalcIsotopeChainG_Stable(k,mIn,t,mult,L) Result (nChange)
285 ! Vars In
286   Integer(kind=StandardInteger) :: k, mIn
287   Real(kind=DoubleReal) :: t, mult
288   Real(kind=DoubleReal), Dimension(1:100) :: L ! Lambda
289 ! Vars Out
290   Real(kind=DoubleReal) :: nChange
291 ! Private
292   Integer(kind=StandardInteger) :: m
293   Real(kind=DoubleReal) :: multP
294   Real(kind=DoubleReal) :: p, q, r, s
295   Integer(kind=StandardInteger) :: i, j
296 ! In
297   m = mIn-1
298 ! Calculate isotope amount change
299   nChange = 0.0D0

```

```

300     multP = (-1.0D0)**(m-k)
301 ! term A
302     r = 1.0D0
303     Do i=k,m
304         r = r * L(i)
305     End Do
306     nChange = nChange + (1.0D0/r)*t*mult
307 ! term B
308     p = CalcIsotopeChainC(L,k,m)
309     q = 1.0D0
310     Do i=k,m
311         q = q*L(i)*L(i)
312     End Do
313     r = (-1.0D0)*(p/q)
314     nChange = nChange + r*mult
315 ! Loop through pfrac
316     Do i=k,m
317         r = 1.0D0*L(i)*L(i)
318         Do j=k,m
319             If(j.ne.i)Then
320                 r = r * (L(i)-L(j))
321             End If
322         End Do
323         nChange = nChange + (1.0D0/r)*exp(-1.0D0*L(i)*t)*multP*mult
324     End Do
325 End Function CalcIsotopeChainG_Stable
326
327 Function CalcIsotopeChainC(L,k,m) Result (numerator)
328 ! Calculates numerator in isotope activity function
329 Implicit None ! Force declaration of all variables
330 ! Vars In
331 Real(kind=DoubleReal), Dimension(:) :: L
332 Integer(kind=StandardInteger) :: k, m
333 ! Vars Out
334 Real(kind=DoubleReal) :: numerator
335 ! Vars Private
336 Integer(kind=StandardInteger) :: i, j
337 Real(kind=DoubleReal) :: tempMult
338 numerator = 0.0D0
339 Do i=k,m
340     tempMult = 1.0D0
341     Do j=k,m
342         If(j.ne.i)Then
343             tempMult = tempMult * L(j)
344         End If
345     End Do
346     numerator = numerator + tempMult
347 End Do
348 End Function CalcIsotopeChainC

```

Bibliography

- [1] A. J. Koning D. Rochman. “Modern Nuclear Data Evaluation With The TALYS Code System”. In: *Nuclear Data Sheets* 113 (2012).
- [2] J. P. Biersack James F. Ziegler M. D. Ziegler. “SRIM - The stopping and range of ions in matter”. In: *Nuclear Instruments and Methods in Physics Research Section B* 268 (2010), pp. 1818–1823.
- [3] A. A. Sonzogni. *Nuclear Data Services*. www-nds.iaea.org. 2006. URL: <http://www-nds.iaea.org/ndspub/download-endf/ENDF-B-VII.0/decay-index.htm> (visited on 08/14/2015).
- [4] M. Herman M. B. Chadwick. “ENDF/B-VII.0: Next generation evaluated nuclear data library for nuclear science and technology.” In: *Nucl. Data Sheets* 107 (2006), p. 2931.
- [5] P. Blaise M. Coste A. Courcelle T.D. Huynh C. Jouanne P. Leconte O. Litaize S. Mengelle G. Nogure J-M. Ruggiri O. Srot J. Tommasi C. Vaglio J-F. Vidal A. Santamarina D. Bernard. *The JEFF-3.1.1 Nuclear Data Library*. 2009. ISBN: 978-92-64-99074-6.
- [6] H. Stehfest. “Algorithm 368: Numerical Inversion of Laplace Transform”. In: *Communications of the ACM* 13 (1970), pp. 47–49.
- [7] J. D. Hunter. “Matplotlib: A 2D graphics environment”. In: *Computing In Science & Engineering* 9 (2007), pp. 90–95.

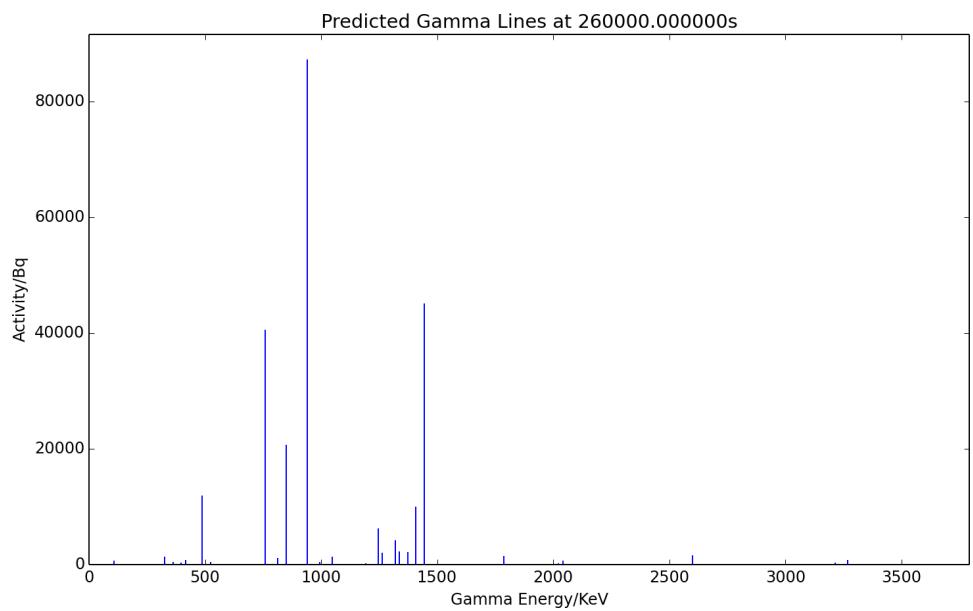


Figure C.1: Decay of Po-218: Analytic and Gaver-Stehfest Calculations [56]

Appendix D

Activity V2

D.1 Sample Input File

Alex Dickinson-Lomas from the University of Birmingham requested a calculation to predict the radioactivity of a thin 0.1mm piece of steel in a 5MeV proton beam. A SRIM exyz file had been prepared and the following input file was set up.

Listing D.1: Activity V2 Input File

```
1 # Data files
2 data_isotopes="/cloud/Code/python/activity/data/isotopes" xs="/cloud/Code/python/activity/data/xs"
3
4 # Sim
5 sim1 exyz="EXYZ.txt" target_composition=Fe,96.375,C,0.772,Cu,0.024,Mn,1.36,Ni,0.698,Si,0.381,Cr,0.092,V,0.008,P
   ,0.009,Si,0.003,Mo,0.278 target_depth=0.1,mm target_density=7808,kgm3 beam_projectile='proton' beam_energy=5,
   MeV beam_area=64,mm2 beam_duration=300,s beam_current=0.5,uA end_time=260000,s
```

D.2 Sample Output

The activity code processed the input file, and the calculation duration was approximately 15-20 minutes.

Listing D.2: Activity V2 Steel Terminal Output

```
1 Beam Details
2 =====
3 Projectile: Proton
4 Energy: None
5 Duration: 300.0
6 Area: 300.0
7 Current: 5e-07
8 Flux: 3120754564499.9995
9
10 Target
11
12 Depth: 0.0001 m
13 Density: 7808.0 kgm3
14 Mass: 55.107863928784624 amu
15
16 Composition:
17
18 Isotope Mass          PBM          PBN
19 =====
```

20	Fe54	5.3940E+01	5.6335E+00	5.7297E+00	4.3986E+02	4.9109E+27
21	Fe56	5.5935E+01	8.8428E+01	8.6731E+01	6.9045E+03	7.4336E+28
22	Fe57	5.6935E+01	2.0423E+00	1.9679E+00	1.5946E+02	1.6866E+27
23	Fe58	5.7933E+01	2.7216E-01	2.5773E-01	2.1250E+01	2.2090E+26
24	C12	1.2000E+01	7.6380E-01	3.4919E+00	5.9638E+01	2.9929E+27
25	C13	1.3003E+01	8.3222E-03	3.5111E-02	6.4979E-01	3.0094E+25
26	Cu63	6.2930E+01	1.6602E-02	1.4473E-02	1.2962E+00	1.2405E+25
27	Cu65	6.4928E+01	7.3999E-03	6.2526E-03	5.7779E-01	5.3590E+24
28	Mn55	5.4938E+01	1.3600E+00	1.3581E+00	1.0619E+02	1.1640E+27
29	Ni60	5.9931E+01	1.8304E-01	1.6755E-01	1.4292E+01	1.4361E+26
30	Ni61	6.0931E+01	7.9569E-03	7.1643E-03	6.2128E-01	6.1404E+24
31	Ni62	6.1928E+01	2.5369E-02	2.2474E-02	1.9808E+00	1.9262E+25
32	Ni64	6.3928E+01	6.4613E-03	5.5449E-03	5.0450E-01	4.7525E+24
33	Si28	2.7977E+01	2.7669E-03	5.4257E-03	2.1604E-01	4.6503E+24
34	Si29	2.8977E+01	1.4050E-04	2.6600E-04	1.0970E-02	2.2799E+23
35	Si30	2.9974E+01	9.2618E-05	1.6952E-04	7.2316E-03	1.4529E+23
36	Cr52	5.1941E+01	7.7086E-02	8.1421E-02	6.0189E+00	6.9785E+25
37	Cr53	5.2941E+01	8.7411E-03	9.0582E-03	6.8250E-01	7.7636E+24
38	Cr54	5.3939E+01	2.1764E-03	2.2137E-03	1.6994E-01	1.8973E+24
39	V51	5.0944E+01	7.9800E-03	8.5936E-03	6.2308E-01	7.3655E+24
40	P31	3.0974E+01	9.0000E-03	1.5941E-02	7.0272E-01	1.3663E+25
41	Mo94	9.3905E+01	2.5718E-02	1.5025E-02	2.0081E+00	1.2878E+25
42	Mo95	9.4906E+01	4.4261E-02	2.5586E-02	3.4559E+00	2.1929E+25
43	Mo96	9.5905E+01	4.6376E-02	2.6529E-02	3.6210E+00	2.2738E+25
44	Mo97	9.6906E+01	2.6571E-02	1.5043E-02	2.0747E+00	1.2893E+25

45

46 Starting Tally

47 =====

48	26054	Fe54	3.1429536676960092e+19
49	26056	Fe56	4.7574989041735316e+20
50	26057	Fe57	1.079453575560004e+19
51	26058	Fe58	1.4137464501727972e+18
52	6012	C12	1.9154473653970047e+19
53	6013	C13	1.9259847573081418e+17
54	29063	Cu63	7.938956724877902e+16
55	29065	Cu65	3.429786778008423e+16
56	25055	Mn55	7.449656284946808e+18
57	28060	Ni60	9.190961022927041e+17
58	28061	Ni61	3.929859828235443e+16
59	28062	Ni62	1.2327721842728802e+17
60	28064	Ni64	3.0415894209574764e+16
61	14028	Si28	2.9762044600901344e+16
62	14029	Si29	1459126206750858.0
63	14030	Si30	929869335659313.8
64	24052	Cr52	4.4662210592794496e+17
65	24053	Cr53	4.968735444487695e+16
66	24054	Cr54	1.214272751722711e+16
67	23051	V51	4.713925180516024e+16
68	15031	P31	8.744174716558614e+16
69	42094	Mo94	8.241841437276395e+16
70	42095	Mo95	1.4034621025229758e+17
71	42096	Mo96	1.4552033702553504e+17
72	42097	Mo97	8.251470736805168e+16
73	27053	Co53	0.0
74	25050	Mn50	0.0
75	26053	Fe53	0.0
76	26052	Fe52	0.0
77	25051	Mn51	0.0
78	25052	Mn52	0.0
79	24049	Cr49	0.0
80	27055	Co55	0.0

81	23047	V47	0.0
82	25053	Mn53	0.0
83	24050	Cr50	0.0
84	26055	Fe55	0.0
85	25054	Mn54	0.0
86	24051	Cr51	0.0
87	27057	Co57	0.0
88	23049	V49	0.0
89	27056	Co56	0.0
90	27058	Co58	0.0
91	23050	V50	0.0
92	25056	Mn56	0.0
93	27059	Co59	0.0
94	25057	Mn57	0.0
95	7014	N14	0.0
96	30062	Zn62	0.0
97	30061	Zn61	0.0
98	28059	Ni59	0.0
99	28058	Ni58	0.0
100	29062	Cu62	0.0
101	29061	Cu61	0.0
102	29060	Cu60	0.0
103	30064	Zn64	0.0
104	30063	Zn63	0.0
105	29064	Cu64	0.0
106	28063	Ni63	0.0
107	27060	Co60	0.0
108	30066	Zn66	0.0
109	27061	Co61	0.0
110	22048	Ti48	0.0
111	29059	Cu59	0.0
112	28057	Ni57	0.0
113	27062	Co62	0.0
114	26059	Fe59	0.0
115	27063	Co63	0.0
116	26060	Fe60	0.0
117	13024	Al24	0.0
118	14027	Si27	0.0
119	13026	Al26	0.0
120	15029	P29	0.0
121	14026	Si26	0.0
122	13025	Al25	0.0
123	11021	Na21	0.0
124	13027	Al27	0.0
125	12024	Mg24	0.0
126	12023	Mg23	0.0
127	15028	P28	0.0
128	15030	P30	0.0
129	11022	Na22	0.0
130	13028	Al28	0.0
131	12025	Mg25	0.0
132	11023	Na23	0.0
133	13029	Al29	0.0
134	12026	Mg26	0.0
135	23048	V48	0.0
136	22047	Ti47	0.0
137	21045	Sc45	0.0
138	21046	Sc46	0.0
139	23052	V52	0.0
140	22049	Ti49	0.0
141	21047	Sc47	0.0

```

142 23053      V53      0.0
143 22050      Ti50     0.0
144 22046      Ti46     0.0
145 20044      Ca44     0.0
146 16030      S30     0.0
147 16032      S32     0.0
148 10020      Ne20     0.0
149 43093      Tc93    0.0
150 43092      Tc92    0.0
151 41090      Nb90    0.0
152 41089      Nb89    0.0
153 42093      Mo93    0.0
154 42092      Mo92    0.0
155 42091      Mo91    0.0
156 41091      Nb91    0.0
157 41092      Nb92    0.0
158 40089      Zr89    0.0
159 43095      Tc95    0.0
160 39087      Y87     0.0
161 41093      Nb93    0.0
162 40090      Zr90    0.0
163 43094      Tc94    0.0
164 43096      Tc96    0.0
165 39088      Y88     0.0
166 41094      Nb94    0.0
167 40091      Zr91    0.0
168 43097      Tc97    0.0
169 39089      Y89     0.0
170 41095      Nb95    0.0
171 40092      Zr92    0.0
172 43098      Tc98    0.0
173 39090      Y90     0.0
174 41096      Nb96    0.0
175 40093      Zr93    0.0
176 1          Nn1     0.0
177 2004       He4     0.0
178 1001       H1      0.0
179 1002       H2      0.0
180 2003       He3     0.0
181 1003       H3      0.0
182 1025052    Mn52-M  0.0
183 1027060    Co60-M  0.0
184 1013026    Al126-M 0.0
185 10021      Ne21    0.0
186 10022      Ne22    0.0
187 1041093    Nb93-M  0.0
188 1040090    Zr90-M  0.0
189 1039089    Y89-M   0.0
190 1040089    Zr89-M  0.0
191 1041091    Nb91-M  0.0
192 1038087    Sr87-M  0.0
193 37087      Rb87    0.0
194 38087      Sr87    0.0
195 38088      Sr88    0.0
196 44098      Ru98    0.0
197
198
199 Run Sim
200 =====
201 Plot Ion Energy Lost
202

```

```

203 Ion Min/Max energies
204 =====
205 Max ion energy: 4950000.0
206 Min ion energy: 37528.5
207
208
209 Number Density
210 =====
211 Fe54      4.9108651057750145e+27
212 Fe56      7.4335920377711426e+28
213 Fe57      1.6866462118125062e+27
214 Fe58      2.2089788283949954e+26
215 C12       2.99288650843282e+27
216 C13       3.0093511832939716e+25
217 Cu63      1.2404619882621723e+25
218 Cu65      5.359041840638161e+24
219 Mn55      1.1640087945229387e+27
220 Ni60      1.43608765983235e+26
221 Ni61      6.140405981617879e+24
222 Ni62      1.926206537926375e+25
223 Ni64      4.752483470246057e+24
224 Si28      4.650319468890835e+24
225 Si29      2.2798846980482155e+23
226 Si30      1.4529208369676776e+23
227 Cr52      6.97847040512414e+25
228 Cr53      7.763649132012024e+24
229 Cr54      1.8973011745667357e+24
230 V51       7.365508094556287e+24
231 P31       1.3662772994622835e+25
232 Mo94      1.2877877245744368e+25
233 Mo95      2.1929095351921498e+25
234 Mo96      2.273755266023985e+25
235 Mo97      1.2892923026258074e+25
236
237 Calculate reaction rates
238
239 Reaction Rates
240 =====
241 26054 -6077.33918634
242 26056 -483804.853992
243 26057 -8178.39195382
244 26058 -1909.94156575
245 6013 -6.52012152744
246 29063 -1652.38584892
247 29065 -198.453009044
248 25055 -12210.915484
249 28060 1004.98674045
250 28061 -58.1671144121
251 28062 -9.28061752454
252 28064 -24.2493668631
253 14028 21113.5234259
254 14029 -0.0461122607693
255 14030 -0.645300628062
256 24052 7290.04169226
257 24053 -28.9023619139
258 24054 -10.2857142806
259 23051 -50.6697439819
260 15031 -21120.2628724
261 42094 -352.232677673
262 42095 -25.4917606732
263 42096 -155.434866026

```

```

264 42097 -3.89613150928
265 25051 1.54062703256e-08
266 27055 6077.33918633
267 25053 1493.67422813
268 25054 318.212949211
269 27057 487357.891417
270 23049 2.64460811847e-05
271 27058 7891.04027139
272 23050 0.00286551559081
273 27059 1854.07086162
274 7014 6.52012152744
275 29062 56.2053440309
276 29061 705.207008156
277 30064 124.84393152
278 30066 28.0983150696
279 27061 0.138917914494
280 22048 12.0822038765
281 15029 0.0853179405141
282 13027 0.383149852247
283 15030 0.0461122607692
284 16032 6.91627938818
285 41091 10.0503816208
286 41092 0.599431762138
287 43095 342.182296081
288 39087 6.58933417483e-16
289 41093 0.415508798647
290 40090 9.2262969886e-13
291 43096 24.8923288821
292 39088 8.97947641477e-14
293 41094 0.0754109801463
294 40091 9.70141425186e-15
295 43097 155.019357228
296 39089 5.86184699163e-15
297 40092 3.41253896966e-13
298 43098 3.82072052817
299 39090 3.32385561384e-15
300 40093 3.49317591935e-15
301 1 2.98443186628e-08
302 2004 32176.4777246
303 1001 61663841.0917
304 1002 7.0398304228e-12
305
306
307 Saturation Times
308 =====
309 Mn51      11980.384679
310 Co55      272748.238212
311 Mn53      5.01914153515e+14
312 Mn54      116576230.889
313 Co57      101493998.416
314 V49       123226813.841
315 Co58      26456423.5172
316 V50       1.90941486654e+25
317 Cu62      2528.32793551
318 Cu61      52371.3958826
319 Zn64      3.13696345947e+26
320 Co61      25672.2528836
321 P29       17.901426169
322 P30       647.770582862
323 Nb91      92745119373.8
324 Nb92      4.77365601936e+15

```

325	Tc95	311178.822832		
326	Y87	1241603.5031		
327	Tc96	1598214.43406		
328	Y88	39817188.8104		
329	Nb94	2.72592648801e+12		
330	Tc97	3.546133358e+14		
331	Tc98	5.86464032836e+14		
332	Y90	995772.233062		
333	Zr93	2.19582904139e+14		
334	Nn1	2656.25700712		
335				
336				
337	End of Beam			
338	=====			
339	26054	Fe54	3.1429536677e+19	0.0
340	26056	Fe56	4.75749890417e+20	0.0
341	26057	Fe57	1.07945357559e+19	0.0
342	26058	Fe58	1.41374645018e+18	0.0
343	6012	C12	1.9154473653970047e+19	0.0
344	6013	C13	1.92598475731e+17	0.0
345	29063	Cu63	7.93895672483e+16	0.0
346	29065	Cu65	3.429786778e+16	0.0
347	25055	Mn55	7.44965628495e+18	0.0
348	28060	Ni60	9.19096102293e+17	0.0
349	28061	Ni61	3.92985982828e+16	0.0
350	28062	Ni62	1.23277218427e+17	0.0
351	28064	Ni64	3.04158942096e+16	0.0
352	14028	Si28	2.97620446072e+16	0.0
353	14029	Si29	1.45912620675e+15	0.0
354	14030	Si30	9.29869335659e+14	0.0
355	24052	Cr52	4.4662210593e+17	0.0
356	24053	Cr53	4.96873544458e+16	0.0
357	24054	Cr54	1.21427275174e+16	0.0
358	23051	V51	4.71392518051e+16	0.0
359	15031	P31	8.74417471593e+16	0.0
360	42094	Mo94	8.24184143727e+16	0.0
361	42095	Mo95	1.40346210252e+17	0.0
362	42096	Mo96	1.45520337026e+17	0.0
363	42097	Mo97	8.25147073681e+16	0.0
364	25051	Mn51	4.45277850945e-06	1.1134310496e-09
365	24051	Cr51	1.69097660998e-07	4.89672869074e-14
366	27055	Co55	1820201.2799	19.9921940992
367	26055	Fe55	3000.47354256	2.39917424244e-05
368	25053	Mn53	448102.268439	2.67454985681e-09
369	25054	Mn54	95463.5167853	0.00245318566229
370	27057	Co57	146206720.102	4.31548856931
371	23049	V49	0.00793379542383	1.92876260142e-10
372	27058	Co58	2367271.87331	0.268052586418
373	27059	Co59	556221.258486	0.0
374	7014	N14	1956.03645823	0.0
375	29062	Cu62	14190.4346381	16.8137773681
376	29061	Cu61	209757.185203	11.9984651685
377	30066	Zn66	8429.49452087	0.0
378	27061	Co61	40.9543387597	0.00477902094999
379	22048	Ti48	3624.66116296	0.0
380	15029	P29	0.509829542009	0.0853179405141
381	13027	A127	114.944955674	0.0
382	15030	P30	7.48094498488	0.0345969837483
383	16032	S32	2074.88381646	0.0
384	41091	Nb91	3015.11448623	9.73903083617e-08
385	40091	Zr91	2.91042427556e-12	0.0

386	41092	Nb92	179.829528641	1.12852941331e-13
387	40092	Zr92	1.0237616909e-10	0.0
388	43095	Tc95	102506.592248	0.986835491746
389	39087	Y87	1.97608498356e-13	4.76788406747e-19
390	1038087	Sr87-M	7.10403001573e-17	4.85730190185e-21
391	38087	Sr87	3.9568373186e-13	0.0
392	41093	Nb93	124.652639594	0.0
393	40090	Zr90	2.78782773177e-10	0.0
394	43096	Tc96	7465.59941256	0.0139937023624
395	39088	Y88	2.6938125231e-11	2.0267480843e-18
396	38088	Sr88	5.38765544753e-11	0.0
397	41094	Nb94	22.6232940439	2.48624944215e-11
398	43097	Tc97	46505.8071684	3.92875657448e-10
399	39089	Y89	1.75855409749e-12	0.0
400	43098	Tc98	1146.21615845	5.8550167548e-12
401	44098	Ru98	2292.4932886	0.0
402	39090	Y90	9.96706834899e-13	2.99854397767e-18
403	40093	Zr93	1.0479527758e-12	1.42970417663e-26
404	1041093	Nb93-M	2.91835212081e-22	3.97651955467e-31
405	1	Nn1	7.59596051337e-06	8.56674034086e-09
406	1001	H1	18499152327.5	0.0
407	2004	He4	9652943.31739	0.0
408	1002	H2	2.11194912684e-09	0.0
409				
410		End of Sim		
411		=====		
412	26054	Fe54	3.1429536677e+19	0.0
413	26056	Fe56	4.75749890417e+20	0.0
414	26057	Fe57	1.07945357559e+19	0.0
415	26058	Fe58	1.41374645018e+18	0.0
416	6012	C12	1.9154473653970047e+19	0.0
417	6013	C13	1.92598475731e+17	0.0
418	29063	Cu63	7.93895672483e+16	0.0
419	29065	Cu65	3.429786778e+16	0.0
420	25055	Mn55	7.44965628495e+18	0.0
421	28060	Ni60	9.19096102293e+17	0.0
422	28061	Ni61	3.9298598283e+16	0.0
423	28062	Ni62	1.23277218427e+17	0.0
424	28064	Ni64	3.04158942096e+16	0.0
425	14028	Si28	2.97620446072e+16	0.0
426	14029	Si29	1.45912620675e+15	0.0
427	14030	Si30	9.29869335659e+14	0.0
428	24052	Cr52	4.4662210593e+17	0.0
429	24053	Cr53	4.96873544458e+16	0.0
430	24054	Cr54	1.21427275174e+16	0.0
431	23051	V51	4.71392518051e+16	0.0
432	15031	P31	8.74417471593e+16	0.0
433	42094	Mo94	8.24184143727e+16	0.0
434	42095	Mo95	1.40346210253e+17	0.0
435	42096	Mo96	1.45520337026e+17	0.0
436	42097	Mo97	8.25147073681e+16	0.0
437	25051	Mn51	2.79301596638e-34	6.98402288008e-38
438	24051	Cr51	4.29182988878e-06	1.24282774984e-12
439	27055	Co55	105034.147734	1.1536433315
440	26055	Fe55	1715632.35011	0.0137181711003
441	25053	Mn53	448102.267745	2.67454985267e-09
442	25054	Mn54	94828.5456205	0.00243686841137
443	27057	Co57	145090272.195	4.2825351033
444	23049	V49	0.00788386324816	1.91662373119e-10
445	27058	Co58	2298672.19924	0.260284860087
446	27059	Co59	556221.258486	0.0

447	7014	N14	1956.03645823	0.0
448	29062	Cu62	3.27537379744e-130	3.88088224441e-133
449	29061	Cu61	0.0741568656357	4.24189792818e-06
450	30066	Zn66	8429.49452087	0.0
451	27061	Co61	2.8255717303e-12	3.29720046856e-16
452	22048	Ti48	3624.66116296	0.0
453	13027	Al27	114.944955674	0.0
454	16032	S32	2074.88381646	0.0
455	41091	Nb91	3015.08919407	9.73894914074e-08
456	40091	Zr91	2.91042427556e-12	0.0
457	41092	Nb92	179.829528612	1.12852941313e-13
458	40092	Zr92	1.0237616909e-10	0.0
459	43095	Tc95	8413.0472009	0.0809927770448
460	39087	Y87	1.05602858852e-13	2.54797841382e-19
461	1038087	Sr87-M	3.86284994055e-15	2.64118089609e-19
462	37087	Rb87	2.64641489592e-16	1.20849009065e-34
463	38087	Sr87	3.9568373186e-13	0.0
464	41093	Nb93	124.652639594	0.0
465	40090	Zr90	2.79323174816e-10	0.0
466	43096	Tc96	4588.33716227	0.00860049153979
467	39088	Y88	2.64168875912e-11	1.9875316437e-18
468	38088	Sr88	5.38765544753e-11	0.0
469	41094	Nb94	22.6232875871	2.48624873256e-11
470	43097	Tc97	46505.8070664	3.92875656586e-10
471	39089	Y89	1.75855409749e-12	0.0
472	43098	Tc98	1146.21615693	5.85501674703e-12
473	44098	Ru98	2292.4932886	0.0
474	39090	Y90	4.56305196266e-13	1.37277196296e-18
475	40093	Zr93	1.04795277209e-12	1.42970417157e-26
476	1041093	Nb93-M	3.001699921e-21	4.09008849481e-30
477	1	Nn1	4.78609027493e-133	5.39776273996e-136
478	1001	H1	18499152327.5	0.0
479	2004	He4	9652943.31739	0.0
480	1002	H2	2.11194912684e-09	0.0
481				
482	Gamma Dose - Beam End			
483	=====			
484				
485	Activity/Bq		54.4306601689	
486	Power eV/s		28342871818.2	
487	Power J/s		4.54103823697e-09	
488	Dose Gy/s		4.51575003676e-12	
489	Dose Gy/hr		1.62567001323e-08	
490	Percentage of annual dose/hr		0.0142506237521	
491				
492	Gamma Dose - Sim End			
493	=====			
494				
495	Activity/Bq		5.80221584491	
496	Energy eV/s		2227911995.02	
497	Energy J/s		3.56951604018e-10	
498	Dose Gy		8.52319971389e-13	
499	Dose Gy/hr		1.27786970412e-09	
500	Percentage of annual dose/hr		0.00112018061534	
501				
502				
503	Absorbed Dose Calculations			
504	=====			
505	Absorbed dose assumptions:			
506	1. radiation from point, emitted isotropically			
507	2. 80Kg human			

```
508 3. 1m from point source
509 4. 1m squared surface area
510 5. all energy absorbed
511
512
513 Dose Limits
514 =====
515 employees 18+      20 millisieverts/year
516 trainees 18+       6 millisieverts/year
517 public and under 18s 1 millisievert/year
518 public and under 18s 1.140771128E-04 millisieverts/hour
519
520 Dose averaged over area of skin not exceeding 1cm2
521 Source: http://www.hse.gov.uk/radiation/ionising/doses/
522
523
524
525 Run Time: 1007.034s
```

D.3 Output Plots

The program produces many different plots, including those of the cross sections for each target isotope. Four of the most important plots for the 5MeV proton irradiated steel are included here.

First is the total activity, from the time the beam starts until the beam is turned off. It shows a steady rise in activity, and from this graph it is hard to see any curve towards a saturation of radioactive isotopes as the beam duration is so short.

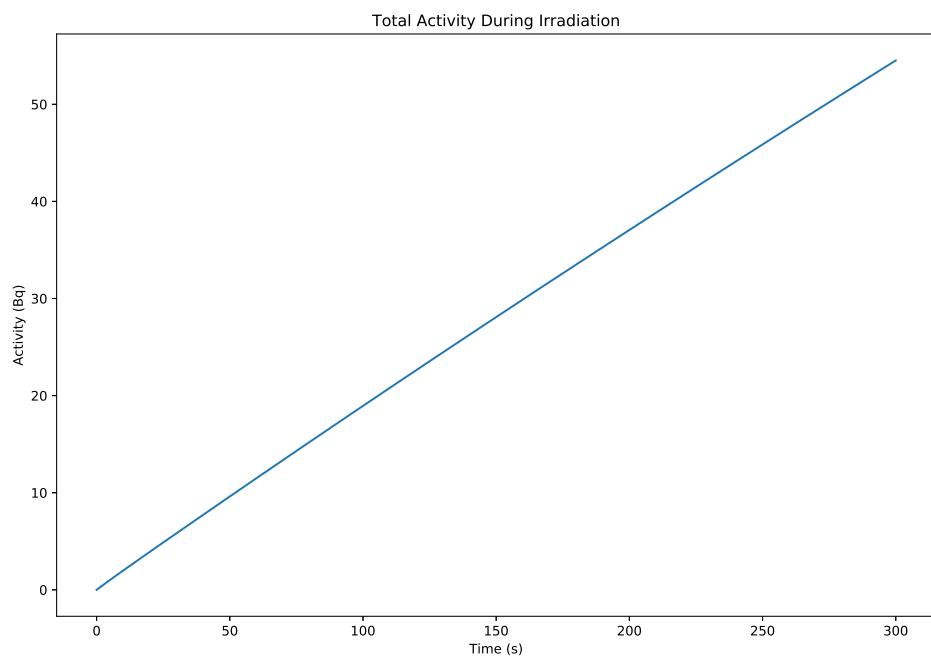


Figure D.1: Decay of Po-218: Analytic and Gaver-Stehfest Calculations [56]

The cooling period is much longer than the time spent within the beam, so a second graph shows the overall activity from when irradiation begins, to when it ceases up until the end of the simulation time.

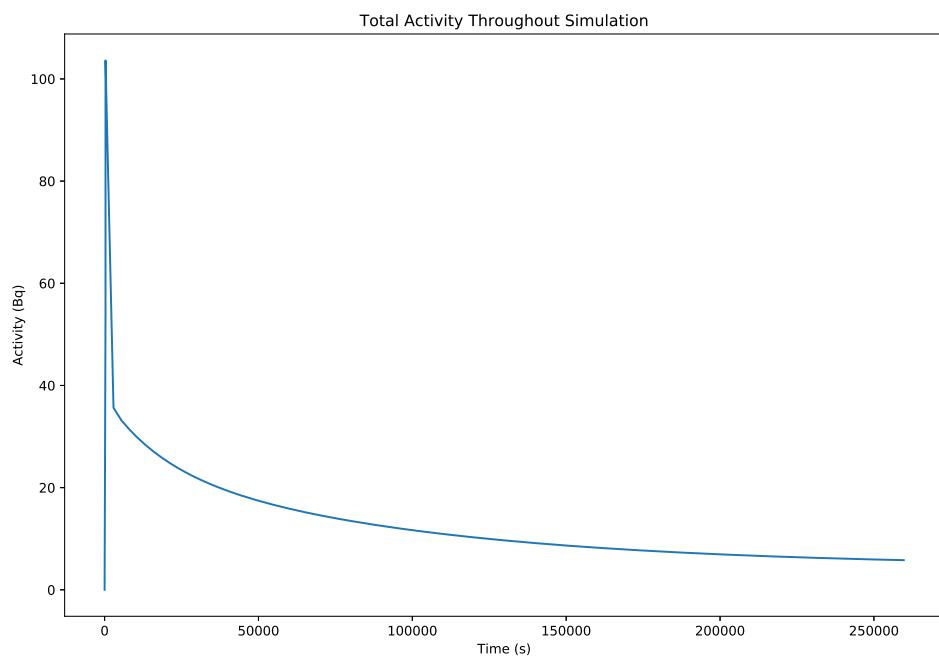


Figure D.2: Decay of Po-218: Analytic and Gaver-Stehfest Calculations [56]

The predicted gamma lines at the end of irradiation are predicted in the end of beam gamma line plot.

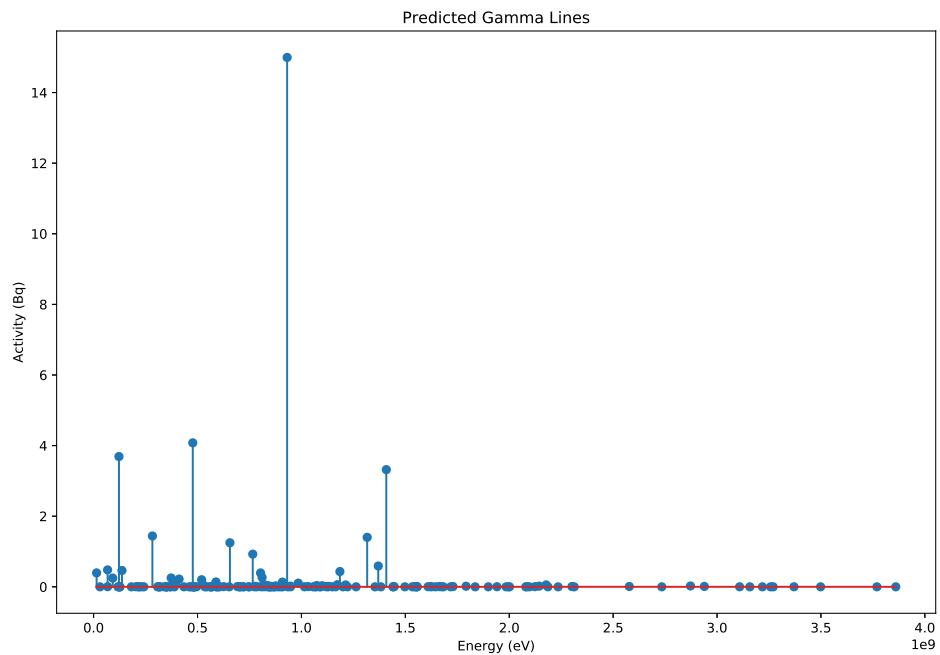


Figure D.3: Decay of Po-218: Analytic and Gaver-Stehfest Calculations [56]

The predicted gamma lines at the end of simulation are predicted in the end of sim gamma line plot.

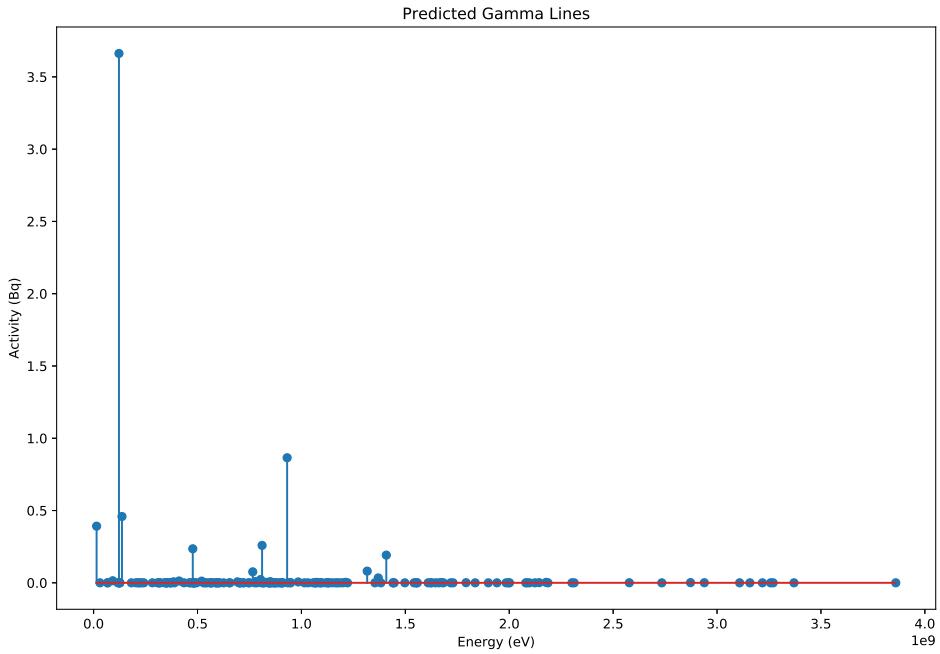


Figure D.4: Decay of Po-218: Analytic and Gaver-Stehfest Calculations [56]

D.4 An Example to Show Saturation In Beam

With the first steel example, the beam duration was too short to show a saturation of radioactive isotopes.

Listing D.3: Activity V2 input file: steel irradiated for 1 day

```

1 # Data files
2 data_isotopes="/cloud/Code/python/activity/data/isotopes" xs="/cloud/Code/python/activity/data/xs"
3
4 # Sim
5 sim1 exyz="EXYZ.txt" target_composition=Fe,96.375,C,0.772,Cu,0.024,Mn,1.36,Ni,0.698,Si,0.381,Cr,0.092,V,0.008,P
   ,0.009,Si,0.003,Mo,0.278 target_depth=0.1,mm target_density=7808,kgm3 beam_projectile='proton' beam_energy=5,
   MeV beam_area=64,mm2 beam_duration=300,s beam_current=0.5,uA end_time=260000,s

```

D.5 Accompanying Manual

UNIVERSITY OF BIRMINGHAM



Department of Metallurgy & Materials

Activity V2 Manual

Contents

1	Overview	2
1.1	Reason for the Program	2
1.2	Program Requirements	2
2	Installation	3
3	How To Use	4
3.1	SRIM	4
3.2	Activity	5
3.3	Input Keywords	11
4	Examples	12
4.1	Iron - 28MeV Protons	12
4.2	Iron Fe56 (only) - 28MeV Protons	12
4.3	Steel - 5MeV Protons	12
5	Decay Equation	14
5.1	Bateman Equation	14
5.1.1	Laplace Transform	14
5.1.2	Constructing the Differential Equations	15
5.1.3	Numerical Inversion of the Laplace Transform	15
5.1.4	Analytic Solution by Partial Fraction Expansion	16
5.1.5	Preference: Analytic over Numeric	18
5.2	Python Isotopes Class	19
6	Future Plans	22
6.1	Selection of Cross Section Databases	22

Chapter 1

Overview

1.1 Reason for the Program

The Activity program calculates how radioactive a target becomes after being irradiated by high energy ions. It uses the TENDL-2019[1] database that contains cross-section data for protons and deuterons, and the JEFF-3.3 Radioactive Decay Data File[2].

1.2 Program Requirements

The user must provide a exyz file from the SRIM[3] ion transport program and a breakdown of the composition of the target, as well as other irradiation parameters (beam projectile, beam duration, beam area, target thickness, target composition, simulation end time etc).

Chapter 2

Installation

The program needs Python 3 installed in order to run. At the time of writing, it has only been developed to run on a Linux operating system, but it shouldn't require much adjusting to run on a Windows computer too.

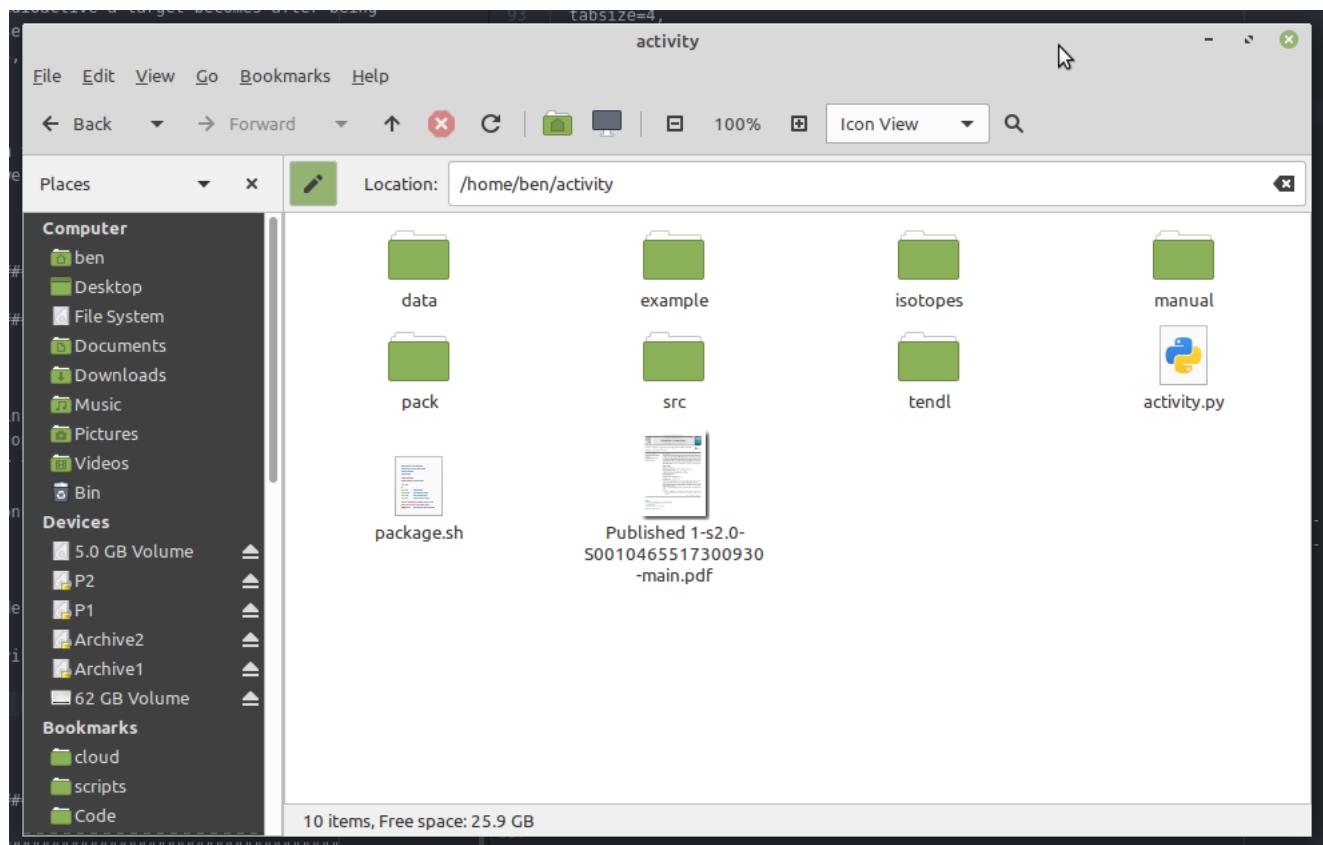
Download and install the latest version of Python 3:

<https://www.python.org/downloads>

The latest version of the activity code with data files must also be downloaded:

https://github.com/BenPalmer1983/activity_v2

For example, I have installed to /home/ben/activity

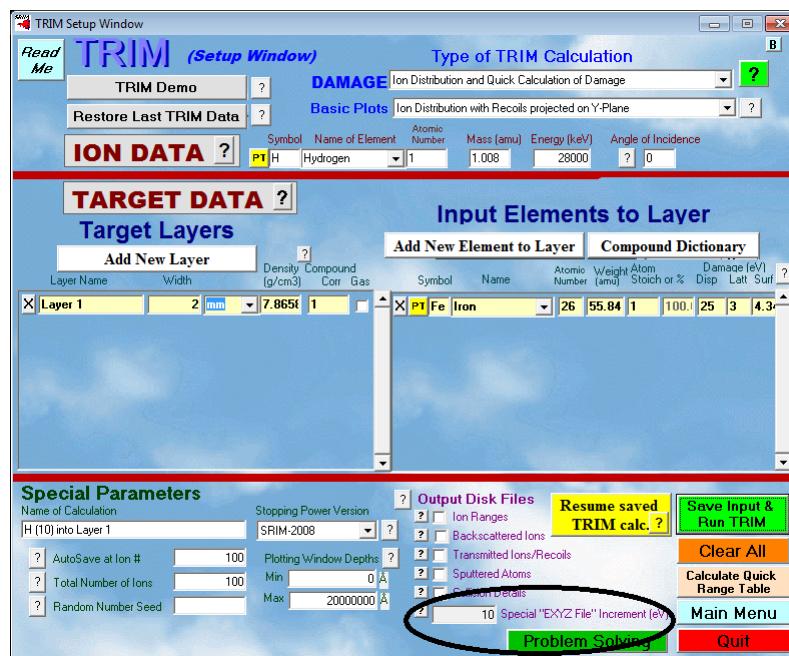


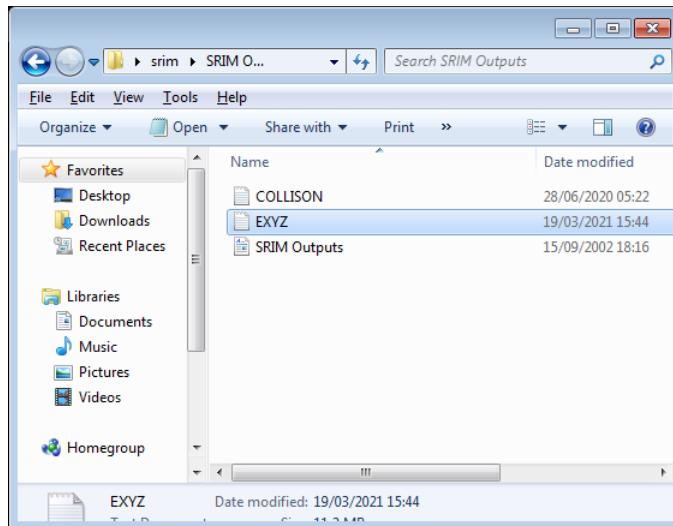
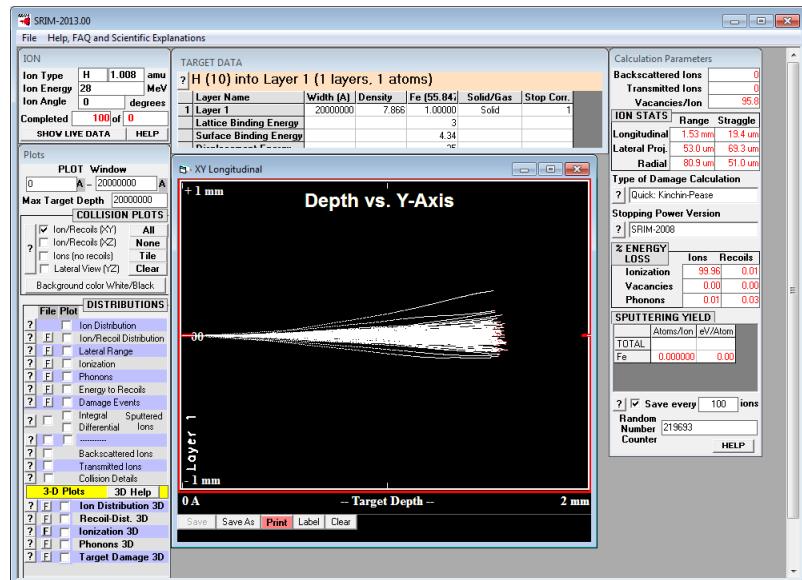
Chapter 3

How To Use

3.1 SRIM

If you haven't done so already, run the required simulation in SRIM. When setting up the calculation, be sure to set an increment for the EXYZ file. Sane values that have been used in examples range from 10eV to 10,000KeV, but this will depend on the resolution of the data in the TENDL database, the thickness of your target and the energy of the projectiles.

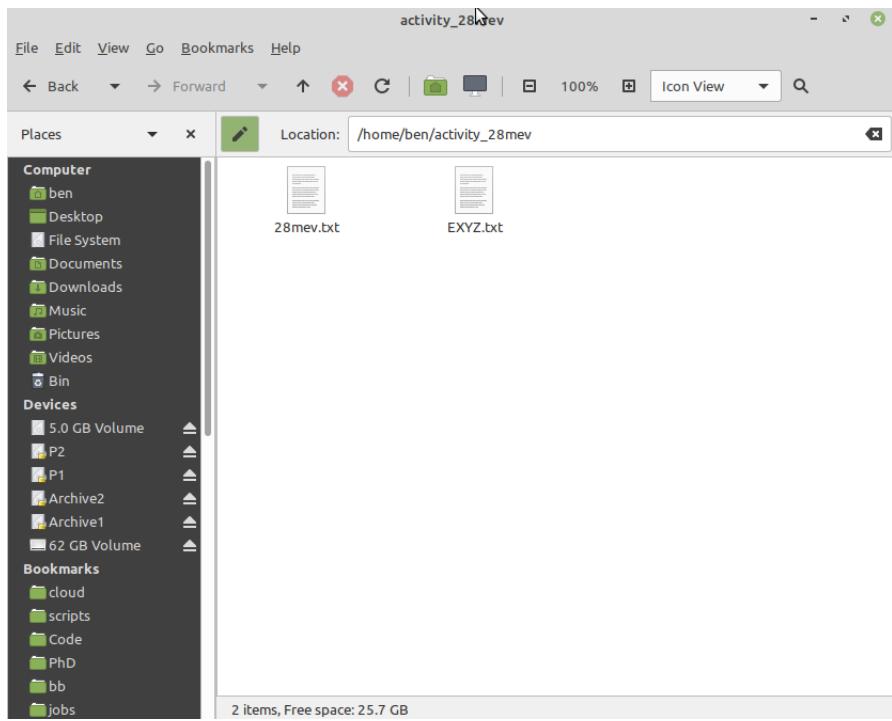




Take a copy of the EXYZ.txt file as it will be required by the activity program.

3.2 Activity

Create a directory to run the calculation. Copy in the EXYZ.txt file and create an input file.



The input file is just a text file, and will specify the calculation details. The data file paths must be specified, and then the simulations can be detailed, each with it's own unique name. The example below is for two simulations - the first with a 300s beam, and the second with a 3000s beam.

```

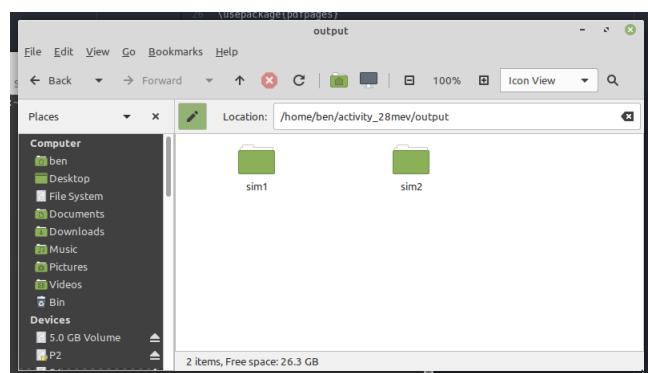
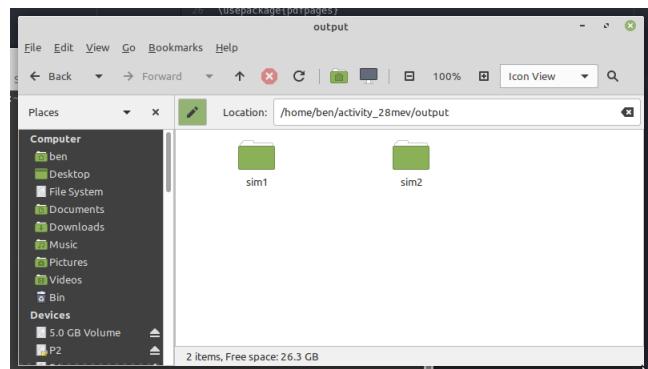
1 # Data files
2 data_isotopes="/home/ben/activity/data/isotopes" xs="/home/ben/activity/data/xs"
3
4 # Sim 1
5 sim1 exyz="EXYZ.txt" target_composition=Fe,100.0 target_depth=0.5,mm target_density=7808,kgm3
   beam_projectile='proton' beam_energy=28,MeV beam_area=64,mm2 beam_duration=300,s beam_current=0.5,
   uA end_time=260000,s
6
7 # Sim 2
8 sim2 exyz="EXYZ.txt" target_composition=Fe,100.0 target_depth=0.5,mm target_density=7808,kgm3
   beam_projectile='proton' beam_energy=28,MeV beam_area=64,mm2 beam_duration=3000,s beam_current
   =5.0,uA end_time=260000,s

```

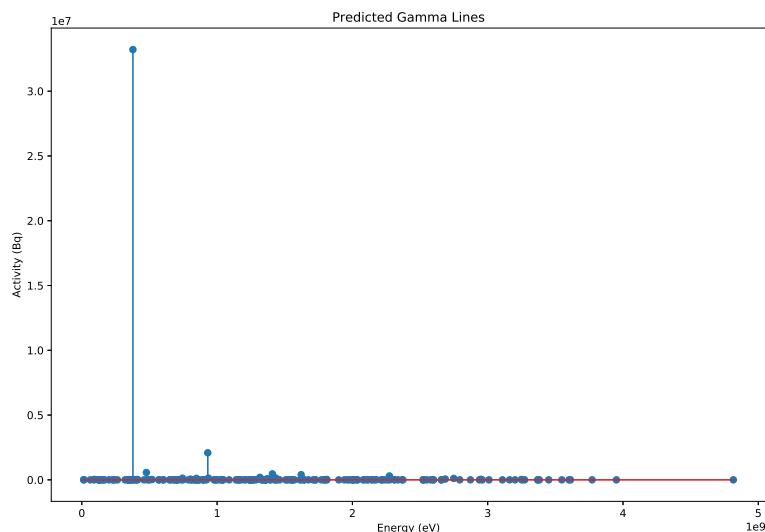
The program is run from the terminal/command line. Depending on your computer system, python3 might run through the command python3 or python, and on my computer the command is python3.

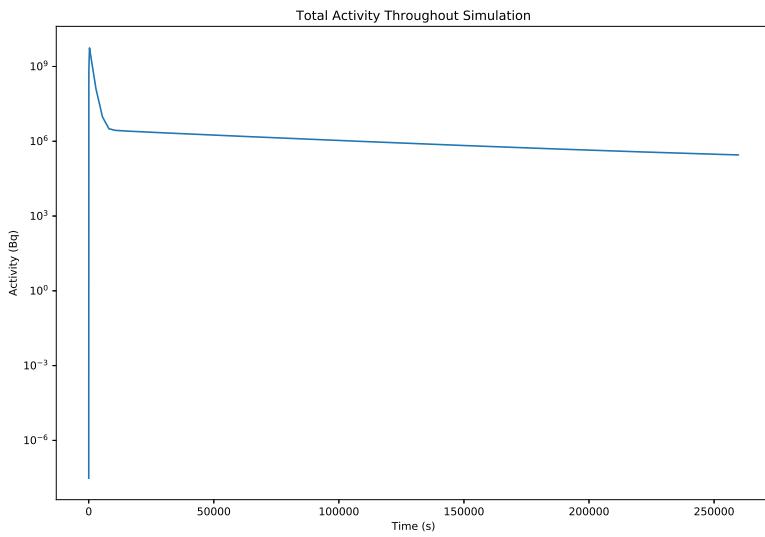
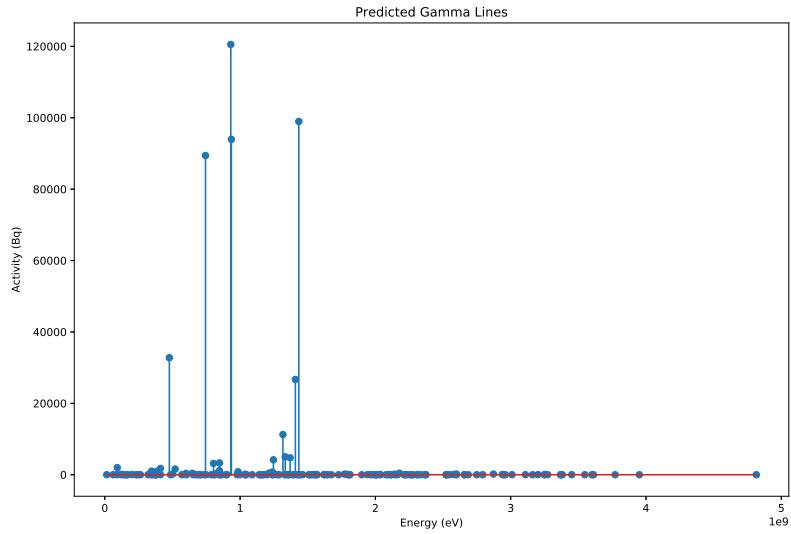
```
1 python3 /home/ben/activity/activity.py /home/ben/activity_28mev/28mev.txt
```

The program outputs into a directory for each simulation.



Within each directory are various plots and data files, including the activity over time and predicted gamma lines.





The program analyses the exyz file and output a chart showing the distribution of energy lost by projectiles as they pass through the target.

As a target is irradiated, the residual radioactive isotopes begin to decay. To begin with, the production outweighs the decay, but at a certain point there's enough of the radioactive isotope within the target that the decay balances with the production. At this point the radioactive isotope in the target is saturated, and will not increase above this amount unless the beam parameters are adjusted.

The saturation times depend on the isotope and the various reaction cross sections with the projectile. The saturation plots for each radioactive isotope are created and saved by the program.

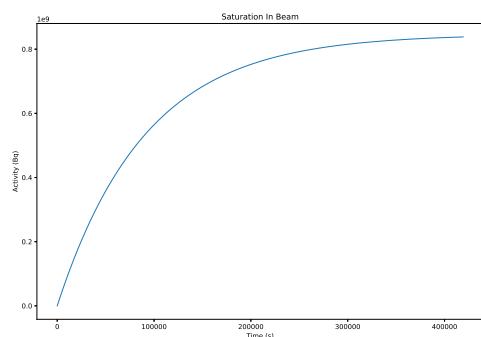
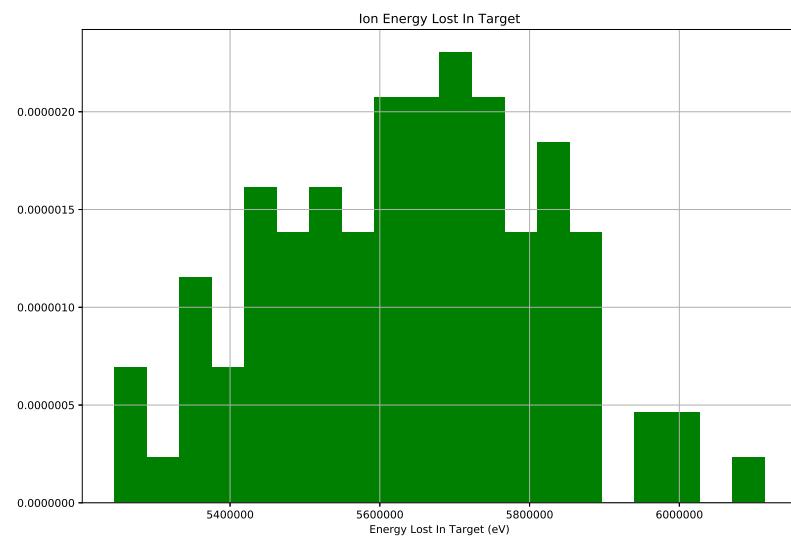


Figure 3.1: Saturation of Cobalt-55

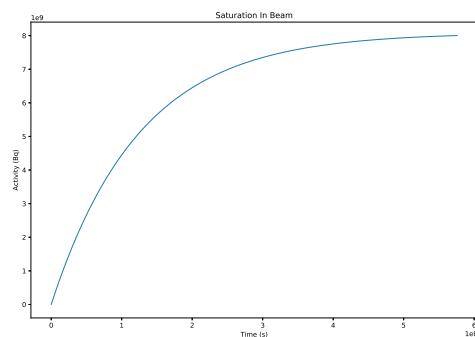


Figure 3.2: Saturation of Iron-55

The program calculates the dose a human may receive from the target at the time it is removed from the beam, and at the time the simulation stops.

Listing 3.1: Predicted Dose

```
1 Gamma Dose - Beam End
2 =====
3
4 Activity/Bq          54.4306601689
5 Power eV/s           28342871818.2
6 Power J/s            4.54103823697e-09
7 Dose Gy/s            4.51575003676e-12
8 Dose Gy/hr           1.62567001323e-08
9 Percentage of annual dose/hr 0.0142506237521
10
11 Gamma Dose - Sim End
12 =====
13
14 Activity/Bq          5.80221584491
15 Energy eV/s           2227911995.02
16 Energy J/s            3.56951604018e-10
17 Dose Gy               8.52319971389e-13
18 Dose Gy/hr            1.27786970412e-09
19 Percentage of annual dose/hr 0.00112018061534
20
21
22 Absorbed Dose Calculations
23 =====
24 Absorbed dose assumptions:
25 1. radiation from point, emitted isotropically
26 2. 80Kg human
27 3. 1m from point source
28 4. 1m squared surface area
29 5. all energy absorbed
30
31
32 Dose Limits
33 =====
34 employees 18+          20 millisieverts/year
35 trainees 18+            6 millisieverts/year
36 public and under 18s    1 millisievert/year
37 public and under 18s    1.140771128E-04 millisieverts/hour
38
39 Dose averaged over area of skin not exceeding 1cm2
40 Source: http://www.hse.gov.uk/radiation/ionising/doses/
```

3.3 Input Keywords

Command	<i>Sub Command</i>	Description
data		data fields are input here
data	isotopes	path to isotopes directory
data	xs	path to cross section directory
sim1		each simulation is numbered sequentially from 1 i.e. sim1 sim2 sim3...
sim1	exyz	path to exyz SRIM file
sim1	target_composition	isotopes that make up the beam target
sim1	target_depth	the depth of material through which the beam passes
sim1	target_density	how dense (specify kgm3 or GCM3)
sim1	beam_projectile	currently, only "proton", but this could be extended in the future
sim1	beam_energy	energy of the protons/projectiles
sim1	beam_area	how much area the beam covers
sim1	beam_duration	how long the beam will be on for (specify s,minute,hour) t=0 to beam_duration is the first part of the calculation
sim1	end_time	the time the calculation and final activity is measured after t=0 (specify s,minute,hour) end_time to beam_duration is the second part of the calculation

Chapter 4

Examples

4.1 Iron - 28MeV Protons

In this example, natural iron makes up the target .

```
1 # Data files
2 data isotopes="/home/ben/activity/data/isotopes" xs="/home/ben/activity/data/xs"
3
4 # Sim
5 sim1 exyz="EXYZ.txt" target_composition=Fe,100.0 target_depth=0.5,mm target_density=7808,kgm3
   beam_projectile='proton' beam_energy=28,MeV beam_area=64,mm2 beam_duration=300,s beam_current=0.5,
   uA end_time=260000,s
```

The simulation section defines what the target material is made of as well as the beam parameters.

4.2 Iron Fe56 (only) - 28MeV Protons

In this example, the target is made of Fe56 only, and contains none of the other naturally occurring stable isotopes.

```
1 # Data files
2 data isotopes="/home/ben/activity/data/isotopes" xs="/home/ben/activity/data/xs"
3
4 # Sim
5 sim1 exyz="EXYZ.txt" target_composition=Fe56,100.0 target_depth=0.5,mm target_density=7808,kgm3
   beam_projectile='proton' beam_energy=28,MeV beam_area=64,mm2 beam_duration=300,s beam_current=0.5,
   uA end_time=260000,s
```

4.3 Steel - 5MeV Protons

This example was provided by Alex Dickinson-Lomas, with a steel containing a wider range of elements.

```
1 # Data files
2 data isotopes="/home/ben/activity/data/isotopes" xs="/home/ben/activity/data/xs"
3
```

```
4 # Sim
5 sim1 exyz="EXYZ.txt" target_composition=Fe,96.375,C,0.772,Cu,0.024,Mn,1.36,Ni,0.698,Si,0.381,Cr,0.092,
  V,0.008,P,0.009,Si,0.003,Mo,0.278 target_depth=0.1,mm target_density=7808,kgm3 beam_projectile='
  proton' beam_energy=5,MeV beam_area=64,mm2 beam_duration=300,s beam_current=0.5,uA end_time
  =260000,s
```

Chapter 5

Decay Equation

5.1 Bateman Equation

The Bateman equation was derived using Laplace transforms, and this same method has been used to develop a modified equation that incorporates branching factors and production rates for each isotope in the decay chain, as illustrated by Figure 5.1.

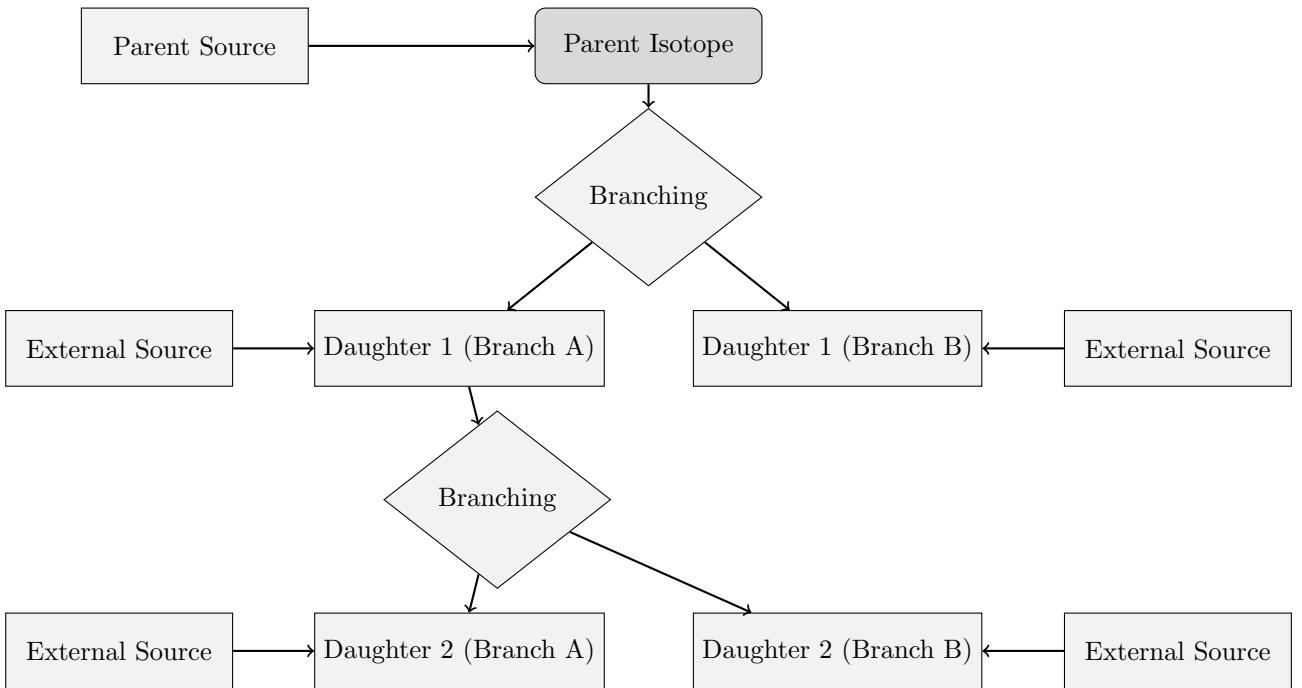


Figure 5.1: An example of several decay chains including branching factors and possible external source terms for each isotope on each chain.

5.1.1 Laplace Transform

Laplace Transforms (5.1) are a useful mathematical tool, and allow ordinary differential equations to be solved by simple algebraic manipulation in the s domain. Bateman took advantage of Laplace Transforms in deriving his equation, and this is the method that has been taken here as well.

$$F(s) = \int_0^{\infty} f(t) \exp(-st) dt \quad (5.1)$$

5.1.2 Constructing the Differential Equations

The first step is to set up differential equations for the parent isotope, unstable daughter isotopes and stable daughter isotope. The parent isotope has a source term, due to production, and a loss term, due to decay. The unstable daughter isotopes have two source terms, from the production by irradiation induced transmutation and the decay of preceding isotopes in the decay chain, and a loss term, due to decay. Finally, the stable daughter that finalizes the decay chain has two source terms (the same as the unstable daughters) but no loss term.

The variables (and vectors) used in these equations are defined as follows:

- $\vec{\lambda}$ vector containing isotope decay constants λ_i
- \vec{b} vector containing isotope to isotope branching factors b_i
- \vec{w} vector containing isotope production rates w_i
- t time at which activity/amount of isotope is measured
- $N_i(0)$ starting amount of the i^{th} isotope
- $N_i(t)$ amount of the i^{th} isotope at time t
- $N'_i(t)$ change in amount of the i^{th} isotope, with respect to time, at time t

The differential equations for the parent isotope (first isotope), unstable daughter isotopes (i^{th} isotopes) and stable, final, daughter isotope (z^{th} isotope) in the time domain are as follows:

$$N'_1(t) = \omega_1 - \lambda_1 N_1(t) \quad (5.2)$$

$$N'_i(t) = \omega_i + b_{i-1} \lambda_{i-1} N_{i-1}(t) - \lambda_i N_i(t) \quad (5.3)$$

$$N'_z(t) = \omega_z + b_{z-1} \lambda_{z-1} N_{z-1}(t) \quad (5.4)$$

Applying the Laplace Transform to these three differential equations allows them to be manipulated and solved algebraically in the s-domain.

$$N_1(s) = \frac{1}{s + \lambda_1} N_1(0) + \frac{1}{s(s + \lambda_1)} \omega_1 \quad (5.5)$$

$$N_i(s) = \frac{1}{s(s + \lambda_i)} (\omega_i) + \frac{1}{s + \lambda_i} (b_{i-1} \lambda_{i-1} N_{i-1}(s)) + \frac{1}{s + \lambda_i} N_i(0) \quad (5.6)$$

$$N_z(s) = \frac{1}{s^2} \omega_z + \frac{1}{s} b_{z-1} \lambda_{z-1} N_{z-1}(s) + \frac{1}{s} N_z(0) \quad (5.7)$$

5.1.3 Numerical Inversion of the Laplace Transform

The Gaver-Stehfest[4] algorithm was developed in the 1960s and 1970s and is a method of calculating the inverse of a Laplace Transform in the real number domain. It is an easy to implement and reasonably accurate method,

although it is an approximation to the real value. A comparison between an analytic and numeric inversion for the unstable isotope Po-218 is discussed at the end of this section (figure 5.2).

$$f(t) \approx f_n(t) = \frac{\ln(2)}{t} \sum_{k=1}^{2n} a_k(n) F(s) \text{ where } n \geq 1, t > 0 \quad (5.8)$$

$$s = \frac{k \ln(2)}{t} \quad (5.9)$$

$$a_k(n) = \frac{(-1)^{(n+k)}}{n!} \sum_{j=\text{Floor}(\frac{k+1}{2})} j^{n+1} \binom{n}{j} \binom{2j}{j} \binom{j}{k-j} \quad (5.10)$$

The equation for the i^{th} isotope may be calculated by recursively calculating the equations by numeric inversion, starting from the first (parent isotope) and inserting the result into each subsequent recursion until the i^{th} isotope is reached (changing the equations appropriately for the parent, unstable daughter and stable daughter isotopes).

5.1.4 Analytic Solution by Partial Fraction Expansion

The equation for the i^{th} isotope in the s domain can be written in full by substituting the preceding equation until the parent isotope is reached, and this full equation may be rearranged with the production amount of each isotope and starting amount of each isotope in individual terms. Each of these terms is multiplied by a fraction that can be expanded, using partial fractions, and inverted analytically.

This is illustrated with an example unstable isotope, fourth in the decay chain (including the parent isotope):

$$\begin{aligned} N_4(s) = & \frac{1}{(s + \lambda_1)(s + \lambda_2)(s + \lambda_3)(s + \lambda_4)} b_2 b_3 b_4 \lambda_1 \lambda_2 \lambda_3 N_1(0) \\ & + \frac{1}{(s + \lambda_2)(s + \lambda_3)(s + \lambda_4)} b_3 b_4 \lambda_2 \lambda_3 N_2(0) \\ & + \frac{1}{(s + \lambda_3)(s + \lambda_4)} b_4 \lambda_3 N_3(0) \\ & + \frac{1}{(s + \lambda_4)} N_4(0) \\ & + \frac{1}{s(s + \lambda_1)(s + \lambda_2)(s + \lambda_3)(s + \lambda_4)} b_2 b_3 b_4 \lambda_1 \lambda_2 \lambda_3 \omega_1 \\ & + \frac{1}{s(s + \lambda_2)(s + \lambda_3)(s + \lambda_4)} b_3 b_4 \lambda_2 \lambda_3 \omega_2 \\ & + \frac{1}{s(s + \lambda_3)(s + \lambda_4)} b_4 \lambda_3 \omega_3 \\ & + \frac{1}{s(s + \lambda_4)} \omega_4 \end{aligned} \quad (5.11)$$

An example stable isotope, fourth (last) in the decay chain (including the parent isotope):

$$\begin{aligned}
N_4(s) = & \frac{1}{s(s+\lambda_1)(s+\lambda_2)(s+\lambda_3)} b_2 b_3 b_4 \lambda_1 \lambda_2 \lambda_3 N_1(0) \\
& + \frac{1}{s(s+\lambda_2)(s+\lambda_3)} b_3 b_4 \lambda_2 \lambda_3 N_2(0) \\
& + \frac{1}{s(s+\lambda_3)} b_4 \lambda_3 N_3(0) \\
& + N_4(0) \\
& + \frac{1}{s^2(s+\lambda_1)(s+\lambda_2)(s+\lambda_3)} b_2 b_3 b_4 \lambda_1 \lambda_2 \lambda_3 \omega_1 \\
& + \frac{1}{s^2(s+\lambda_2)(s+\lambda_3)} b_3 b_4 \lambda_2 \lambda_3 \omega_2 \\
& + \frac{1}{s^2(s+\lambda_3)} b_4 \lambda_3 \omega_3 \\
& + \frac{1}{s^2} \omega_4
\end{aligned} \tag{5.12}$$

By using partial fraction expansion and standard Laplace Transforms, the set of equations below is used to calculate the amount of the m^{th} isotope in the decay chain, providing the m^{th} isotope is unstable.

$$N_m(t; \vec{\lambda}, \vec{b}, \vec{w}) = \sum_{k=1,m} r(k; \vec{\lambda}, \vec{b}) [f(t; k, m, \vec{\lambda}) N_k(0) + g(t; k, m, \vec{\lambda}) w_k] \tag{5.13}$$

$$r(k, m, \vec{\lambda}) = \begin{cases} \prod_{i=k, m-1} (b_{i+1} \lambda_i), & \text{if } k < m \\ 1, & \text{if } k = m \end{cases} \tag{5.14}$$

$$f(t; k, m, \vec{\lambda}) = (-1)^{m-k} \sum_{i=k, m} \left[\exp(-\lambda_i t) \prod_{j=k, m; j \neq i} \left(\frac{1}{\lambda_i - \lambda_j} \right) \right] \tag{5.15}$$

$$g(t; k, m, \vec{\lambda}) = \frac{1}{\prod_{i=k, m} \lambda_i} + (-1)^{m-k+1} \sum_{i=k, m} \left[\frac{1}{\lambda_i} \exp(-\lambda_i t) \prod_{j=k, m; j \neq i} \left(\frac{1}{\lambda_i - \lambda_j} \right) \right] \tag{5.16}$$

The set of equations below is used to calculate the amount of the m^{th} isotope in the decay chain, where the m^{th} isotope is stable.

$$N_m(t; \vec{\lambda}, \vec{b}, \vec{w}) = N_m + w_m t + \sum_{k=1, m-1} r(k; \vec{\lambda}, \vec{b}) [f(t; k, m-1, \vec{\lambda}) N_k(0) + g(t; k, m, \vec{\lambda}) w_k] \tag{5.17}$$

$$r(k, m, \vec{\lambda}) = \begin{cases} \prod_{i=k, m-1} (b_{i+1} \lambda_i), & \text{if } k < m \\ 1, & \text{if } k = m \end{cases} \tag{5.18}$$

$$f(t; k, m, \vec{\lambda}) = \frac{1}{\prod_{i=k,m} \lambda_i} + (-1)^{m-k+1} \sum_{i=k,m} \left[\frac{1}{\lambda_i} \exp(-\lambda_i t) \prod_{j=k,m; j \neq i} \left(\frac{1}{\lambda_i - \lambda_j} \right) \right] \quad (5.19)$$

$$g(t; k, m, \vec{\lambda}) = \frac{1}{\prod_{i=k,m} \lambda_i} t + \frac{\sum_{i=k,m} \left[\prod_{j=k,m; j \neq i} \lambda_j \right]}{\prod_{i=k,m} \lambda_i^2} + (-1)^{m-k+1} \sum_{i=k,m} \left[\frac{1}{\lambda_i^2} \exp(-\lambda_i t) \prod_{j=k,m; j \neq i} \left(\frac{1}{\lambda_i - \lambda_j} \right) \right] \quad (5.20)$$

5.1.5 Preference: Analytic over Numeric

The numeric solution only requires the equation to be solved in the s-domain; the Gaver-Stehfest algorithm performs the inversion. It is worth the extra effort to derive and implement an analytic solution, as the numeric is only an approximation. Examples of the pitfalls of the numeric solution are that it can give negative amounts of an isotope and the difference between the numeric and analytic calculated amounts can become quite large when the isotope decays away to a very small value. Figure 5.2 shows the predicted decay of a sample of Po-218 irradiated for 1,000s, and sampled until 10,000s. In the region between 4,000s and 9,000s the amount from the numeric calculation drops below zero, whereas the analytic calculation remains above zero, as would be expected.

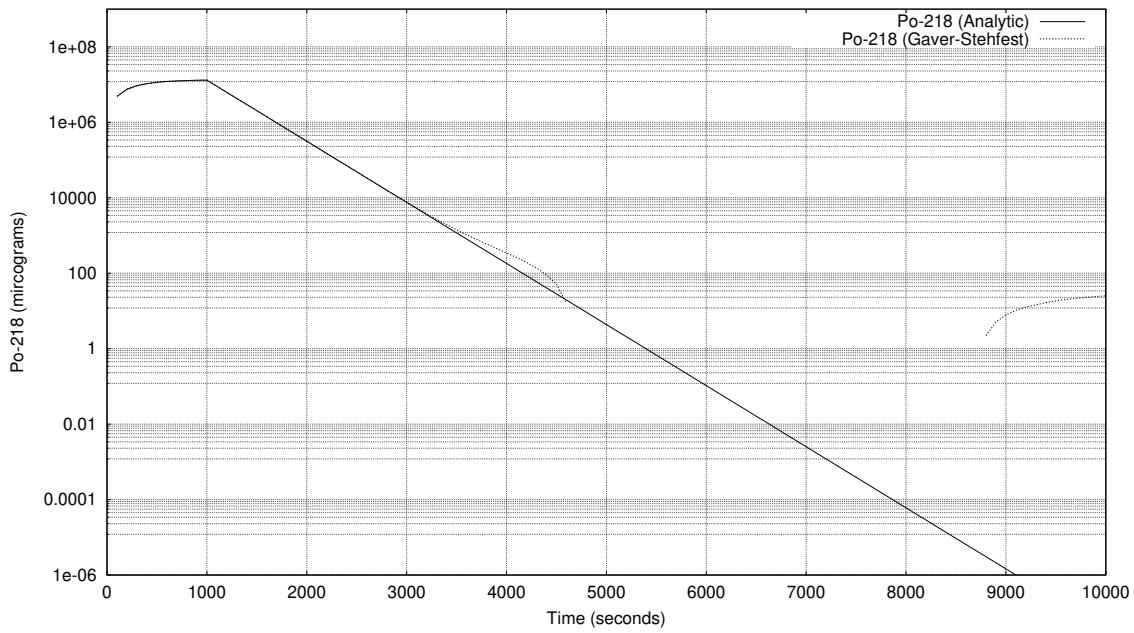


Figure 5.2: Decay of Po-218: Analytic and Gaver-Stehfest Calculations [jeff311]

5.2 Python Isotopes Class

The decay equations are computed within the isotopes class in the isotopes.py file.

```

1
2 class isotopes:
3
4 ######
5 # DECAY EQUATIONS
6 #####
7
8 @staticmethod
9 def calculate_activity(t, l, b, w, n0):
10    nt = numpy.zeros((len(n0),))
11    for m in range(0,len(n0)):
12        if(l[m] > 0.0):
13            nt[m] = isotopes.activity_unstable(t, l, b, w, n0, m)
14        elif(l[m] == 0.0):
15            nt[m] = isotopes.activity_stable(t, l, b, w, n0, m)
16    return nt
17
18 @staticmethod
19 def activity_unstable(t, l, b, w, n0, m):
20    s = 0.0
21    for k in range(0, m+1):
22        s = s + isotopes.r(k, m, b, l) * ( isotopes.f_unstable(t,k,m,l) * n0[k] + isotopes.g_unstable(t,
23                                     k,m,l) * w[k])
24    return s
25
26 @staticmethod
27 def activity_stable(t, l, b, w, n0, m):
28    s = n0[m] + w[m] * t

```

```

28     for k in range(0, m):
29         s = s + isotopes.r(k, m, b, l) * (isotopes.f_stable(t,k,m,l) * n0[k] + isotopes.g_stable(t,k,m,l)
30             ) * w[k])
31
32     @staticmethod
33     def r(k, m, b, l):
34         if(k == m):
35             return 1.0
36         else:
37             p = 1.0
38             for i in range(k, m):
39                 p = p * (b[i] * l[i])
40
41             return p
42
43     @staticmethod
44     def f_unstable(t,k,m,l):
45         s = 0.0
46         for i in range(k, m+1):
47             p = 1.0
48             for j in range(k, m+1):
49                 if(i != j):
50                     p = p * (1 / (l[i] - l[j]))
51
52             s = s + numpy.exp(-1 * l[i] * t) * p
53
54             s = (-1)**(m-k) * s
55
56             return s
57
58     @staticmethod
59     def g_unstable(t,k,m,l):
60         pa = 1.0
61         for i in range(k,m+1):
62             pa = pa * l[i]
63
64             pa = 1.0 / pa
65
66             s = 0.0
67             for i in range(k, m+1):
68                 pb = 1.0
69                 for j in range(k, m+1):
70                     if(i != j):
71                         pb = pb * (1 / (l[i]-l[j]))
72
73                     s = s + (1/l[i]) * numpy.exp(-l[i]*t) * pb
74
75             return pa + s * (-1)**(m-k+1)
76
77
78     @staticmethod
79     def f_stable(t,k,m_in,l):
80         m = m_in - 1
81
82         p = 1.0
83         for i in range(k, m+1):
84             p = p * l[i]
85
86         s = 0.0
87         for i in range(k, m+1):
88

```

```

80     r = l[i]
81     for j in range(k, m+1):
82         if(i != j):
83             r = r * (l[i] - l[j])
84         s = s + (1/r)*numpy.exp(-1*l[i]*t)
85
86     return (1.0/p) + s * (-1.0)**(m-k+1)
87
88
89 @staticmethod
90 def g_stable(t,k,m_in,l):
91     m = m_in - 1
92
93     pa = 1.0
94     for i in range(k,m+1):
95         pa = pa * l[i]
96     pa = 1.0 / pa
97
98     sa = 0.0
99     for i in range(k, m+1):
100        pb = 1.0
101        for j in range(k,m+1):
102            if(j != i):
103                pb = pb * l[j]
104            sa = sa + pb
105        pc = 1.0
106        for i in range(k, m+1):
107            pc = pc * l[i]**2
108
109        sb = 0.0
110        for i in range(k, m+1):
111            pd = 1.0
112            for j in range(k, m+1):
113                if(i != j):
114                    pd = pd * (1 / (l[i]-l[j]))
115                sb = sb + (1/(l[i]**2)) * numpy.exp(-l[i]*t) * pd
116
117    return pa * t + sa / pc + sb * (-1)**(m-k+1)

```

Chapter 6

Future Plans

6.1 Selection of Cross Section Databases

The Scanditronix MC40 Cyclotron at the University of Birmingham has several beamlines and is capable of accelerating protons, deuterons, Helium 3 and Helium 4 and fluxes and energy ranges detailed below.

Particle	Energy (MeV)	Max Current (micro A)	Flux (ions per second)
p	8-40	60	3.75×10^{14}
d	8-40	30	1.87×10^{14}
$^4He^{2+}$	8-53	30	9.36×10^{13}
$^3He^{2+}$	4-20	60	1.87×10^{14}

Table 6.1: Beam Characteristics of the Scanditronix MC-40

The current data file is for Protons, and the range of energies (for Fe-54 and Fe-56 at the very least) only range up to 30MeV. Previous versions of the TENDL data files have covered larger ranges (TENDL-2009 extended up to 200MeV).

As the energies do not cover the full range of possible energies for our own cyclotron, it would be desirable to use the TALYS program to calculate the reaction cross sections for a range up to at least 100MeV (for Protons, Deuterons, Helium 3 and Helium 4 ions) and save this cross section data into an alternate database.

Bibliography

- [1] A.J. Koning et al. “TENDL: Complete Nuclear Data Library for Innovative Nuclear Science and Technology”. In: *Nuclear Data Sheets* 155 (2019).
- [2] A.J.M. Plompen et al. “The JEFF-3.3 Nuclear Data Library”. In: *Eur. Phys. J. A* 56 (181 2020).
- [3] J. P. Biersack James F. Ziegler M. D. Ziegler. “SRIM - The stopping and range of ions in matter”. In: *Nuclear Instruments and Methods in Physics Research Section B* 268 (2010), pp. 1818–1823.
- [4] H. Stehfest. “Algorithm 368: Numerical Inversion of Laplace Transform”. In: *Communications of the ACM* 13 (1970), pp. 47–49.

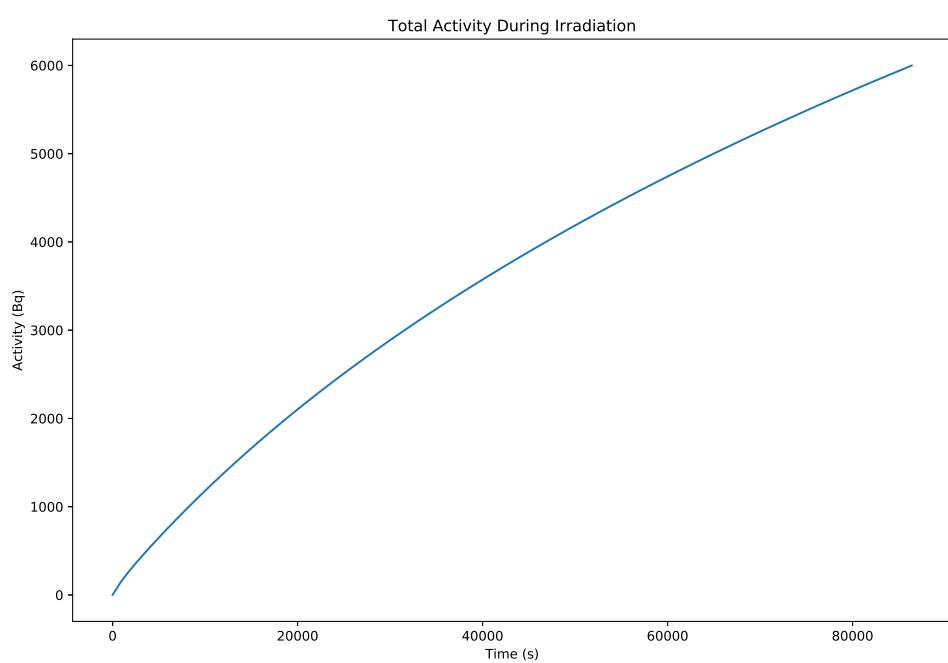


Figure D.5: Gradual saturation of steel with radioactive isotopes under proton irradiation

Appendix E

Activity V1 Full Results for Iron

Output file produced by Activity V1: <https://github.com/BenPalmer1983/activity>

Listing E.1: Isotope Activity at End of Simulation

```
1 =====
2 ACTIVITY code University of Birmingham
3 =====
4
5 Compiled: 15:08:41 11/01/2016
6 MPI Processes: 1
7
8 Date: 21:21 24/06/2016
9 Read user input          0.0000
10 Read Isotope Data        0.0000
11 Read Activity User Input File    0.0720
12 Read Trajectory File       0.0720
13 Data point count: 100887   0.9560
14
15 XS Data Files Loaded:
16 Reading xs file 1_1-26_54_0.dat
17 Reading xs file 1_1-26_56_0.dat
18 Reading xs file 1_1-26_57_0.dat
19 Reading xs file 1_1-26_58_0.dat
20
21 Calculation Settings
22
23 Beam Duration/s:        0.3000000000D+03
24 Beam Current/uA:         0.5000000000D+00
25 Beam Energy/MeV:         0.3600000000D+02
26 Beam Area/mm2:          0.1000000000D+03
27 Target Thickness/Angstrom: 0.5000000000D+07
28 Target Density/kgm-3:     0.8000000000D+04
29 Activity Measurement Time/s: 0.2600000000D+06
30 Input File:              /home/ben/activity/examples/Fe36MeV.in
31 Isotope File:             /home/ben/activity/data/isotopes.txt
32 Decay Modes File:        /home/ben/activity/data/decaymodes.txt
33 Gamma Energies File:     /home/ben/activity/data/gammaenergies.txt
34 XS File Directory:       /home/ben/activity/data/xs
35
36
37
38 Create material tally      1.2640
39 Fill tally with blank data 1.2640
40 Fill tally with isotope data 1.2640
41
```

```

42 -----
43 Starting Isotope Tally - Input Material
44 Element Z A M mk sim Half life Decay Constant Activity Reaction Rate Atoms/mg
45 -----
46     FE 026 054 0 000 001 -0.1000000000D+01 0.0000000000D+00 0.0000000000D+00 0.0000000000D+00 0.5405383878D+28
47     FE 026 056 0 000 001 -0.1000000000D+01 0.0000000000D+00 0.0000000000D+00 0.0000000000D+00 0.7890272999D+29
48     FE 026 057 0 000 001 -0.1000000000D+01 0.0000000000D+00 0.0000000000D+00 0.0000000000D+00 0.1758807969D+28
49     FE 026 058 0 000 001 -0.1000000000D+01 0.0000000000D+00 0.0000000000D+00 0.0000000000D+00 0.2263810953D+27
50 Total starting isotopes: 4
51
52
53 -----
54 Pre-Simulation Summary
55 1.2840
56 -----
57
58 Beam flux ions/s: 0.3120754681E+13
59
60
61 Calculate projectile-target-product reaction rates 1.2840
62 Reaction rates to calculate: 424
63 Trajectory fit equation:
64     0 0.3665497627E+05
65     1 -0.1783622595E-02
66     2 0.2947794093E-09
67     3 -0.3735372827E-16
68     4 0.1919778602E-23
69     5 -0.3566818320E-31
70 Affected material depth: 0.5000000000E+07
71
72 -----
73 Calculated variables
74 -----
75 Beam flux ions/s: 0.3120754681E+13
76 Affected depth/m: 0.5000000000E-03
77 Affected depth/ang: 0.5000000000E+07
78 Affected volume/m3: 0.500000054E-07
79 Number Density: 0.8626809746E+29
80 VPI: 0.6020000000E+02
81 DPA: 0.1306643605E-04
82
83 -----
84 Reaction rates in the target material 1.2840
85      Sym Z A M      Sym Z A M Reaction Rate Average XS Number Density
86 -----
87     1: FE 26 54 0 NN 0 1 0 0.3508066664E+09 0.4160433001E-28 0.5403805016E+28
88     2: FE 26 54 0 H 1 1 0 0.1351444089E+10 0.1602761044E-27 0.5403805016E+28
89     3: FE 26 54 0 H 1 2 0 0.3205323639E+08 0.3801391343E-29 0.5403805016E+28
90     4: FE 26 54 0 H 1 3 0 0.7165361660E+06 0.8497845101E-31 0.5403805016E+28
91     5: FE 26 54 0 HE 2 3 0 0.8617826538E+07 0.1022041294E-29 0.5403805016E+28
92     6: FE 26 54 0 HE 2 4 0 0.5675883691E+08 0.6731381151E-29 0.5403805016E+28
93     67: FE 26 54 0 TI 22 46 0 0.4687831440E+01 0.5559588940E-36 0.5403805016E+28
94     76: FE 26 54 0 V 23 47 0 0.3452942314E+06 0.4095057628E-31 0.5403805016E+28
95     78: FE 26 54 0 V 23 49 0 0.5791127831E+07 0.6868056297E-30 0.5403805016E+28
96     85: FE 26 54 0 CR 24 49 0 0.2416421986E+07 0.2865784131E-30 0.5403805016E+28
97     86: FE 26 54 0 CR 24 50 0 0.4290008294E+08 0.5087785892E-29 0.5403805016E+28
98     87: FE 26 54 0 CR 24 51 0 0.2135062196E+07 0.2532102172E-30 0.5403805016E+28
99     88: FE 26 54 0 CR 24 52 0 0.8659953579E+08 0.1027037400E-28 0.5403805016E+28
100    92: FE 26 54 0 MN 25 50 0 0.1542445948E+07 0.1829281950E-30 0.5403805016E+28

```

101	93:	FE	26	54	0	MN	25	51	0	0.1778372502E+07	0.2109081827E-30	0.5403805016E+28
102	94:	FE	26	54	0	MN	25	52	0	0.9863724488E+08	0.1169800030E-28	0.5403805016E+28
103	95:	FE	26	54	0	MN	25	53	0	0.2232277074E+09	0.2647395303E-28	0.5403805016E+28
104	99:	FE	26	54	0	FE	26	52	0	0.1143967233E+08	0.1356701421E-29	0.5403805016E+28
105	100:	FE	26	54	0	FE	26	53	0	0.1888610876E+09	0.2239820325E-28	0.5403805016E+28
106	101:	FE	26	54	0	FE	26	54	0	0.1103802931E+09	0.1309068094E-28	0.5403805016E+28
107	104:	FE	26	54	0	CO	27	53	0	0.4251641576E+06	0.5042284431E-31	0.5403805016E+28
108	105:	FE	26	54	0	CO	27	54	0	0.8026455518E+07	0.9519069510E-30	0.5403805016E+28
109	106:	FE	26	54	0	CO	27	55	0	0.1175568630E+06	0.1394179471E-31	0.5403805016E+28
110	107:	FE	26	56	0	NN	0	1	0	0.1294835964E+11	0.1052010224E-27	0.7887968323E+29
111	108:	FE	26	56	0	H	1	1	0	0.1322708565E+11	0.1074655765E-27	0.7887968323E+29
112	109:	FE	26	56	0	H	1	2	0	0.6485018658E+09	0.5268857307E-29	0.7887968323E+29
113	110:	FE	26	56	0	H	1	3	0	0.2804930290E+08	0.2278910553E-30	0.7887968323E+29
114	111:	FE	26	56	0	HE	2	3	0	0.1152252073E+09	0.9361656570E-30	0.7887968323E+29
115	112:	FE	26	56	0	HE	2	4	0	0.8396082900E+09	0.6821532069E-29	0.7887968323E+29
116	169:	FE	26	56	0	TI	22	48	0	0.7417367941E+02	0.6026359420E-36	0.7887968323E+29
117	179:	FE	26	56	0	V	23	49	0	0.1002075285E+08	0.8141521197E-31	0.7887968323E+29
118	181:	FE	26	56	0	V	23	51	0	0.3129496674E+07	0.2542609711E-31	0.7887968323E+29
119	188:	FE	26	56	0	CR	24	50	0	0.3047989029E+03	0.2476387520E-35	0.7887968323E+29
120	189:	FE	26	56	0	CR	24	51	0	0.1903672236E+09	0.1546669008E-29	0.7887968323E+29
121	190:	FE	26	56	0	CR	24	52	0	0.2832038545E+09	0.2300935089E-29	0.7887968323E+29
122	191:	FE	26	56	0	CR	24	53	0	0.1158364999E+07	0.9411322022E-32	0.7887968323E+29
123	192:	FE	26	56	0	CR	24	54	0	0.1165756880E+08	0.9471378543E-31	0.7887968323E+29
124	196:	FE	26	56	0	MN	25	51	0	0.9817102039E+01	0.7976061834E-37	0.7887968323E+29
125	197:	FE	26	56	0	MN	25	52	0	0.1545890268E+09	0.1255983315E-29	0.7887968323E+29
126	198:	FE	26	56	0	MN	25	53	0	0.8815593471E+08	0.7162370153E-30	0.7887968323E+29
127	199:	FE	26	56	0	MN	25	54	0	0.1267995071E+10	0.1030202911E-28	0.7887968323E+29
128	200:	FE	26	56	0	MN	25	55	0	0.7253812202E+09	0.5893475939E-29	0.7887968323E+29
129	204:	FE	26	56	0	FE	26	53	0	0.2156782879E+02	0.1752312805E-36	0.7887968323E+29
130	205:	FE	26	56	0	FE	26	54	0	0.2975937641E+10	0.2417848215E-28	0.7887968323E+29
131	206:	FE	26	56	0	FE	26	55	0	0.4948211136E+10	0.4020253414E-28	0.7887968323E+29
132	207:	FE	26	56	0	FE	26	56	0	0.1551234326E+10	0.1260325181E-28	0.7887968323E+29
133	210:	FE	26	56	0	CO	27	55	0	0.3083905589E+09	0.2505568504E-29	0.7887968323E+29
134	211:	FE	26	56	0	CO	27	56	0	0.2971765420E+09	0.2414458427E-29	0.7887968323E+29
135	212:	FE	26	56	0	CO	27	57	0	0.3041404939E+07	0.2471038170E-31	0.7887968323E+29
136	213:	FE	26	57	0	NN	0	1	0	0.3649443293E+09	0.1330164742E-27	0.1758294237E+28
137	214:	FE	26	57	0	H	1	1	0	0.2670335825E+09	0.9732954535E-28	0.1758294237E+28
138	215:	FE	26	57	0	H	1	2	0	0.1672940269E+08	0.6097604437E-29	0.1758294237E+28
139	216:	FE	26	57	0	H	1	3	0	0.12754404383E+07	0.4648779807E-30	0.1758294237E+28
140	217:	FE	26	57	0	HE	2	3	0	0.2836366558E+07	0.1033811047E-29	0.1758294237E+28
141	218:	FE	26	57	0	HE	2	4	0	0.2332170710E+08	0.8500395818E-29	0.1758294237E+28
142	273:	FE	26	57	0	TI	22	49	0	0.7752833017E+00	0.2825785826E-36	0.1758294237E+28
143	282:	FE	26	57	0	V	23	49	0	0.3281858226E+02	0.1196185760E-34	0.1758294237E+28
144	283:	FE	26	57	0	V	23	50	0	0.1835618293E+06	0.6690540276E-31	0.1758294237E+28
145	284:	FE	26	57	0	V	23	51	0	0.6679861824E+00	0.2434704685E-36	0.1758294237E+28
146	285:	FE	26	57	0	V	23	52	0	0.3050010636E+01	0.1111680957E-35	0.1758294237E+28
147	293:	FE	26	57	0	CR	24	51	0	0.3269383508E+04	0.1191638922E-32	0.1758294237E+28
148	294:	FE	26	57	0	CR	24	52	0	0.7230173808E+07	0.2635284756E-29	0.1758294237E+28
149	295:	FE	26	57	0	CR	24	53	0	0.1653080211E+07	0.6025217646E-30	0.1758294237E+28
150	296:	FE	26	57	0	CR	24	54	0	0.4518951495E+05	0.1647086821E-31	0.1758294237E+28
151	297:	FE	26	57	0	CR	24	55	0	0.7219801309E+04	0.2631504144E-32	0.1758294237E+28
152	301:	FE	26	57	0	MN	25	52	0	0.2058316554E+07	0.7502240448E-30	0.1758294237E+28
153	302:	FE	26	57	0	MN	25	53	0	0.9698739328E+07	0.3535038104E-29	0.1758294237E+28
154	303:	FE	26	57	0	MN	25	54	0	0.2412989059E+07	0.8794966005E-30	0.1758294237E+28
155	304:	FE	26	57	0	MN	25	55	0	0.1977898329E+08	0.7209128654E-29	0.1758294237E+28
156	305:	FE	26	57	0	MN	25	56	0	0.5440385638E+07	0.1982935089E-29	0.1758294237E+28
157	309:	FE	26	57	0	FE	26	54	0	0.2296404190E+07	0.8370032473E-30	0.1758294237E+28
158	310:	FE	26	57	0	FE	26	55	0	0.9449535161E+08	0.3444207100E-28	0.1758294237E+28
159	311:	FE	26	57	0	FE	26	56	0	0.1023020929E+09	0.3728750553E-28	0.1758294237E+28
160	312:	FE	26	57	0	FE	26	57	0	0.2808424683E+08	0.1023626672E-28	0.1758294237E+28
161	315:	FE	26	57	0	CO	27	55	0	0.4515569494E+07	0.1645854134E-29	0.1758294237E+28

```

162 316: FE 26 57 0 CO 27 56 0 0.1114292806E+08 0.4061422208E-29 0.1758294237E+28
163 317: FE 26 57 0 CO 27 57 0 0.5878105202E+07 0.2142476992E-29 0.1758294237E+28
164 318: FE 26 57 0 CO 27 58 0 0.3965181061E+05 0.1445246198E-31 0.1758294237E+28
165 319: FE 26 58 0 NN 0 1 0 0.5793566623E+08 0.1640601429E-27 0.2263149715E+27
166 320: FE 26 58 0 H 1 1 0 0.2791132972E+08 0.7903830301E-28 0.2263149715E+27
167 321: FE 26 58 0 H 1 2 0 0.2037087602E+07 0.5768551650E-29 0.2263149715E+27
168 322: FE 26 58 0 H 1 3 0 0.1593981581E+06 0.4513779906E-30 0.2263149715E+27
169 323: FE 26 58 0 HE 2 3 0 0.2598326212E+06 0.7357847030E-30 0.2263149715E+27
170 324: FE 26 58 0 HE 2 4 0 0.2551528049E+07 0.7225325669E-29 0.2263149715E+27
171 376: FE 26 58 0 TI 22 50 0 0.9741805634E+00 0.2758649600E-35 0.2263149715E+27
172 385: FE 26 58 0 V 23 50 0 0.2503943210E+00 0.7090576626E-36 0.2263149715E+27
173 386: FE 26 58 0 V 23 51 0 0.9310532844E+04 0.2636523317E-31 0.2263149715E+27
174 388: FE 26 58 0 V 23 53 0 0.1852730029E-06 0.5246494484E-42 0.2263149715E+27
175 396: FE 26 58 0 CR 24 52 0 0.4688542450E+04 0.1327684645E-31 0.2263149715E+27
176 397: FE 26 58 0 CR 24 53 0 0.1090733133E+06 0.3088698988E-30 0.2263149715E+27
177 398: FE 26 58 0 CR 24 54 0 0.7904949881E+05 0.2238495372E-30 0.2263149715E+27
178 399: FE 26 58 0 CR 24 55 0 0.5894353392E-05 0.1669141865E-40 0.2263149715E+27
179 400: FE 26 58 0 CR 24 56 0 0.3341354293E-04 0.9461927315E-40 0.2263149715E+27
180 405: FE 26 58 0 MN 25 53 0 0.1200142917E+07 0.3398521693E-29 0.2263149715E+27
181 406: FE 26 58 0 MN 25 54 0 0.1053914972E+07 0.2984438639E-29 0.2263149715E+27
182 407: FE 26 58 0 MN 25 55 0 0.2219588486E+06 0.6285351112E-30 0.2263149715E+27
183 408: FE 26 58 0 MN 25 56 0 0.4947583955E+06 0.1401039090E-29 0.2263149715E+27
184 409: FE 26 58 0 MN 25 57 0 0.3707982841E+06 0.1050013290E-29 0.2263149715E+27
185 414: FE 26 58 0 FE 26 55 0 0.4036879251E+06 0.1143148996E-29 0.2263149715E+27
186 415: FE 26 58 0 FE 26 56 0 0.1504003971E+08 0.4258984535E-28 0.2263149715E+27
187 416: FE 26 58 0 FE 26 57 0 0.8926880873E+07 0.2527882127E-28 0.2263149715E+27
188 417: FE 26 58 0 FE 26 58 0 0.3876895164E+07 0.1097845276E-28 0.2263149715E+27
189 421: FE 26 58 0 CO 27 56 0 0.2685676346E+07 0.7605202007E-29 0.2263149715E+27
190 422: FE 26 58 0 CO 27 57 0 0.3377108415E+07 0.9563174557E-29 0.2263149715E+27
191 423: FE 26 58 0 CO 27 58 0 0.7015543919E+06 0.1986636580E-29 0.2263149715E+27
192 424: FE 26 58 0 CO 27 59 0 0.7735041844E+04 0.2190381423E-31 0.2263149715E+27
193
194
195 Prepare full isotope tally 1.2840
196
197 -----

```

```

198          Full Starting Isotope Tally
199 Key   Element Z A M mk sim Half life      Activity      Reaction Rate      Atoms
200 -----

```

```

201 000001  NN 000 001 0 000 001 -0.1000000000D+01 0.0000000000D+00 0.0000000000D+00 0.0000000000D+00
202 000591  H 001 001 0 000 001 -0.1000000000D+01 0.0000000000D+00 0.0000000000D+00 0.0000000000D+00
203 000592  H 001 002 0 000 001 -0.1000000000D+01 0.0000000000D+00 0.0000000000D+00 0.0000000000D+00
204 000593  H 001 003 0 000 001 0.3885235200D+09 0.0000000000D+00 0.0000000000D+00 0.0000000000D+00
205 001183  HE 002 003 0 000 001 -0.1000000000D+01 0.0000000000D+00 0.0000000000D+00 0.0000000000D+00
206 001184  HE 002 004 0 000 001 -0.1000000000D+01 0.0000000000D+00 0.0000000000D+00 0.0000000000D+00
207 008881  P 015 031 0 000 001 -0.1000000000D+01 0.0000000000D+00 0.0000000000D+00 0.0000000000D+00
208 009472  S 016 032 0 000 001 -0.1000000000D+01 0.0000000000D+00 0.0000000000D+00 0.0000000000D+00
209 009473  S 016 033 0 000 001 -0.1000000000D+01 0.0000000000D+00 0.0000000000D+00 0.0000000000D+00
210 009474  S 016 034 0 000 001 -0.1000000000D+01 0.0000000000D+00 0.0000000000D+00 0.0000000000D+00
211 009476  S 016 036 0 000 001 -0.1000000000D+01 0.0000000000D+00 0.0000000000D+00 0.0000000000D+00
212 010062  CL 017 032 0 000 001 0.2980000000D+00 0.0000000000D+00 0.0000000000D+00 0.0000000000D+00
213 010063  CL 017 033 0 000 001 0.2511000000D+01 0.0000000000D+00 0.0000000000D+00 0.0000000000D+00
214 010064  CL 017 034 0 000 001 0.1526400000D+01 0.0000000000D+00 0.0000000000D+00 0.0000000000D+00
215 010065  CL 017 035 0 000 001 -0.1000000000D+01 0.0000000000D+00 0.0000000000D+00 0.0000000000D+00
216 010066  CL 017 036 0 000 001 0.9492336000D+13 0.0000000000D+00 0.0000000000D+00 0.0000000000D+00
217 010067  CL 017 037 0 000 001 -0.1000000000D+01 0.0000000000D+00 0.0000000000D+00 0.0000000000D+00
218 010068  CL 017 038 0 000 001 0.2234400000D+04 0.0000000000D+00 0.0000000000D+00 0.0000000000D+00
219 010069  CL 017 039 0 000 001 0.3372000000D+04 0.0000000000D+00 0.0000000000D+00 0.0000000000D+00
220 010070  CL 017 040 0 000 001 0.8100000000D+02 0.0000000000D+00 0.0000000000D+00 0.0000000000D+00

```

221 010071 CL 017 041 0 000 001 0.3840000000D+02 0.0000000000D+00 0.0000000000D+00 0.0000000000D+00
 222 010653 AR 018 033 0 000 001 0.1730000000D+00 0.0000000000D+00 0.0000000000D+00 0.0000000000D+00
 223 010654 AR 018 034 0 000 001 0.8445000000D+00 0.0000000000D+00 0.0000000000D+00 0.0000000000D+00
 224 010655 AR 018 035 0 000 001 0.1775000000D+01 0.0000000000D+00 0.0000000000D+00 0.0000000000D+00
 225 010656 AR 018 036 0 000 001 -0.1000000000D+01 0.0000000000D+00 0.0000000000D+00 0.0000000000D+00
 226 010657 AR 018 037 0 000 001 0.3027456000D+07 0.0000000000D+00 0.0000000000D+00 0.0000000000D+00
 227 010658 AR 018 038 0 000 001 -0.1000000000D+01 0.0000000000D+00 0.0000000000D+00 0.0000000000D+00
 228 010659 AR 018 039 0 000 001 0.8483184000D+10 0.0000000000D+00 0.0000000000D+00 0.0000000000D+00
 229 010660 AR 018 040 0 000 001 -0.1000000000D+01 0.0000000000D+00 0.0000000000D+00 0.0000000000D+00
 230 010661 AR 018 041 0 000 001 0.6576600000D+04 0.0000000000D+00 0.0000000000D+00 0.0000000000D+00
 231 010662 AR 018 042 0 000 001 0.1037534400D+10 0.0000000000D+00 0.0000000000D+00 0.0000000000D+00
 232 010663 AR 018 043 0 000 001 0.3222000000D+03 0.0000000000D+00 0.0000000000D+00 0.0000000000D+00
 233 011245 K 019 035 0 000 001 0.1780000000D+00 0.0000000000D+00 0.0000000000D+00 0.0000000000D+00
 234 011246 K 019 036 0 000 001 0.3420000000D+00 0.0000000000D+00 0.0000000000D+00 0.0000000000D+00
 235 011247 K 019 037 0 000 001 0.1226000000D+01 0.0000000000D+00 0.0000000000D+00 0.0000000000D+00
 236 011248 K 019 038 0 000 001 0.4581600000D+03 0.0000000000D+00 0.0000000000D+00 0.0000000000D+00
 237 011249 K 019 039 0 000 001 -0.1000000000D+01 0.0000000000D+00 0.0000000000D+00 0.0000000000D+00
 238 011250 K 019 040 0 000 001 0.3935692800D+17 0.0000000000D+00 0.0000000000D+00 0.0000000000D+00
 239 011251 K 019 041 0 000 001 -0.1000000000D+01 0.0000000000D+00 0.0000000000D+00 0.0000000000D+00
 240 011252 K 019 042 0 000 001 0.4449600000D+05 0.0000000000D+00 0.0000000000D+00 0.0000000000D+00
 241 011253 K 019 043 0 000 001 0.8028000000D+05 0.0000000000D+00 0.0000000000D+00 0.0000000000D+00
 242 011254 K 019 044 0 000 001 0.1327800000D+04 0.0000000000D+00 0.0000000000D+00 0.0000000000D+00
 243 011255 K 019 045 0 000 001 0.1038000000D+04 0.0000000000D+00 0.0000000000D+00 0.0000000000D+00
 244 011256 K 019 046 0 000 001 0.1050000000D+03 0.0000000000D+00 0.0000000000D+00 0.0000000000D+00
 245 011837 CA 020 037 0 000 001 0.1811000000D+00 0.0000000000D+00 0.0000000000D+00 0.0000000000D+00
 246 011838 CA 020 038 0 000 001 0.4400000000D+00 0.0000000000D+00 0.0000000000D+00 0.0000000000D+00
 247 011839 CA 020 039 0 000 001 0.8596000000D+00 0.0000000000D+00 0.0000000000D+00 0.0000000000D+00
 248 011840 CA 020 040 0 000 001 -0.1000000000D+01 0.0000000000D+00 0.0000000000D+00 0.0000000000D+00
 249 011841 CA 020 041 0 000 001 0.3216672000D+13 0.0000000000D+00 0.0000000000D+00 0.0000000000D+00
 250 011842 CA 020 042 0 000 001 -0.1000000000D+01 0.0000000000D+00 0.0000000000D+00 0.0000000000D+00
 251 011843 CA 020 043 0 000 001 -0.1000000000D+01 0.0000000000D+00 0.0000000000D+00 0.0000000000D+00
 252 011844 CA 020 044 0 000 001 -0.1000000000D+01 0.0000000000D+00 0.0000000000D+00 0.0000000000D+00
 253 011845 CA 020 045 0 000 001 0.1404950400D+08 0.0000000000D+00 0.0000000000D+00 0.0000000000D+00
 254 011846 CA 020 046 0 000 001 -0.1000000000D+01 0.0000000000D+00 0.0000000000D+00 0.0000000000D+00
 255 011847 CA 020 047 0 000 001 0.3919104000D+06 0.0000000000D+00 0.0000000000D+00 0.0000000000D+00
 256 011848 CA 020 048 0 000 001 -0.1000000000D+01 0.0000000000D+00 0.0000000000D+00 0.0000000000D+00
 257 012429 SC 021 039 0 000 001 0.3000000000D-06 0.0000000000D+00 0.0000000000D+00 0.0000000000D+00
 258 012430 SC 021 040 0 000 001 0.1823000000D+00 0.0000000000D+00 0.0000000000D+00 0.0000000000D+00
 259 012431 SC 021 041 0 000 001 0.5963000000D+00 0.0000000000D+00 0.0000000000D+00 0.0000000000D+00
 260 012432 SC 021 042 0 000 001 0.6813000000D+00 0.0000000000D+00 0.0000000000D+00 0.0000000000D+00
 261 012433 SC 021 043 0 000 001 0.1400760000D+05 0.0000000000D+00 0.0000000000D+00 0.0000000000D+00
 262 012434 SC 021 044 0 000 001 0.1429200000D+05 0.0000000000D+00 0.0000000000D+00 0.0000000000D+00
 263 012435 SC 021 045 0 000 001 -0.1000000000D+01 0.0000000000D+00 0.0000000000D+00 0.0000000000D+00
 264 012436 SC 021 046 0 000 001 0.7239456000D+07 0.0000000000D+00 0.0000000000D+00 0.0000000000D+00
 265 012437 SC 021 047 0 000 001 0.2893708800D+06 0.0000000000D+00 0.0000000000D+00 0.0000000000D+00
 266 012438 SC 021 048 0 000 001 0.1572120000D+06 0.0000000000D+00 0.0000000000D+00 0.0000000000D+00
 267 012439 SC 021 049 0 000 001 0.3432000000D+04 0.0000000000D+00 0.0000000000D+00 0.0000000000D+00
 268 012440 SC 021 050 0 000 001 0.1025000000D+03 0.0000000000D+00 0.0000000000D+00 0.0000000000D+00
 269 013021 TI 022 041 0 000 001 0.8040000000D-01 0.0000000000D+00 0.0000000000D+00 0.0000000000D+00
 270 013022 TI 022 042 0 000 001 0.1990000000D+00 0.0000000000D+00 0.0000000000D+00 0.0000000000D+00
 271 013023 TI 022 043 0 000 001 0.5090000000D+00 0.0000000000D+00 0.0000000000D+00 0.0000000000D+00
 272 013024 TI 022 044 0 000 001 0.1892160000D+10 0.0000000000D+00 0.0000000000D+00 0.0000000000D+00
 273 013025 TI 022 045 0 000 001 0.1108800000D+05 0.0000000000D+00 0.0000000000D+00 0.0000000000D+00
 274 013026 TI 022 046 0 000 001 -0.1000000000D+01 0.0000000000D+00 0.0000000000D+00 0.0000000000D+00
 275 013027 TI 022 047 0 000 001 -0.1000000000D+01 0.0000000000D+00 0.0000000000D+00 0.0000000000D+00
 276 013028 TI 022 048 0 000 001 -0.1000000000D+01 0.0000000000D+00 0.0000000000D+00 0.0000000000D+00
 277 013029 TI 022 049 0 000 001 -0.1000000000D+01 0.0000000000D+00 0.0000000000D+00 0.0000000000D+00
 278 013030 TI 022 050 0 000 001 -0.1000000000D+01 0.0000000000D+00 0.0000000000D+00 0.0000000000D+00
 279 013031 TI 022 051 0 000 001 0.3456000000D+03 0.0000000000D+00 0.0000000000D+00 0.0000000000D+00
 280 013032 TI 022 052 0 000 001 0.1020000000D+03 0.0000000000D+00 0.0000000000D+00 0.0000000000D+00
 281 013613 V 023 043 0 000 001 0.8000000000D+00 0.0000000000D+00 0.0000000000D+00 0.0000000000D+00

282 013614 V 023 044 0 000 001 0.1110000000D+00 0.0000000000D+00 0.0000000000D+00 0.0000000000D+00
 283 013615 V 023 045 0 000 001 0.5470000000D+00 0.0000000000D+00 0.0000000000D+00 0.0000000000D+00
 284 013616 V 023 046 0 000 001 0.4225000000D+00 0.0000000000D+00 0.0000000000D+00 0.0000000000D+00
 285 013617 V 023 047 0 000 001 0.1956000000D+04 0.0000000000D+00 0.0000000000D+00 0.0000000000D+00
 286 013618 V 023 048 0 000 001 0.1380110400D+07 0.0000000000D+00 0.0000000000D+00 0.0000000000D+00
 287 013619 V 023 049 0 000 001 0.2851200000D+08 0.0000000000D+00 0.0000000000D+00 0.0000000000D+00
 288 013620 V 023 050 0 000 001 0.4415040000D+25 0.0000000000D+00 0.0000000000D+00 0.0000000000D+00
 289 013621 V 023 051 0 000 001 -0.1000000000D+01 0.0000000000D+00 0.0000000000D+00 0.0000000000D+00
 290 013622 V 023 052 0 000 001 0.2245800000D+03 0.0000000000D+00 0.0000000000D+00 0.0000000000D+00
 291 013623 V 023 053 0 000 001 0.9600000000D+02 0.0000000000D+00 0.0000000000D+00 0.0000000000D+00
 292 013624 V 023 054 0 000 001 0.4980000000D+02 0.0000000000D+00 0.0000000000D+00 0.0000000000D+00
 293 013625 V 023 055 0 000 001 0.6540000000D+01 0.0000000000D+00 0.0000000000D+00 0.0000000000D+00
 294 014205 CR 024 045 0 000 001 0.5000000000D-01 0.0000000000D+00 0.0000000000D+00 0.0000000000D+00
 295 014206 CR 024 046 0 000 001 0.2600000000D+00 0.0000000000D+00 0.0000000000D+00 0.0000000000D+00
 296 014207 CR 024 047 0 000 001 0.5000000000D+00 0.0000000000D+00 0.0000000000D+00 0.0000000000D+00
 297 014208 CR 024 048 0 000 001 0.776160000D+05 0.0000000000D+00 0.0000000000D+00 0.0000000000D+00
 298 014209 CR 024 049 0 000 001 0.253800000D+04 0.0000000000D+00 0.0000000000D+00 0.0000000000D+00
 299 014210 CR 024 050 0 000 001 -0.1000000000D+01 0.0000000000D+00 0.0000000000D+00 0.0000000000D+00
 300 014211 CR 024 051 0 000 001 0.2393496000D+07 0.0000000000D+00 0.0000000000D+00 0.0000000000D+00
 301 014212 CR 024 052 0 000 001 -0.1000000000D+01 0.0000000000D+00 0.0000000000D+00 0.0000000000D+00
 302 014213 CR 024 053 0 000 001 -0.1000000000D+01 0.0000000000D+00 0.0000000000D+00 0.0000000000D+00
 303 014214 CR 024 054 0 000 001 -0.1000000000D+01 0.0000000000D+00 0.0000000000D+00 0.0000000000D+00
 304 014215 CR 024 055 0 000 001 0.2098200000D+03 0.0000000000D+00 0.0000000000D+00 0.0000000000D+00
 305 014216 CR 024 056 0 000 001 0.356400000D+03 0.0000000000D+00 0.0000000000D+00 0.0000000000D+00
 306 014797 MN 025 047 0 000 001 0.1000000000D+00 0.0000000000D+00 0.0000000000D+00 0.0000000000D+00
 307 014798 MN 025 048 0 000 001 0.1581000000D+00 0.0000000000D+00 0.0000000000D+00 0.0000000000D+00
 308 014799 MN 025 049 0 000 001 0.3820000000D+00 0.0000000000D+00 0.0000000000D+00 0.0000000000D+00
 309 014800 MN 025 050 0 000 001 0.2838800000D+00 0.0000000000D+00 0.0000000000D+00 0.0000000000D+00
 310 014801 MN 025 051 0 000 001 0.2772000000D+04 0.0000000000D+00 0.0000000000D+00 0.0000000000D+00
 311 014802 MN 025 052 0 000 001 0.4830624000D+06 0.0000000000D+00 0.0000000000D+00 0.0000000000D+00
 312 014803 MN 025 053 0 000 001 0.1179446400D+15 0.0000000000D+00 0.0000000000D+00 0.0000000000D+00
 313 014804 MN 025 054 0 000 001 0.2696716800D+08 0.0000000000D+00 0.0000000000D+00 0.0000000000D+00
 314 014805 MN 025 055 0 000 001 -0.1000000000D+01 0.0000000000D+00 0.0000000000D+00 0.0000000000D+00
 315 014806 MN 025 056 0 000 001 0.9284040000D+04 0.0000000000D+00 0.0000000000D+00 0.0000000000D+00
 316 014807 MN 025 057 0 000 001 0.8540000000D+02 0.0000000000D+00 0.0000000000D+00 0.0000000000D+00
 317 015389 FE 026 049 0 000 001 0.7000000000D-01 0.0000000000D+00 0.0000000000D+00 0.0000000000D+00
 318 015390 FE 026 050 0 000 001 0.1550000000D+00 0.0000000000D+00 0.0000000000D+00 0.0000000000D+00
 319 015391 FE 026 051 0 000 001 0.3050000000D+00 0.0000000000D+00 0.0000000000D+00 0.0000000000D+00
 320 015392 FE 026 052 0 000 001 0.2979000000D+05 0.0000000000D+00 0.0000000000D+00 0.0000000000D+00
 321 015393 FE 026 053 0 000 001 0.5106000000D+03 0.0000000000D+00 0.0000000000D+00 0.0000000000D+00
 322 015394 FE 026 054 0 000 001 -0.1000000000D+01 0.0000000000D+00 0.0000000000D+00 0.5405383878D+28
 323 015395 FE 026 055 0 000 001 0.8631403200D+08 0.0000000000D+00 0.0000000000D+00 0.0000000000D+00
 324 015396 FE 026 056 0 000 001 -0.1000000000D+01 0.0000000000D+00 0.0000000000D+00 0.7890272999D+29
 325 015397 FE 026 057 0 000 001 -0.1000000000D+01 0.0000000000D+00 0.0000000000D+00 0.1758807969D+28
 326 015398 FE 026 058 0 000 001 -0.1000000000D+01 0.0000000000D+00 0.0000000000D+00 0.2263810953D+27
 327 015981 CO 027 051 0 000 001 -0.1000000000D+01 0.0000000000D+00 0.0000000000D+00 0.0000000000D+00
 328 015982 CO 027 052 0 000 001 0.1150000000D+00 0.0000000000D+00 0.0000000000D+00 0.0000000000D+00
 329 015983 CO 027 053 0 000 001 0.2400000000D+00 0.0000000000D+00 0.0000000000D+00 0.0000000000D+00
 330 015984 CO 027 054 0 000 001 0.1932300000D+00 0.0000000000D+00 0.0000000000D+00 0.0000000000D+00
 331 015985 CO 027 055 0 000 001 0.6310800000D+05 0.0000000000D+00 0.0000000000D+00 0.0000000000D+00
 332 015986 CO 027 056 0 000 001 0.6672931200D+07 0.0000000000D+00 0.0000000000D+00 0.0000000000D+00
 333 015987 CO 027 057 0 000 001 0.2347833600D+08 0.0000000000D+00 0.0000000000D+00 0.0000000000D+00
 334 015988 CO 027 058 0 000 001 0.6122304000D+07 0.0000000000D+00 0.0000000000D+00 0.0000000000D+00
 335 015989 CO 027 059 0 000 001 -0.1000000000D+01 0.0000000000D+00 0.0000000000D+00 0.0000000000D+00
 336
 337
 338
 339
 340
 341
 342

```

343
344
345
346
347 =====
348
349 START of SIMULATION
350
351 Simulation time from 0.000000000E+00 to 0.260000000E+06
352 Beam from 0.000000000E+00 to 0.300000000E+03
353
354 =====
355
356
357
358
359 -----
360 Simulation tally at 0.300000000E+03
361 -----
362
363 -----
364 Isotope Tally 1.2880
365 Key Element Z A M mk sim Half life Decay Constant Activity Reaction Rate Start Atoms Beam
      End Atoms Atoms At Time
366 -----
367 000001 NN 000 001 0 000 001 -0.100000000D+01 0.000000000D+00 0.000000000D+00 0.000000000D+00 0.000000000D
      +00 0.4116613889D+13 0.4116613889D+13
368 000593 H 001 003 0 000 001 0.3885235200D+09 0.1784054619D-08 0.1616389319D+02 0.000000000D+00 0.000000000D
      +00 0.9060200854D+10 0.9060200854D+10
369 001183 HE 002 003 0 000 001 -0.100000000D+01 0.000000000D+00 0.000000000D+00 0.000000000D+00 0.000000000D
      +00 0.242400000D+04 0.242400000D+04
370 013027 TI 022 047 0 000 001 -0.100000000D+01 0.000000000D+00 0.000000000D+00 0.000000000D+00 0.000000000D
      +00 0.5316233671D+07 0.5316233671D+07
371 013029 TI 022 049 0 000 001 -0.100000000D+01 0.000000000D+00 0.000000000D+00 0.000000000D+00 0.000000000D
      +00 0.1736864062D+05 0.1736864062D+05
372 013617 V 023 047 0 000 001 0.195600000D+04 0.3543697242D-03 0.3482463420D+05 0.000000000D+00 0.000000000D
      +00 0.9827203574D+08 0.9827203574D+08
373 013619 V 023 049 0 000 001 0.285120000D+08 0.2431071761D-07 0.1160219131D+03 0.000000000D+00 0.000000000D
      +00 0.4772459413D+10 0.4772459413D+10
374 013621 V 023 051 0 000 001 -0.100000000D+01 0.000000000D+00 0.000000000D+00 0.000000000D+00 0.000000000D
      +00 0.2509195042D+07 0.2509195042D+07
375 013622 V 023 052 0 000 001 0.224580000D+03 0.3086415445D-02 0.1841703348D+01 0.000000000D+00 0.000000000D
      +00 0.5967127177D+03 0.5967127177D+03
376 013623 V 023 053 0 000 001 0.960000000D+02 0.7220283131D-02 0.1640359913D-06 0.000000000D+00 0.000000000D
      +00 0.2271877547D-04 0.2271877547D-04
377 014209 CR 024 049 0 000 001 0.253800000D+04 0.2731076362D-03 0.1900894319D+06 0.000000000D+00 0.000000000D
      +00 0.6960238630D+09 0.6960238630D+09
378 014210 CR 024 050 0 000 001 -0.100000000D+01 0.000000000D+00 0.000000000D+00 0.000000000D+00 0.000000000D
      +00 0.4621020723D+09 0.4621020723D+09
379 014211 CR 024 051 0 000 001 0.2393496000D+07 0.2895961307D-06 0.1672958542D+05 0.000000000D+00 0.000000000D
      +00 0.5776867728D+11 0.5776867728D+11
380 014212 CR 024 052 0 000 001 -0.100000000D+01 0.000000000D+00 0.000000000D+00 0.000000000D+00 0.000000000D
      +00 0.1648354473D+08 0.1648354473D+08
381 014213 CR 024 053 0 000 001 -0.100000000D+01 0.000000000D+00 0.000000000D+00 0.000000000D+00 0.000000000D
      +00 0.4169728000D+07 0.4169728000D+07
382 014214 CR 024 054 0 000 001 -0.100000000D+01 0.000000000D+00 0.000000000D+00 0.000000000D+00 0.000000000D
      +00 0.1470632000D+07 0.1470632000D+07

```

383 014215 CR 024 055 0 000 001 0.2098200000D+03 0.3303532459D-02 0.4539932896D+04 0.0000000000D+00 0.0000000000D
 +00 0.1374266169D+07 0.1374266169D+07
 384 014216 CR 024 056 0 000 001 0.3564000000D+03 0.1944857409D-02 0.1476992152D-04 0.0000000000D+00 0.0000000000D
 +00 0.7594346737D-02 0.7594346737D-02
 385 014800 MN 025 050 0 000 001 0.2838800000D+00 0.2441690787D+01 0.1542445948D+07 0.0000000000D+00 0.0000000000D
 +00 0.6317122367D+06 0.6317122367D+06
 386 014801 MN 025 051 0 000 001 0.2772000000D+04 0.2500530954D-03 0.1285259866D+06 0.0000000000D+00 0.0000000000D
 +00 0.5139947831D+09 0.5139947831D+09
 387 014802 MN 025 052 0 000 001 0.4830624000D+06 0.1434901952D-05 0.1098860016D+06 0.0000000000D+00 0.0000000000D
 +00 0.7658084333D+11 0.7658084333D+11
 388 014803 MN 025 053 0 000 001 0.1179446400D+15 0.5876885805D-14 0.6277577829D-03 0.0000000000D+00 0.0000000000D
 +00 0.1068181012D+12 0.1068181012D+12
 389 014804 MN 025 054 0 000 001 0.2696716800D+08 0.2570337310D-07 0.9804220660D+04 0.0000000000D+00 0.0000000000D
 +00 0.3814371220D+12 0.3814371220D+12
 390 014805 MN 025 055 0 000 001 -0.1000000000D+01 0.0000000000D+00 0.0000000000D+00 0.0000000000D+00 0.0000000000D
 +00 0.2614130226D+07 0.2614130226D+07
 391 014806 MN 025 056 0 000 001 0.9284040000D+04 0.7466008123D-04 0.1314578074D+06 0.0000000000D+00 0.0000000000D
 +00 0.1760750930D+10 0.1760750930D+10
 392 014807 MN 025 057 0 000 001 0.8540000000D+02 0.8116477524D-02 0.3383153467D+06 0.0000000000D+00 0.0000000000D
 +00 0.4168253355D+08 0.4168253355D+08
 393 015392 FE 026 052 0 000 001 0.2979000000D+05 0.2326778048D-04 0.7957468329D+05 0.0000000000D+00 0.0000000000D
 +00 0.3419951609D+10 0.3419951609D+10
 394 015393 FE 026 053 0 000 001 0.5106000000D+03 0.1357515042D-02 0.6332089595D+08 0.0000000000D+00 0.0000000000D
 +00 0.4664471036D+11 0.4664471036D+11
 395 015394 FE 026 054 0 000 001 -0.1000000000D+01 0.0000000000D+00 0.0000000000D+00 0.0000000000D+00 0.5405383878D
 +28 0.5405383878D+28 0.5405383878D+28
 396 015395 FE 026 055 0 000 001 0.8631403200D+08 0.8030527187D-08 0.1215087645D+05 0.0000000000D+00 0.0000000000D
 +00 0.1513085775D+13 0.1513085775D+13
 397 015396 FE 026 056 0 000 001 -0.1000000000D+01 0.0000000000D+00 0.0000000000D+00 0.0000000000D+00 0.7890272999D
 +29 0.7890272999D+29 0.7890272999D+29
 398 015397 FE 026 057 0 000 001 -0.1000000000D+01 0.0000000000D+00 0.0000000000D+00 0.0000000000D+00 0.1758807969D
 +28 0.1758807969D+28 0.1758807969D+28
 399 015398 FE 026 058 0 000 001 -0.1000000000D+01 0.0000000000D+00 0.0000000000D+00 0.0000000000D+00 0.2263810953D
 +27 0.2263810953D+27 0.2263810953D+27
 400 015983 CO 027 053 0 000 001 0.2400000000D+00 0.2888113252D+01 0.4251641576D+06 0.0000000000D+00 0.0000000000D
 +00 0.1472117332D+06 0.1472117332D+06
 401 015984 CO 027 054 0 000 001 0.1932300000D+00 0.3587161313D+01 0.8026455518D+07 0.0000000000D+00 0.0000000000D
 +00 0.2237550759D+07 0.2237550759D+07
 402 015985 CO 027 055 0 000 001 0.6310800000D+05 0.1098350733D-04 0.1029731940D+07 0.0000000000D+00 0.0000000000D
 +00 0.9375256095D+11 0.9375256095D+11
 403 015986 CO 027 056 0 000 001 0.6672931200D+07 0.1038744683D-06 0.9691497262D+04 0.0000000000D+00 0.0000000000D
 +00 0.9330009018D+11 0.9330009018D+11
 404 015987 CO 027 057 0 000 001 0.2347833600D+08 0.2952284100D-07 0.1089088520D+03 0.0000000000D+00 0.0000000000D
 +00 0.3688969230D+10 0.3688969230D+10
 405 015988 CO 027 058 0 000 001 0.6122304000D+07 0.1132167205D-06 0.2517465310D+02 0.0000000000D+00 0.0000000000D
 +00 0.2223580845D+09 0.2223580845D+09
 406
 407 -----
 408 Simulation tally at 0.2600000000E+06
 409 -----
 410
 411 -----

412 Isotope Tally 1.7640
 413 Key Element Z A M mk sim Half life Decay Constant Activity Reaction Rate Start Atoms Beam
 End Atoms At Time
 414 -----

415 000001 NN 000 001 0 000 001 -0.1000000000D+01 0.0000000000D+00 0.0000000000D+00 0.0000000000D+00 0.0000000000D
 +00 0.4116613889D+13 0.4116613889D+13
 416 000593 H 001 003 0 000 001 0.3885235200D+09 0.1784054619D-08 0.1615640588D+02 0.0000000000D+00 0.0000000000D

+00 0.9060200854D+10 0.9056004063D+10
 417 001183 HE 002 003 0 000 001 -0.1000000000D+01 0.0000000000D+00 0.0000000000D+00 0.0000000000D+00 0.0000000000D+00
 +00 0.242400000D+04 0.4199214759D+07
 418 013027 TI 022 047 0 000 001 -0.1000000000D+01 0.0000000000D+00 0.0000000000D+00 0.0000000000D+00 0.0000000000D+00
 +00 0.5316233671D+07 0.1035882694D+09
 419 013029 TI 022 049 0 000 001 -0.1000000000D+01 0.0000000000D+00 0.0000000000D+00 0.0000000000D+00 0.0000000000D+00
 +00 0.1736864062D+05 0.6440824599D+08
 420 013619 V 023 049 0 000 001 0.2851200000D+08 0.2431071761D-07 0.1313773643D+03 0.0000000000D+00 0.0000000000D+00
 +00 0.4772459413D+10 0.5404092399D+10
 421 013621 V 023 051 0 000 001 -0.1000000000D+01 0.0000000000D+00 0.0000000000D+00 0.0000000000D+00 0.0000000000D+00
 +00 0.2509195042D+07 0.8409826886D+10
 422 014210 CR 024 050 0 000 001 -0.1000000000D+01 0.0000000000D+00 0.0000000000D+00 0.0000000000D+00 0.0000000000D+00
 +00 0.4621020723D+09 0.4627337845D+09
 423 014211 CR 024 051 0 000 001 0.2393496000D+07 0.2895961307D-06 0.1444370965D+05 0.0000000000D+00 0.0000000000D+00
 +00 0.5776867728D+11 0.4987535437D+11
 424 014212 CR 024 052 0 000 001 -0.1000000000D+01 0.0000000000D+00 0.0000000000D+00 0.0000000000D+00 0.0000000000D+00
 +00 0.1648354473D+08 0.4857309090D+11
 425 014213 CR 024 053 0 000 001 -0.1000000000D+01 0.0000000000D+00 0.0000000000D+00 0.0000000000D+00 0.0000000000D+00
 +00 0.4169728000D+07 0.4170359063D+07
 426 014214 CR 024 054 0 000 001 -0.1000000000D+01 0.0000000000D+00 0.0000000000D+00 0.0000000000D+00 0.0000000000D+00
 +00 0.1470632000D+07 0.2539147607D+10
 427 014802 MN 025 052 0 000 001 0.4830624000D+06 0.1434901952D-05 0.4510767081D+05 0.0000000000D+00 0.0000000000D+00
 +00 0.7658084333D+11 0.3143606485D+11
 428 014803 MN 025 053 0 000 001 0.1179446400D+15 0.5876885805D-14 0.1176009917D-02 0.0000000000D+00 0.0000000000D+00
 +00 0.1068181012D+12 0.2001076685D+12
 429 014804 MN 025 054 0 000 001 0.2696716800D+08 0.2570337310D-07 0.9738993802D+04 0.0000000000D+00 0.0000000000D+00
 +00 0.3814371220D+12 0.3788994450D+12
 430 014805 MN 025 055 0 000 001 -0.1000000000D+01 0.0000000000D+00 0.0000000000D+00 0.0000000000D+00 0.0000000000D+00
 +00 0.2614130226D+07 0.6439401457D+10
 431 015392 FE 026 052 0 000 001 0.2979000000D+05 0.2326778048D-04 0.1890116552D+03 0.0000000000D+00 0.0000000000D+00
 +00 0.3419951609D+10 0.8123321232D+07
 432 015394 FE 026 054 0 000 001 -0.1000000000D+01 0.0000000000D+00 0.0000000000D+00 0.0000000000D+00 0.5405383878D+28
 +28 0.5405383878D+28 0.5405383878D+28
 433 015395 FE 026 055 0 000 001 0.8631403200D+08 0.8030527187D-08 0.1280863433D+05 0.0000000000D+00 0.0000000000D+00
 +00 0.1513085775D+13 0.1594992960D+13
 434 015396 FE 026 056 0 000 001 -0.1000000000D+01 0.0000000000D+00 0.0000000000D+00 0.0000000000D+00 0.7890272999D+29
 +29 0.7890272999D+29 0.7890272999D+29
 435 015397 FE 026 057 0 000 001 -0.1000000000D+01 0.0000000000D+00 0.0000000000D+00 0.0000000000D+00 0.1758807969D+28
 +28 0.1758807969D+28 0.1758807969D+28
 436 015398 FE 026 058 0 000 001 -0.1000000000D+01 0.0000000000D+00 0.0000000000D+00 0.0000000000D+00 0.2263810953D+27
 +27 0.2263810953D+27 0.2263810953D+27
 437 015985 CO 027 055 0 000 001 0.6310800000D+05 0.1098350733D-04 0.5942036078D+05 0.0000000000D+00 0.0000000000D+00
 +00 0.9375256095D+11 0.5409962319D+10
 438 015986 CO 027 056 0 000 001 0.6672931200D+07 0.1038744683D-06 0.9433552328D+04 0.0000000000D+00 0.0000000000D+00
 +00 0.9330009018D+11 0.9081685308D+11
 439 015987 CO 027 057 0 000 001 0.2347833600D+08 0.2952284100D-07 0.1080770318D+03 0.0000000000D+00 0.0000000000D+00
 +00 0.3688969230D+10 0.3660793752D+10
 440 015988 CO 027 058 0 000 001 0.6122304000D+07 0.1132167205D-06 0.2444523420D+02 0.0000000000D+00 0.0000000000D+00
 +00 0.2223580845D+09 0.2159154063D+09

441

442

443 Gamma Lines at end of Simulation 0.2600000000E+06

444

Z	A	M	Gamma Energy/KeV	Gamma count/s
24	51	0	0.3200824000E+03	0.1432815997E+04
25	52	0	0.2005800000E+03	0.3428182982E+02
25	52	0	0.3460199000E+03	0.4420551289E+03
25	52	0	0.3980899000E+03	0.4014582702E+02
25	52	0	0.3995700000E+03	0.8254703759E+02
25	52	0	0.5020600000E+03	0.9472610871E+02
25	52	0	0.6001600000E+03	0.1759199162E+03

453	25	52	0	0.6474699000E+03	0.1804306833E+03
454	25	52	0	0.7442329000E+03	0.4059690373E+05
455	25	52	0	0.8481800000E+03	0.1497574671E+04
456	25	52	0	0.9018899000E+03	0.1984737065E+02
457	25	52	0	0.9355440000E+03	0.4262674892E+05
458	25	52	0	0.1045750000E+04	0.3157536957E+02
459	25	52	0	0.1246278000E+04	0.1899032941E+04
460	25	52	0	0.1247879000E+04	0.1714091491E+03
461	25	52	0	0.1333648000E+04	0.2286958910E+04
462	25	52	0	0.1434092000E+04	0.4510767081E+05
463	25	52	0	0.1645819000E+04	0.2120060528E+02
464	25	52	0	0.1981120000E+04	0.1533660808E+02
465	25	52	0	0.2257419000E+04	0.1217907112E+01
466	25	54	0	0.8348479000E+03	0.9736656444E+04
467	26	52	0	0.1686880000E+03	0.1874526871E+03
468	26	52	0	0.3777479000E+03	0.3106047431E+01
469	26	52	0	0.1039928000E+04	0.1798237987E+00
470	26	55	0	0.1259999000E+03	0.1639505194E-04
471	27	55	0	0.9190000000E+02	0.6907616941E+03
472	27	55	0	0.3853999000E+03	0.3208699482E+03
473	27	55	0	0.4114999000E+03	0.6372833694E+03
474	27	55	0	0.4771999000E+03	0.1198805779E+05
475	27	55	0	0.5199999000E+03	0.4902179764E+03
476	27	55	0	0.8036999000E+03	0.1109675238E+04
477	27	55	0	0.8270000000E+03	0.1247827576E+03
478	27	55	0	0.9311000000E+03	0.4456527059E+05
479	27	55	0	0.9845999000E+03	0.3075003670E+03
480	27	55	0	0.1212799000E+04	0.1559783876E+03
481	27	55	0	0.1316600000E+04	0.4211417476E+04
482	27	55	0	0.1369999000E+04	0.1733589026E+04
483	27	55	0	0.1408499000E+04	0.1002718588E+05
484	27	55	0	0.1555999000E+04	0.2718481506E+02
485	27	55	0	0.1622299000E+04	0.2673916235E+02
486	27	55	0	0.1792100000E+04	0.4857614494E+02
487	27	55	0	0.1940599000E+04	0.8467401411E+01
488	27	55	0	0.2144200000E+04	0.5347832470E+02
489	27	55	0	0.2177600000E+04	0.1738045553E+03
490	27	55	0	0.2578700000E+04	0.2540220423E+02
491	27	55	0	0.2872400000E+04	0.6996747482E+02
492	27	55	0	0.2938900000E+04	0.3386960565E+02
493	27	55	0	0.3108300000E+04	0.3119568941E+01
494	27	56	0	0.2634099000E+03	0.2075380569E+01
495	27	56	0	0.4113799000E+03	0.2358388082E+01
496	27	56	0	0.4865399000E+03	0.5754466920E+01
497	27	56	0	0.6550000000E+03	0.3584749885E+01
498	27	56	0	0.6746999000E+03	0.3584749885E+01
499	27	56	0	0.7335109000E+03	0.1792374942E+02
500	27	56	0	0.7877419000E+03	0.2971568983E+02
501	27	56	0	0.8467709000E+03	0.9427420519E+04
502	27	56	0	0.8527800000E+03	0.4716776164E+01
503	27	56	0	0.8965309000E+03	0.8112855002E+01
504	27	56	0	0.9773729000E+03	0.1366921732E+03
505	27	56	0	0.9973299000E+03	0.1216928250E+02
506	27	56	0	0.1037839000E+04	0.1336733422E+04
507	27	56	0	0.1089030000E+04	0.4716776164E+01
508	27	56	0	0.1140404000E+04	0.1292396669E+02
509	27	56	0	0.1159919000E+04	0.8961874712E+01
510	27	56	0	0.1175101000E+04	0.2158396773E+03
511	27	56	0	0.1198780000E+04	0.4811111687E+01
512	27	56	0	0.1238281000E+04	0.6311046508E+04
513	27	56	0	0.1272200000E+04	0.2264052559E+01

514	27	56	0	0.1335389000E+04	0.1113159175E+02
515	27	56	0	0.1360215000E+04	0.4046993949E+03
516	27	56	0	0.1442750000E+04	0.1745207181E+02
517	27	56	0	0.1462340000E+04	0.6131809013E+01
518	27	56	0	0.1640404000E+04	0.6792156733E+01
519	27	56	0	0.1771351000E+04	0.1459370545E+04
520	27	56	0	0.1810771000E+04	0.6018606385E+02
521	27	56	0	0.1963714000E+04	0.6829891886E+02
522	27	56	0	0.2015181000E+04	0.2867799908E+03
523	27	56	0	0.2034755000E+04	0.7443072787E+03
524	27	56	0	0.2113123000E+04	0.3547015675E+02
525	27	56	0	0.2212933000E+04	0.3726253170E+02
526	27	56	0	0.2276360000E+04	0.1207494698E+02
527	27	56	0	0.2373700000E+04	0.7735512909E+01
528	27	56	0	0.2523860000E+04	0.6414815583E+01
529	27	56	0	0.2598459000E+04	0.1632004553E+04
530	27	56	0	0.2657400000E+04	0.1981045989E+01
531	27	56	0	0.3009595000E+04	0.1094292070E+03
532	27	56	0	0.3201962000E+04	0.3131939373E+03
533	27	56	0	0.3253415000E+04	0.7660044491E+03
534	27	56	0	0.3272990000E+04	0.1820674656E+03
535	27	56	0	0.3369690000E+04	0.1037689813E+01
536	27	56	0	0.3451152000E+04	0.9169411920E+02
537	27	56	0	0.3547930000E+04	0.1886710466E+02
538	27	56	0	0.3600700000E+04	0.1556536134E+01
539	27	56	0	0.3611799000E+04	0.8018519479E+00
540	27	57	0	0.1441290000E+02	0.9899856117E+01
541	27	57	0	0.1220606000E+03	0.9251393926E+02
542	27	57	0	0.1364735000E+03	0.1154261619E+02
543	27	57	0	0.3396899000E+03	0.3998850178E-02
544	27	57	0	0.3523300000E+03	0.3242310955E-02
545	27	57	0	0.3667999000E+03	0.1296924382E-02
546	27	57	0	0.5700900000E+03	0.1707617103E-01
547	27	57	0	0.6924099000E+03	0.1610347774E+00
548	27	57	0	0.7065399000E+03	0.5403851592E-02
549	27	58	0	0.8107592000E+03	0.2431078541E+02
550	27	58	0	0.8639510000E+03	0.1686720915E+00
551	27	58	0	0.1674725000E+04	0.1271152178E+00
552					
553					
554					Activities Ordered by Most Active
555					
556					
557	15985 CO	27	55	0	0.5942036078E+05
558	14802 MN	25	52	0	0.4510767081E+05
559	14211 CR	24	51	0	0.1444370965E+05
560	15395 FE	26	55	0	0.1280863433E+05
561	14804 MN	25	54	0	0.9738993802E+04
562	15986 CO	27	56	0	0.9433552328E+04
563	15392 FE	26	52	0	0.1890116552E+03
564	13619 V	23	49	0	0.1313773643E+03
565	15987 CO	27	57	0	0.1080770318E+03
566	15988 CO	27	58	0	0.2444523420E+02
567	593 H	1	3	0	0.1615640588E+02
568	14803 MN	25	53	0	0.1176009917E-02
569					
570					

571	Totals	1.8880
572		

573
574 Total Activity/Bq: 0.1514219906E+06
575 Total Gamma Power/eV/s: 0.2559334931E+12
576 Total Gamma Power/Watts: 0.4100506435E-07
577 Absorbed Dose*/Grays/s: 0.4078849177E-10
578 Absorbed Dose*/Grays/hr: 0.1468385704E-06
579 Fraction of annual dosage if exposed for 1 hr: 0.1468385634E-03
580
581 Absorbed dose, assumes all energy absorbed, 80kg human, 1m from point-target, 1m surface area exposed to
irradiation.
582
583 Dose Limits:
584 employees 18+ 20 millisieverts/year
585 trainees 18+ 6 millisieverts/year
586 public and under 18s 1 millisieverts/year
587 public and under 18s millisieverts averaged per hour: 0.1140771128E-03
588 Dose averaged over area of skin not exceeding 1cm²
589 Source: <http://www.hse.gov.uk/radiation/ionising/doses/>
590
591
592
593 -----
594 Activity calculation complete. 1.8880
595 -----

Appendix F

Activity V2 Full Results for Iron

Output file produced by Activity V2: https://github.com/BenPalmer1983/activity_v2

Listing F.1: Isotope Activity at End of Simulation

```
1 3.209E-03 ### Start
2 4.769E-01 ### Loading isotope tendl xs
3 5.289E+00 ### Load complete
4 5.292E+00 ### Load EXYZ.txt
5 5.500E+00 ### Run Simulation
6 5.502E+00 ### Prep sim1
7 5.514E+00 ###
8 5.517E+00 ### #####
9 5.520E+00 ### Starting Tally
10 5.522E+00 ### #####
11 5.523E+00 ###
12 5.526E+00 ### 26054 Fe54 1.630585560412975e+20
13 5.528E+00 ### 26056 Fe56 2.468222518377967e+21
14 5.530E+00 ### 26057 Fe57 5.600277953618698e+19
15 5.531E+00 ### 26058 Fe58 7.334611933451605e+18
16 5.533E+00 ### 27053 Co53 0.0
17 5.535E+00 ### 25050 Mn50 0.0
18 5.536E+00 ### 26053 Fe53 0.0
19 5.538E+00 ### 26052 Fe52 0.0
20 5.540E+00 ### 25051 Mn51 0.0
21 5.542E+00 ### 25052 Mn52 0.0
22 5.543E+00 ### 24049 Cr49 0.0
23 5.545E+00 ### 27055 Co55 0.0
24 5.547E+00 ### 23047 V47 0.0
25 5.548E+00 ### 25053 Mn53 0.0
26 5.550E+00 ### 24050 Cr50 0.0
27 5.552E+00 ### 26055 Fe55 0.0
28 5.553E+00 ### 25054 Mn54 0.0
29 5.555E+00 ### 24051 Cr51 0.0
30 5.557E+00 ### 27057 Co57 0.0
31 5.558E+00 ### 23049 V49 0.0
32 5.560E+00 ### 25055 Mn55 0.0
33 5.562E+00 ### 24052 Cr52 0.0
34 5.563E+00 ### 27056 Co56 0.0
35 5.565E+00 ### 27058 Co58 0.0
36 5.567E+00 ### 23050 V50 0.0
37 5.568E+00 ### 25056 Mn56 0.0
38 5.570E+00 ### 24053 Cr53 0.0
39 5.572E+00 ### 27059 Co59 0.0
40 5.573E+00 ### 23051 V51 0.0
41 5.575E+00 ### 25057 Mn57 0.0
```

```

42  5.577E+00 ### 24054      Cr54      0.0
43  5.578E+00 ### 1          Nn1       0.0
44  5.580E+00 ### 2004      He4       0.0
45  5.582E+00 ### 1001      H1        0.0
46  5.584E+00 ### 1002      H2        0.0
47  5.585E+00 ### 2003      He3       0.0
48  5.587E+00 ### 1003      H3        0.0
49  5.589E+00 ### 1025052    Mn52-M    0.0
50  5.590E+00 ### 22049     Ti49       0.0
51  5.592E+00 ### 22047     Ti47       0.0
52  5.594E+00 ### 22050     Ti50       0.0
53  5.596E+00 ### ##### #####
54  5.601E+00 ### Run sim1
55  5.603E+00 ###
56  5.605E+00 ### ##### #####
57  5.606E+00 ### ##### #####
58  5.608E+00 ### Run Sim
59  5.610E+00 ### ##### #####
60  5.612E+00 ### ##### #####
61  5.613E+00 ###
62  5.615E+00 ###
63  5.617E+00 ### ##### #####
64  5.618E+00 ### Plot Ion Energy Lost
65  5.620E+00 ### ##### #####
66  5.622E+00 ###
67  1.424E+01 ###
68  1.424E+01 ### ##### #####
69  1.424E+01 ### Number Density
70  1.424E+01 ### ##### #####
71  1.424E+01 ###
72  1.425E+01 ### Fe54      5.095579876290546e+27
73  1.425E+01 ### Fe56      7.713195369931147e+28
74  1.425E+01 ### Fe57      1.750086860505843e+27
75  1.425E+01 ### Fe58      2.2920662292036265e+26
76  1.426E+01 ###
77  1.426E+01 ### ##### #####
78  1.426E+01 ### Calculate reaction rates
79  1.426E+01 ### ##### #####
80  1.426E+01 ###
81  1.429E+01 ### Fe54      --> Co53      307283.956904
82  1.432E+01 ### Fe54      --> Mn50      413574.876348
83  1.436E+01 ### Fe54      --> Fe53      236075945.829
84  1.440E+01 ### Fe54      --> Fe52      131778.988756
85  1.444E+01 ### Fe54      --> Mn51      3915365.42731
86  1.449E+01 ### Fe54      --> Mn52      4550309.60658
87  1.452E+01 ### Fe54      --> Cr49      22.5007021259
88  1.457E+01 ### Fe54      --> Co55      207974.632938
89  1.461E+01 ### Fe54      --> Fe54      114453151.897
90  1.464E+01 ### Fe54      --> V47       12118.304033
91  1.467E+01 ### Fe54      --> Mn53      403877118.991
92  1.471E+01 ### Fe54      --> Cr50      37369410.8242
93  1.475E+01 ### Fe56      --> Co55      845335432.044
94  1.478E+01 ### Fe56      --> Mn52      329450373.437
95  1.481E+01 ### Fe56      --> Mn51      2.53001350218
96  1.485E+01 ### Fe56      --> Fe55      8050196691.17
97  1.490E+01 ### Fe56      --> Fe54      378463315.972
98  1.493E+01 ### Fe56      --> Fe53      8.38602414208
99  1.498E+01 ### Fe56      --> Mn53      109015450.05
100 1.503E+01 ### Fe56      --> Mn54      56915074.2745
101 1.506E+01 ### Fe56      --> Cr51      120955.374078
102 1.510E+01 ### Fe56      --> Co57      5848557.76264

```

103	1.513E+01	### Fe56	--> Fe56	1680968637.75
104	1.517E+01	### Fe56	--> V49	1174843.18811
105	1.522E+01	### Fe56	--> Mn55	715362266.42
106	1.526E+01	### Fe56	--> Cr52	374707737.696
107	1.529E+01	### Fe57	--> Fe54	52183.7852258
108	1.532E+01	### Fe57	--> Co56	36640319.083
109	1.535E+01	### Fe57	--> Co55	1074873.1321
110	1.538E+01	### Fe57	--> Mn53	19968722.6313
111	1.541E+01	### Fe57	--> Mn52	4360.74832679
112	1.547E+01	### Fe57	--> Fe56	164897136.47
113	1.552E+01	### Fe57	--> Fe55	33178633.716
114	1.558E+01	### Fe57	--> Mn54	2427065.011
115	1.563E+01	### Fe57	--> Mn55	2319212.32735
116	1.568E+01	### Fe57	--> Cr52	551009.138609
117	1.573E+01	### Fe57	--> Co58	102227.683704
118	1.577E+01	### Fe57	--> Fe57	27066748.1304
119	1.580E+01	### Fe57	--> V50	28017.8372826
120	1.583E+01	### Fe57	--> Mn56	5147637.47599
121	1.587E+01	### Fe57	--> Cr53	1928918.81162
122	1.590E+01	### Fe58	--> Fe55	6414.68780283
123	1.594E+01	### Fe58	--> Co57	11190415.5404
124	1.598E+01	### Fe58	--> Co56	692558.874587
125	1.606E+01	### Fe58	--> Mn54	2024539.65839
126	1.611E+01	### Fe58	--> Mn53	34624.323756
127	1.616E+01	### Fe58	--> Fe57	11896984.9875
128	1.621E+01	### Fe58	--> Fe56	7789545.04264
129	1.627E+01	### Fe58	--> Mn55	232716.626174
130	1.632E+01	### Fe58	--> Mn56	46497.0277258
131	1.635E+01	### Fe58	--> Cr53	607.862336665
132	1.639E+01	### Fe58	--> Co59	13432.4749748
133	1.643E+01	### Fe58	--> Fe58	4608236.90701
134	1.647E+01	### Fe58	--> V51	3894.08635149
135	1.651E+01	### Fe58	--> Mn57	97973.3183318
136	1.654E+01	### Fe58	--> Cr54	69429.6222128
137	1.655E+01	###		
138	1.665E+01	### Fe54	--> Nn1	223779307.551
139	1.675E+01	### Fe54	--> He4	41722308.3148
140	1.690E+01	### Fe54	--> H1	1185614187.94
141	1.696E+01	### Fe54	--> H2	16816480.0199
142	1.701E+01	### Fe54	--> He3	1195803.75217
143	1.706E+01	### Fe54	--> H3	63495.6126689
144	1.719E+01	### Fe56	--> Nn1	10459239812.3
145	1.729E+01	### Fe56	--> He4	815365435.08
146	1.742E+01	### Fe56	--> H1	11592247091.2
147	1.748E+01	### Fe56	--> H2	390691314.24
148	1.753E+01	### Fe56	--> H3	7727635.45562
149	1.758E+01	### Fe56	--> He3	19451763.8089
150	1.776E+01	### Fe57	--> Nn1	317362293.916
151	1.783E+01	### Fe57	--> H2	11740193.0149
152	1.795E+01	### Fe57	--> He4	24925914.0685
153	1.811E+01	### Fe57	--> H1	229114996.365
154	1.817E+01	### Fe57	--> H3	530012.860186
155	1.820E+01	### Fe57	--> He3	621761.67143
156	1.836E+01	### Fe58	--> Nn1	52586048.5862
157	1.843E+01	### Fe58	--> H2	1338414.90645
158	1.855E+01	### Fe58	--> He4	2368936.32236
159	1.869E+01	### Fe58	--> H1	23169852.7974
160	1.873E+01	### Fe58	--> H3	63386.958919
161	1.878E+01	### Fe58	--> He3	45022.5126778
162	1.878E+01	###		
163	1.878E+01	###		

```

164 1.878E+01 ### #####
165 1.878E+01 ### Saturation Times
166 1.878E+01 ### #####
167 1.879E+01 ###
168 1.879E+01 ### Time where saturation is 95%
169 1.879E+01 ### Co53      1.03726274277
170 1.879E+01 ### Mn50      1.22392681719
171 1.880E+01 ### Fe53      2206.77648525
172 1.880E+01 ### Fe52      128719.120064
173 1.880E+01 ### Mn51      11980.384679
174 1.880E+01 ### Mn52      2089254.61649
175 1.881E+01 ### Cr49      10865.3272305
176 1.881E+01 ### Co55      272748.238212
177 1.881E+01 ### V47       8453.6913536
178 1.881E+01 ### Mn53      5.01914153515e+14
179 1.881E+01 ### Cr50      2.45501506924e+25
180 1.881E+01 ### Fe55      374654548.569
181 1.882E+01 ### Mn54      116576230.889
182 1.882E+01 ### Cr51      10345096.7458
183 1.882E+01 ### Co57      101493998.416
184 1.882E+01 ### V49       123226813.841
185 1.882E+01 ### Co56      28841047.3435
186 1.883E+01 ### Co58      26456423.5172
187 1.883E+01 ### V50       1.90941486654e+25
188 1.883E+01 ### Mn56      40123.094881
189 1.883E+01 ### Mn57      369.092659303
190 1.883E+01 ### Nn1       2656.25700712
191 1.883E+01 ### H3        1681683831.36
192 1.884E+01 ###
193 2.136E+01 ###
194 2.136E+01 ### #####
195 2.137E+01 ### End of Beam Tally
196 2.137E+01 ### #####
197 2.137E+01 ###
198 2.137E+01 ### 26054     Fe54       1.63058555949e+20    0.0
199 2.137E+01 ### 26056     Fe56       2.4682225152e+21    0.0
200 2.138E+01 ### 26057     Fe57       5.60027794695e+19    0.0
201 2.138E+01 ### 26058     Fe58       7.33461192328e+18    0.0
202 2.138E+01 ### 27053     Co53      106396.089785      307283.956904
203 2.138E+01 ### 26053     Fe53       58250591793.0      79076054.5775
204 2.138E+01 ### 25053     Mn53      172540199605.0      0.00102982599877
205 2.139E+01 ### 24053     Cr53      578858002.186      0.0
206 2.139E+01 ### 25050     Mn50      0.0          0.0
207 2.139E+01 ### 24050     Cr50      0.0          0.0
208 2.139E+01 ### 26052     Fe52       39396004.7064      916.8791916
209 2.139E+01 ### 1025052   Mn52-M    130481.519119      71.1029065191
210 2.139E+01 ### 24052     Cr52      112577624050.0      0.0
211 2.140E+01 ### 25052     Mn52      100179964870.0      143645.657909
212 2.140E+01 ### 25051     Mn51      1131634453.28      282968.697949
213 2.140E+01 ### 24051     Cr51      79259717.7947      22.9520226273
214 2.140E+01 ### 23051     V51       1168225.90545      0.0
215 2.140E+01 ### 24049     Cr49      6478.58118515      1.78624116249
216 2.141E+01 ### 23049     V49       352451942.801      8.56835965333
217 2.141E+01 ### 22049     Ti49      704918128.027      0.0
218 2.141E+01 ### 27055     Co55      253567495453.0      2785060.44487
219 2.141E+01 ### 26055     Fe55      2.42542960147e+12    19393.6995083
220 2.141E+01 ### 25055     Mn55      215374258612.0      0.0
221 2.141E+01 ### 23047     V47       3448914.86405      1222.19100921
222 2.142E+01 ### 22047     Ti47      7084406.07395      0.0
223 2.142E+01 ### 25054     Mn54      18409932719.4      473.091548602
224 2.142E+01 ### 24054     Cr54      20828886.6638      0.0

```

225	2.142E+01	###	27057	Co57	5111669359.24	150.877816523
226	2.142E+01	###	27056	Co56	11199688888.9	1163.31661116
227	2.143E+01	###	27058	Co58	30667784.2188	3.47259601749
228	2.143E+01	###	23050	V50	0.0	0.0
229	2.143E+01	###	22050	Ti50	0.0	0.0
230	2.143E+01	###	25056	Mn56	1540918363.29	115050.418855
231	2.143E+01	###	27059	Co59	4029742.49245	0.0
232	2.143E+01	###	25057	Mn57	11013470.9442	89390.5893812
233	2.144E+01	###	1	Nn1	2.81319554815e+12	3172727892.23
234	2.144E+01	###	1001	H1	3.90904383849e+12	0.0
235	2.144E+01	###	2004	He4	265314778136.0	0.0
236	2.144E+01	###	1002	H2	126175920654.0	0.0
237	2.144E+01	###	2003	He3	11425023383.8	0.0
238	2.145E+01	###	1003	H3	2515358594.54	4.48083092714
239	2.145E+01	###				
240	3.152E+01	###				
241	3.152E+01	###				
242	3.152E+01	###	End of Sim Tally			
243	3.153E+01	###				
244	3.153E+01	###				
245	3.153E+01	###	26054	Fe54	1.63058555949e+20	0.0
246	3.153E+01	###	26056	Fe56	2.4682225152e+21	0.0
247	3.153E+01	###	26057	Fe57	5.60027794696e+19	0.0
248	3.153E+01	###	26058	Fe58	7.33461192328e+18	0.0
249	3.154E+01	###	27053	Co53	0.0	0.0
250	3.154E+01	###	26053	Fe53	4.53141271569e-143	6.15146092407e-146
251	3.154E+01	###	25053	Mn53	230790897436.0	0.00137750197927
252	3.155E+01	###	24053	Cr53	578858002.186	0.0
253	3.155E+01	###	25050	Mn50	0.0	0.0
254	3.156E+01	###	24050	Cr50	0.0	0.0
255	3.156E+01	###	26052	Fe52	93439.6978154	2.17465997465
256	3.157E+01	###	1025052	Mn52-M	4168.78150109	2.27168171686
257	3.157E+01	###	24052	Cr52	143724178764.0	0.0
258	3.157E+01	###	25052	Mn52	69033917523.9	98986.0848283
259	3.157E+01	###	25051	Mn51	7.09820416491e-20	1.7749279235e-23
260	3.157E+01	###	24051	Cr51	1124387120.91	325.59993095
261	3.158E+01	###	23051	V51	1168225.90545	0.0
262	3.158E+01	###	24049	Cr49	5.183538587e-28	1.42917866225e-31
263	3.158E+01	###	23049	V49	350240187.817	8.51459030248
264	3.158E+01	###	22049	Ti49	704918128.027	0.0
265	3.158E+01	###	27055	Co55	14632033320.7	160711.045225
266	3.158E+01	###	26055	Fe55	2.65898143279e+12	21261.1765249
267	3.159E+01	###	25055	Mn55	215374258612.0	0.0
268	3.159E+01	###	23047	V47	3.71227906462e-34	1.31551930833e-37
269	3.159E+01	###	22047	Ti47	7084406.07395	0.0
270	3.159E+01	###	25054	Mn54	18287479903.7	469.944802056
271	3.159E+01	###	24054	Cr54	20828886.6638	0.0
272	3.160E+01	###	27057	Co57	5072636183.79	149.725699695
273	3.160E+01	###	27056	Co56	10901613948.1	1132.35543595
274	3.160E+01	###	27058	Co58	29779081.9005	3.37196585428
275	3.160E+01	###	23050	V50	0.0	0.0
276	3.161E+01	###	22050	Ti50	0.0	0.0
277	3.161E+01	###	25056	Mn56	5.84462973621	0.000436380742305
278	3.161E+01	###	27059	Co59	4029742.49245	0.0
279	3.161E+01	###	25057	Mn57	0.0	0.0
280	3.161E+01	###	1	Nn1	1.77254842634e-115	1.99908386613e-118
281	3.161E+01	###	1001	H1	3.90904383849e+12	0.0
282	3.162E+01	###	2004	He4	265314778136.0	0.0
283	3.162E+01	###	1002	H2	126175920654.0	0.0
284	3.162E+01	###	2003	He3	11426186786.5	0.0
285	3.162E+01	###	1003	H3	2514195191.88	4.478758455

```
286 3.162E+01 ###
287 4.344E+01 ###
288 4.344E+01 ### Gamma Dose - Beam End
289 4.344E+01 ### Activity/Bq           82822873.7128
290 4.345E+01 ### Power eV/s          1.86841008076e+16
291 4.345E+01 ### Power J/s           0.00299352926319
292 4.345E+01 ### Dose Gy/s          2.97685885361e-06
293 4.345E+01 ### Dose Gy/hr          0.010716691873
294 4.345E+01 ### Percentage of annual dose/hr 9394.25237015
295 4.345E+01 ###
296 4.346E+01 ###
297 4.346E+01 ### Gamma Dose - Sim End
298 4.346E+01 ### Activity/Bq           283043.751191
299 4.346E+01 ### Power eV/s          5.1310279645e+14
300 4.346E+01 ### Power J/s           8.22083038417e-05
301 4.347E+01 ### Dose Gy/s          1.96294899335e-07
302 4.347E+01 ### Dose Gy/hr          0.000294301803729
303 4.347E+01 ### Percentage of annual dose/hr 257.984968681
304 4.347E+01 ###
```

Appendix G

Neutron Activity Code

G.1 Full Source Code

The full source code is available to download from github.

https://github.com/BenPalmer1983/neutron_activation

G.2 Implementation of the Activity Equations in Python

Listing G.1: Decay Equation Code in Python3

```
#####
# DECAY EQUATIONS
#####

@staticmethod
def activity(t, l, b, w, n0):
    nt = numpy.zeros((len(n0),))
    for m in range(0,len(n0)):
        if(l[m] > 0.0):
            nt[m] = isotopes.activity_unstable(t, l, b, w, n0, m)
        elif(l[m] == 0.0):
            nt[m] = isotopes.activity_stable(t, l, b, w, n0, m)
    return nt

@staticmethod
def activity_unstable(t, l, b, w, n0, m):
    s = 0.0
    for k in range(0, m+1):
        s = s + isotopes.r(k, m, b, l) * ( isotopes.f_unstable(t,k,m,l) * n0[k] + isotopes.g_unstable(t,k,m,l) * w[k] )
    return s

@staticmethod
def activity_stable(t, l, b, w, n0, m):
    s = n0[m] + w[m] * t
    for k in range(0, m):
        s = s + isotopes.r(k, m, b, l) * (isotopes.f_stable(t,k,m,l) * n0[k] + isotopes.g_stable(t,k,m,l) * w[k])
    return s

@staticmethod
def r(k, m, b, l):
```

```

31     if(k == m):
32         return 1.0
33     else:
34         p = 1.0
35         for i in range(k, m):
36             p = p * (b[i] * l[i])
37         return p
38
39 @staticmethod
40 def f_unstable(t,k,m,l):
41     s = 0.0
42     for i in range(k, m+1):
43         p = 1.0
44         for j in range(k, m+1):
45             if(i != j):
46                 p = p * (1 / (l[i] - l[j]))
47             s = s + numpy.exp(-1 * l[i] * t) * p
48     s = (-1)**(m-k) * s
49     return s
50
51 @staticmethod
52 def g_unstable(t,k,m,l):
53     pa = 1.0
54     for i in range(k,m+1):
55         pa = pa * l[i]
56     pa = 1.0 / pa
57     s = 0.0
58     for i in range(k, m+1):
59         pb = 1.0
60         for j in range(k, m+1):
61             if(i != j):
62                 pb = pb * (1 / (l[i]-l[j]))
63             s = s + (1/l[i]) * numpy.exp(-1*i*t) * pb
64     return pa + s * (-1)**(m-k+1)
65
66
67 @staticmethod
68 def f_stable(t,k,m_in,l):
69     m = m_in - 1
70
71     p = 1.0
72     for i in range(k, m+1):
73         p = p * l[i]
74
75     s = 0.0
76     for i in range(k, m+1):
77         r = l[i]
78         for j in range(k, m+1):
79             if(i != j):
80                 r = r * (l[i] - l[j])
81             s = s + (1/r)*numpy.exp(-1*l[i]*t)
82
83     return (1.0/p) + s * (-1.0)**(m-k+1)
84
85
86 @staticmethod
87 def g_stable(t,k,m_in,l):
88     m = m_in - 1
89
90     pa = 1.0
91     for i in range(k,m+1):

```

```
92     pa = pa * l[i]
93     pa = 1.0 / pa
94
95     sa = 0.0
96     for i in range(k, m+1):
97         pb = 1.0
98         for j in range(k,m+1):
99             if(j != i):
100                 pb = pb * l[j]
101             sa = sa + pb
102         pc = 1.0
103         for i in range(k, m+1):
104             pc = pc * l[i]**2
105
106         sb = 0.0
107         for i in range(k, m+1):
108             pd = 1.0
109             for j in range(k, m+1):
110                 if(i != j):
111                     pd = pd * (1 / (l[i]-l[j]))
112             sb = sb + (1/(l[i]**2)) * numpy.exp(-l[i]*t) * pd
113
114     return pa * t + sa / pc + sb * (-1)**(m-k+1)
```

Appendix H

Iron Activity

H.1 Activity vs Beam Energy

The beam energy was set to 5MeV, 10MeV, 15MeV, 20MeV and 25MeV within SRIM, and the Activity V2 code was used to calculate how radioactive each sample would be if irradiated to 100 DPA of damage.

The beam settings are:

- target material = pure iron
- target thickness = 0.5mm
- beam current = 60 micro amps (maximum proton current for the Scanditronix MC-40)
- beam area = 64mm^2

H.1.1 5MeV Proton Beam - 60 microamp 100DPA

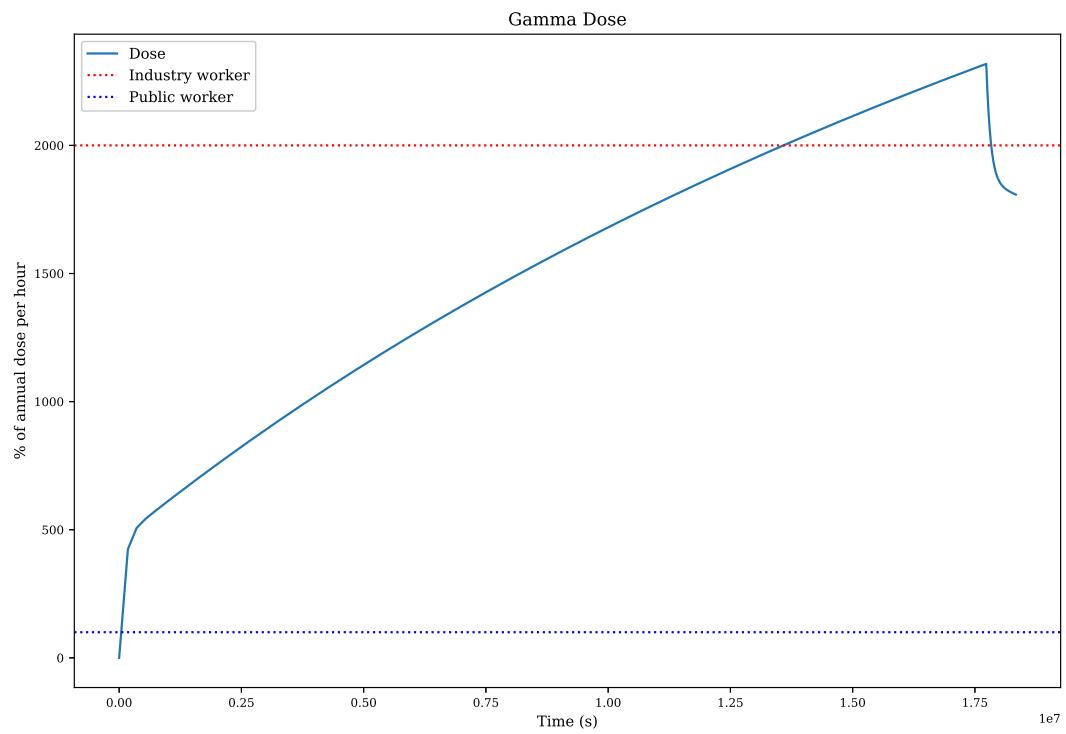


Figure H.1: Percentage of Annual Dose Equivalent

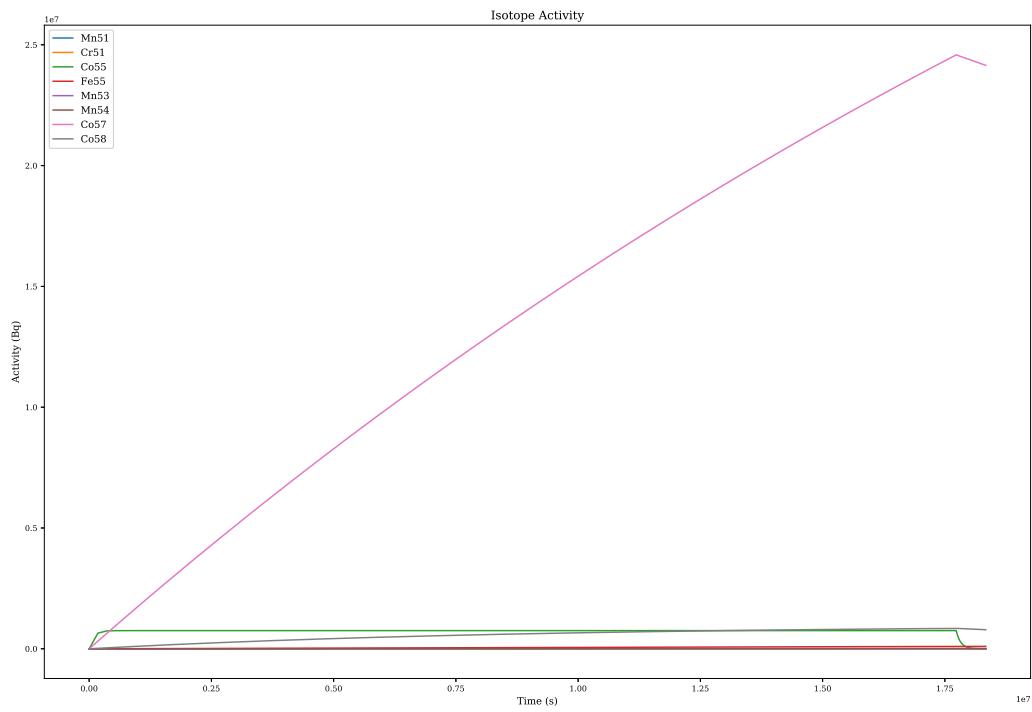


Figure H.2: Activity of all isotopes over time

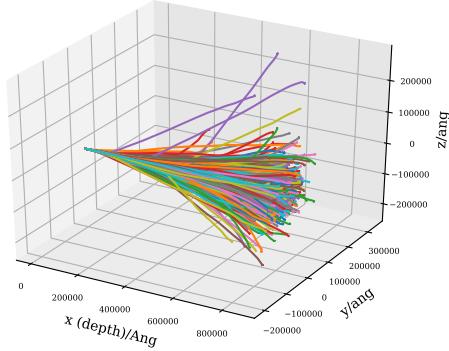


Figure H.3: Ion trajectory in the target

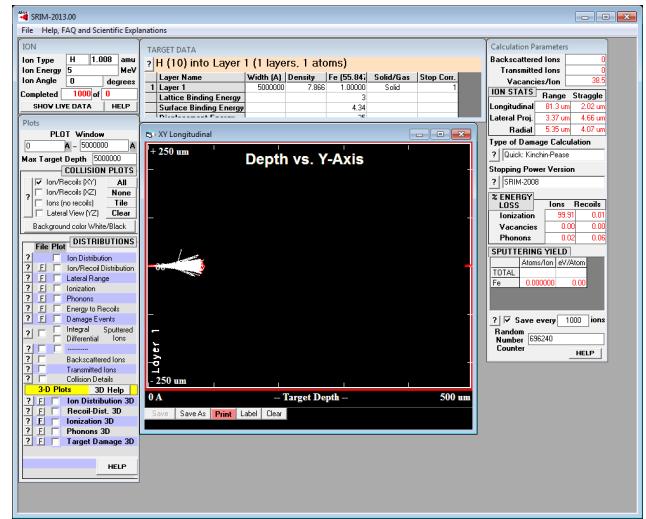


Figure H.4: SRIM output summary

H.1.2 10MeV Proton Beam - 60 microamp 100DPA

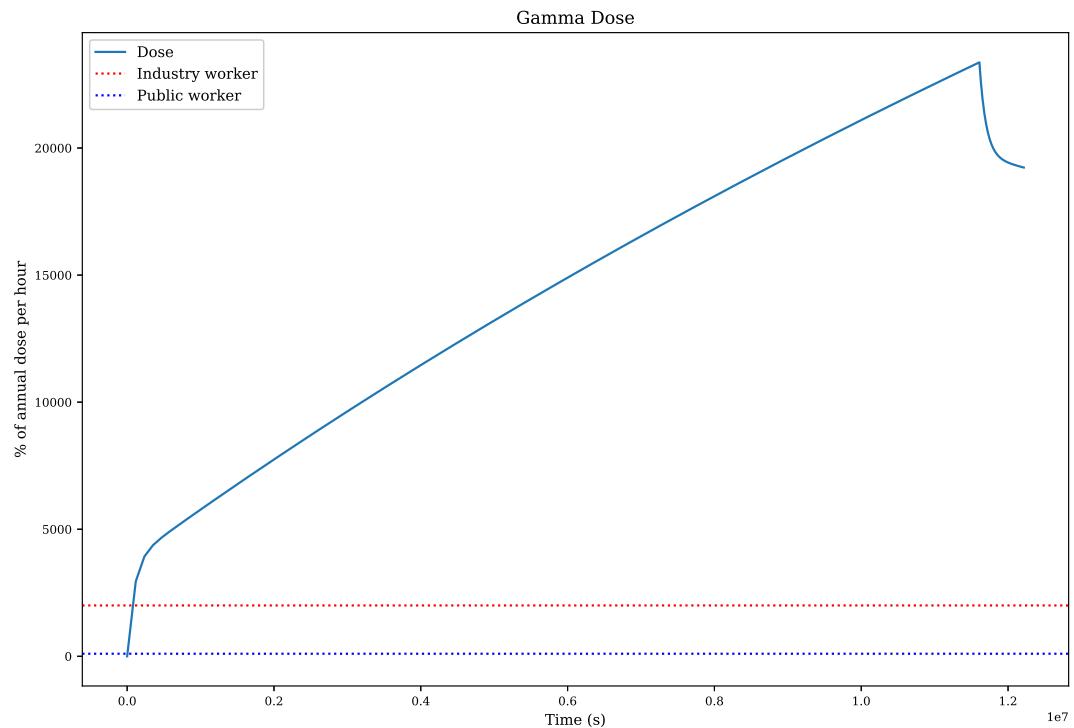


Figure H.5: Percentage of Annual Dose Equivalent

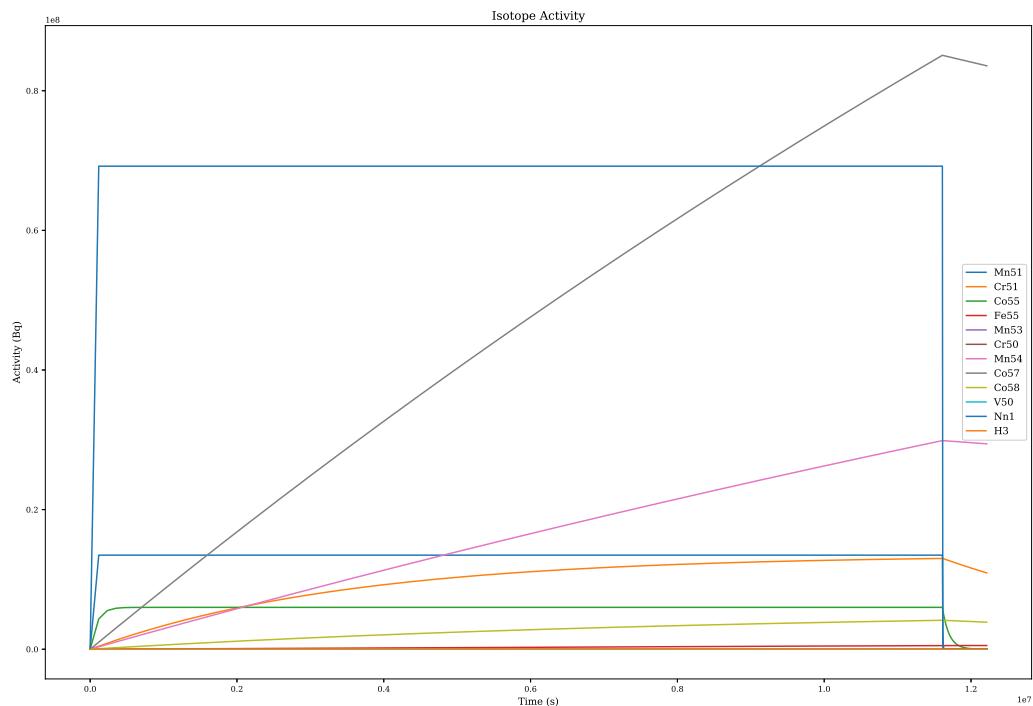


Figure H.6: Activity of all isotopes over time

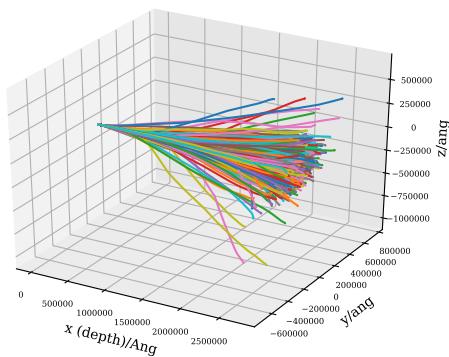


Figure H.7: Ion trajectory in the target

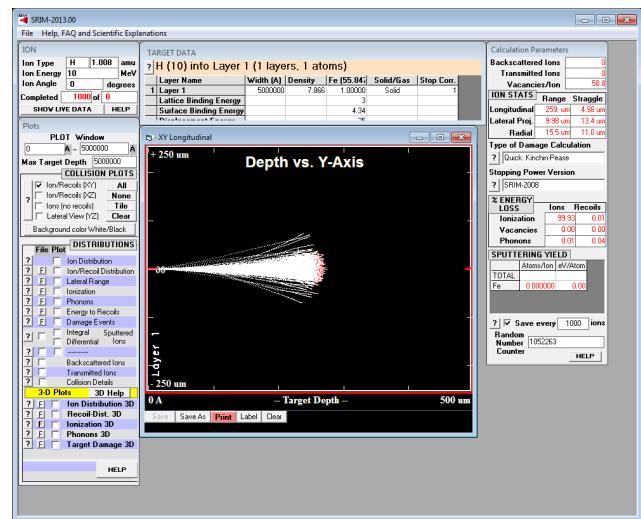


Figure H.8: SRIM output summary

H.1.3 15MeV Proton Beam - 60 microamp 100DPA

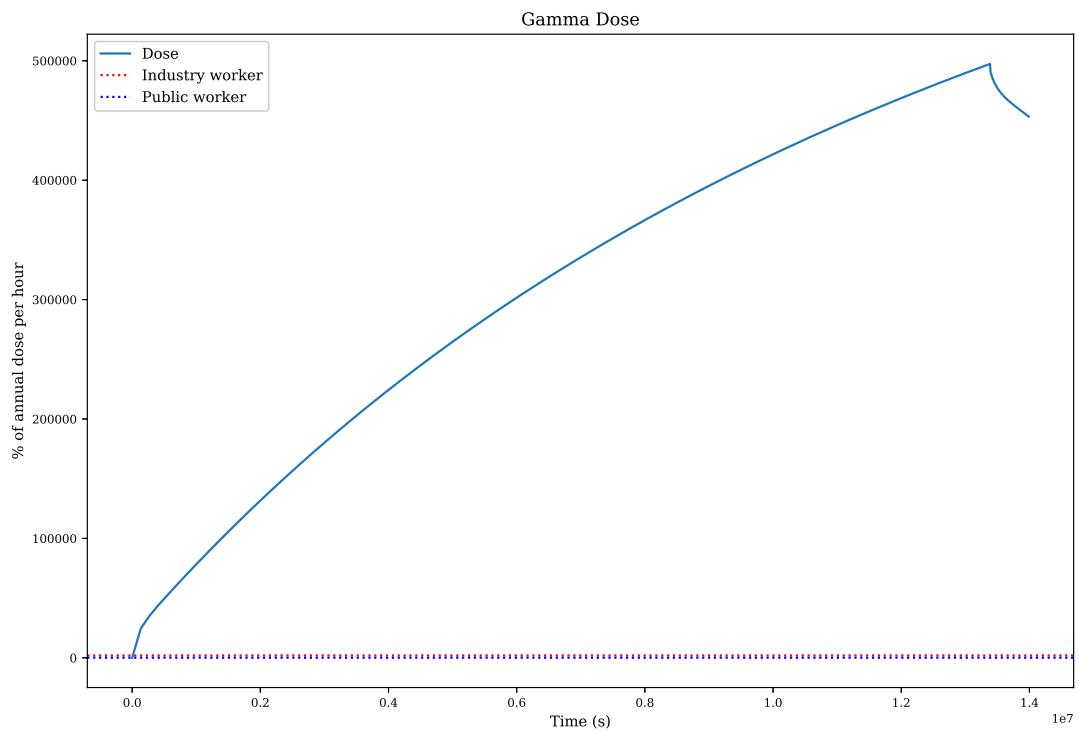


Figure H.9: Percentage of Annual Dose Equivalent

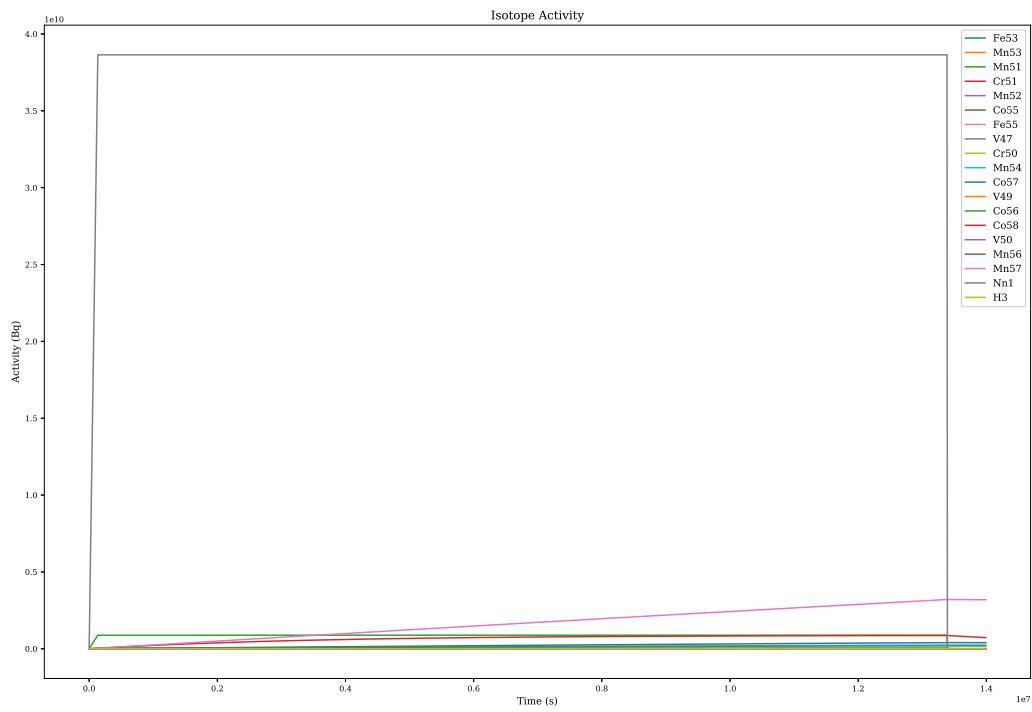


Figure H.10: Activity of all isotopes over time

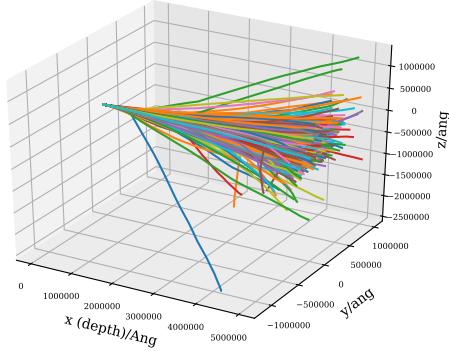


Figure H.11: Ion trajectory in the target

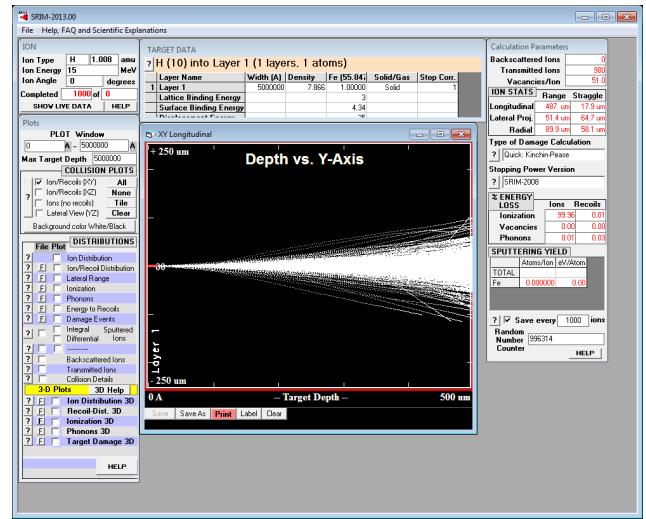


Figure H.12: SRIM output summary

H.1.4 20MeV Proton Beam - 60 microamp 100DPA

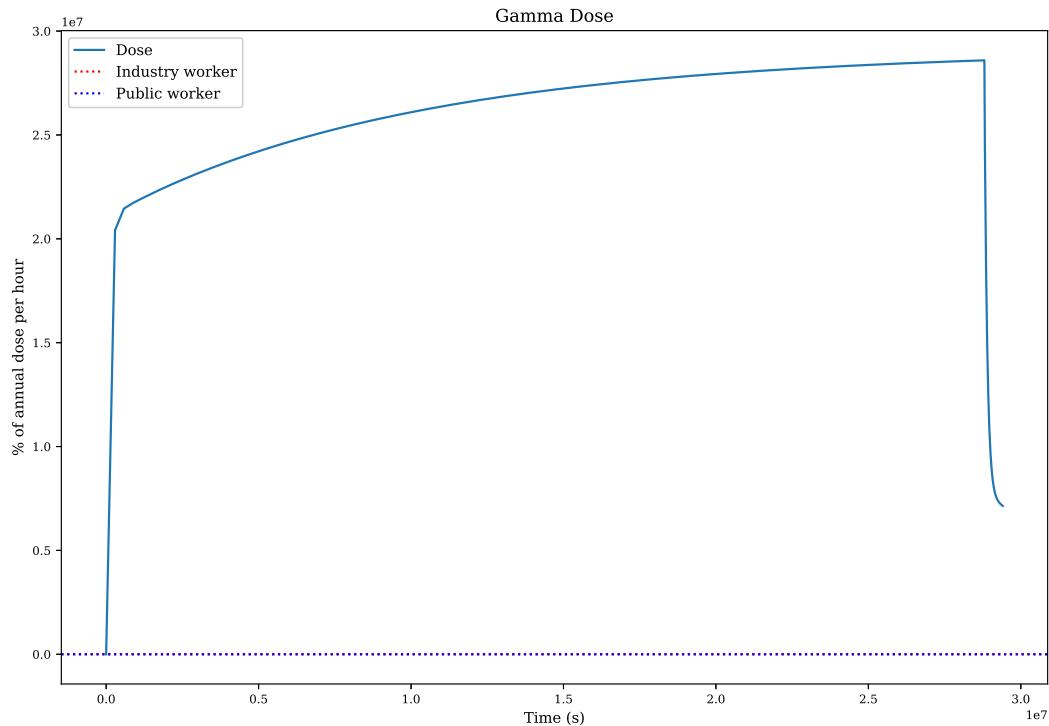


Figure H.13: Percentage of Annual Dose Equivalent

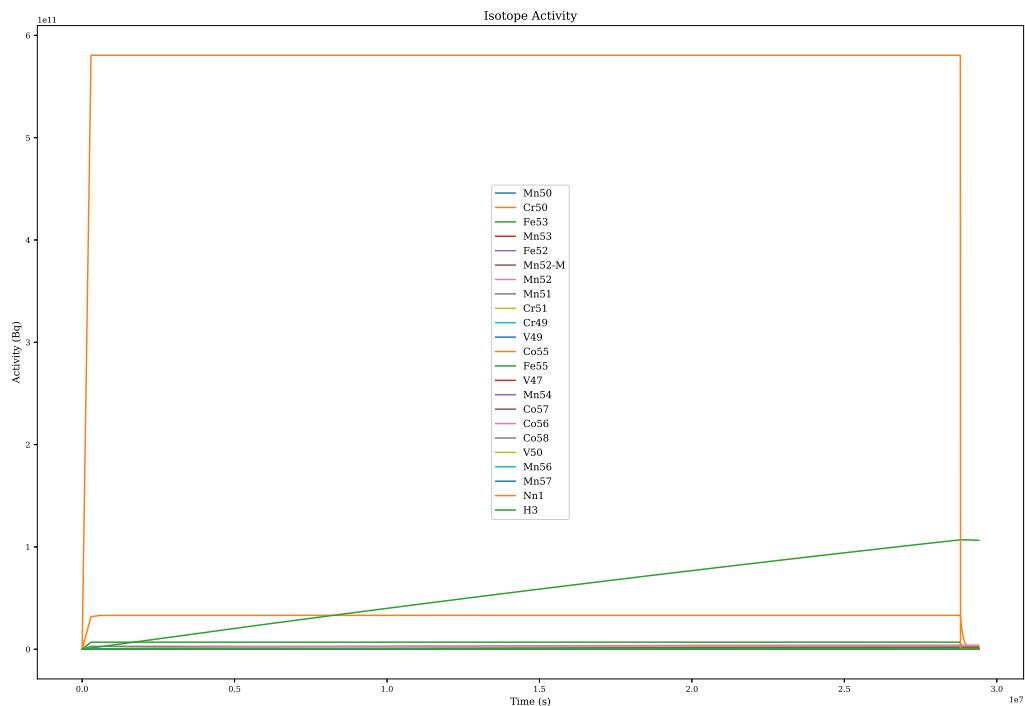


Figure H.14: Activity of all isotopes over time

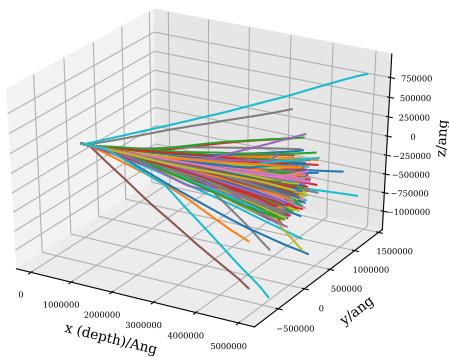


Figure H.15: Ion trajectory in the target

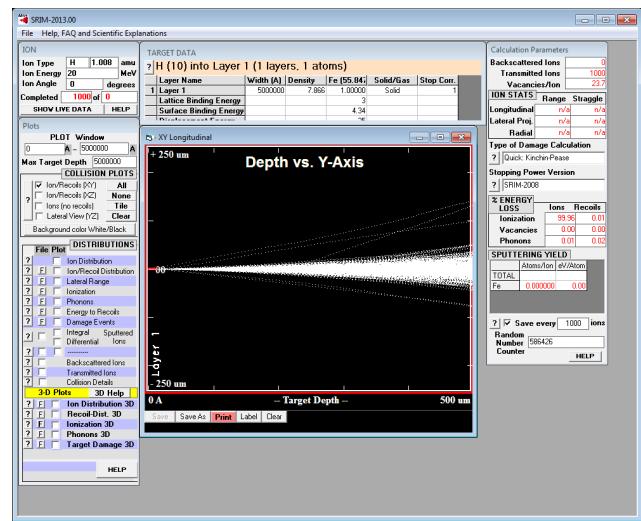


Figure H.16: SRIM output summary

H.1.5 25MeV Proton Beam - 60 microamp 100DPA

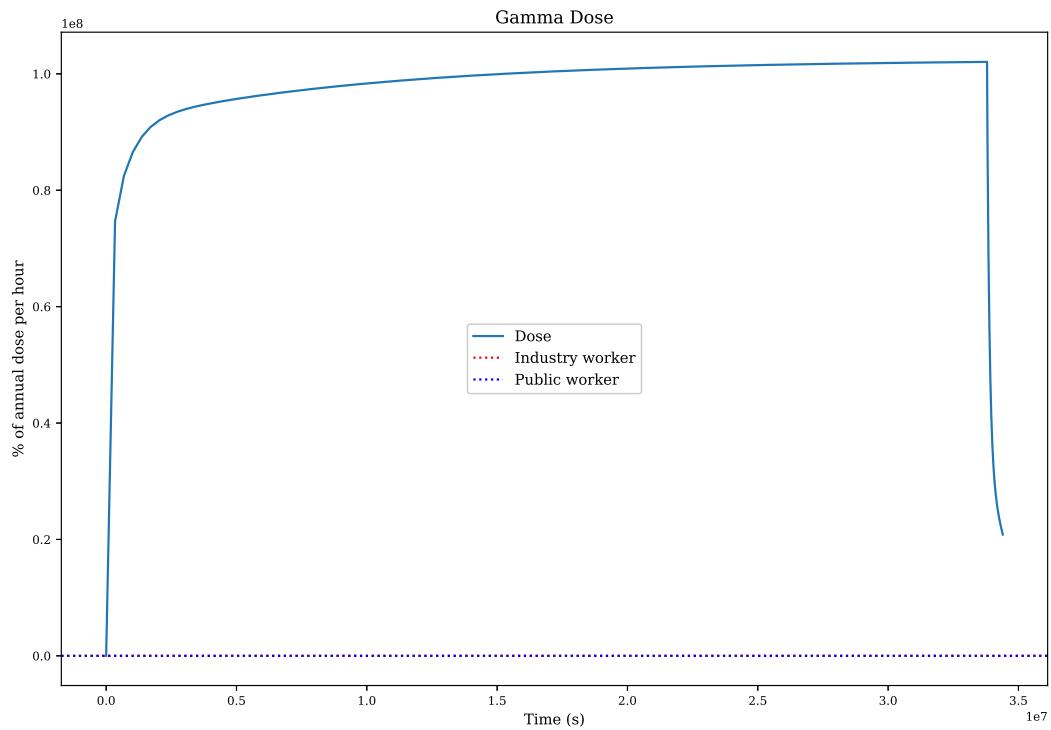


Figure H.17: Percentage of Annual Dose Equivalent

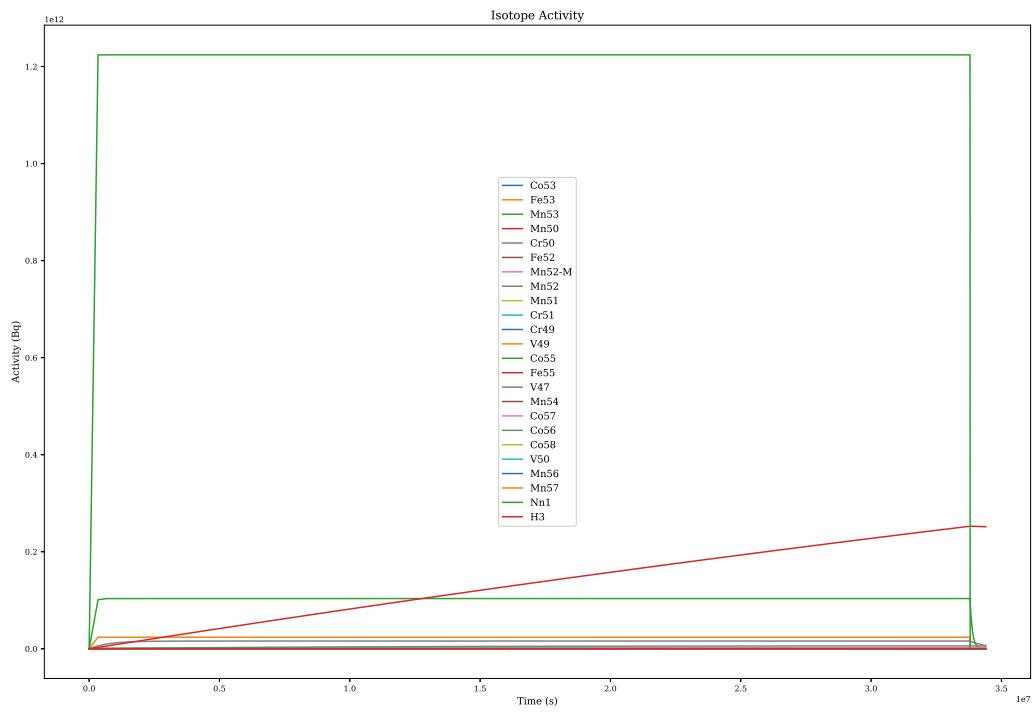


Figure H.18: Activity of all isotopes over time

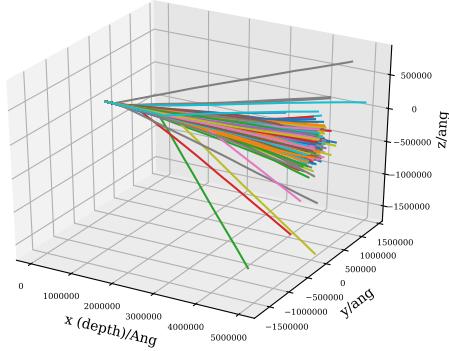


Figure H.19: Ion trajectory in the target

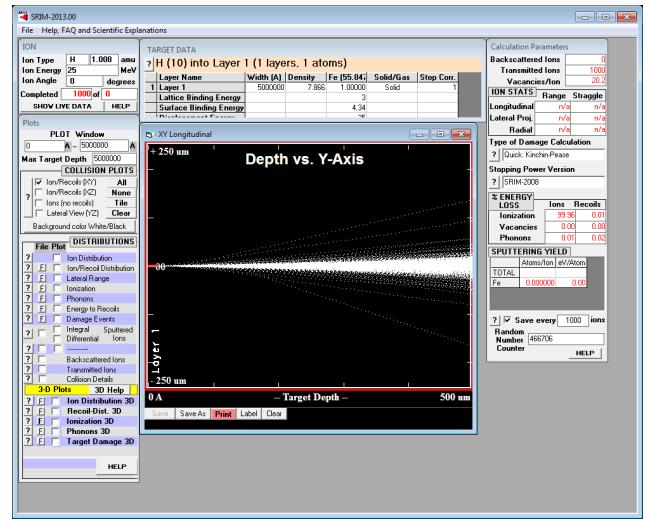


Figure H.20: SRIM output summary

H.1.6 Results Summary

	5MeV	10MeV	15MeV	20MeV	25MeV
VPI	38.5	58.8	51.0	23.7	20.2
DPA/year	178	272	236	109	93
Days for 100 DPA	205	134	155	333	391
Dose	23	234	4970	286,000	1,020,000
Dose (1 week of cooling)	18	192	4530	71,400	208,000

Table H.1: Comparison of proton irradiated iron up to 100DPA at a range of proton energies

Appendix I

DFT Convergence Plots

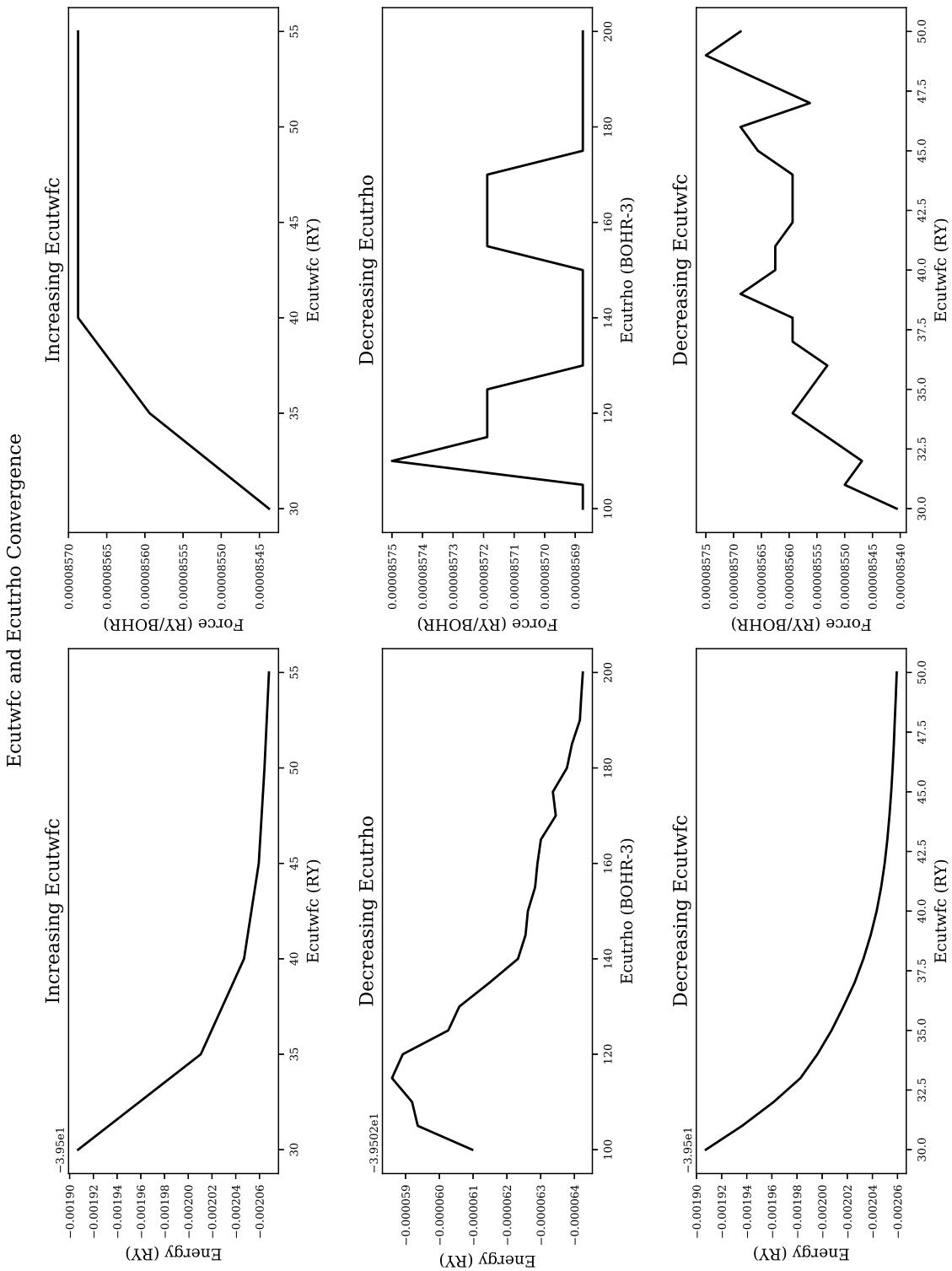


Figure I.1: Ecutwfc and Ecutfro convergence for aluminium (energy and force)

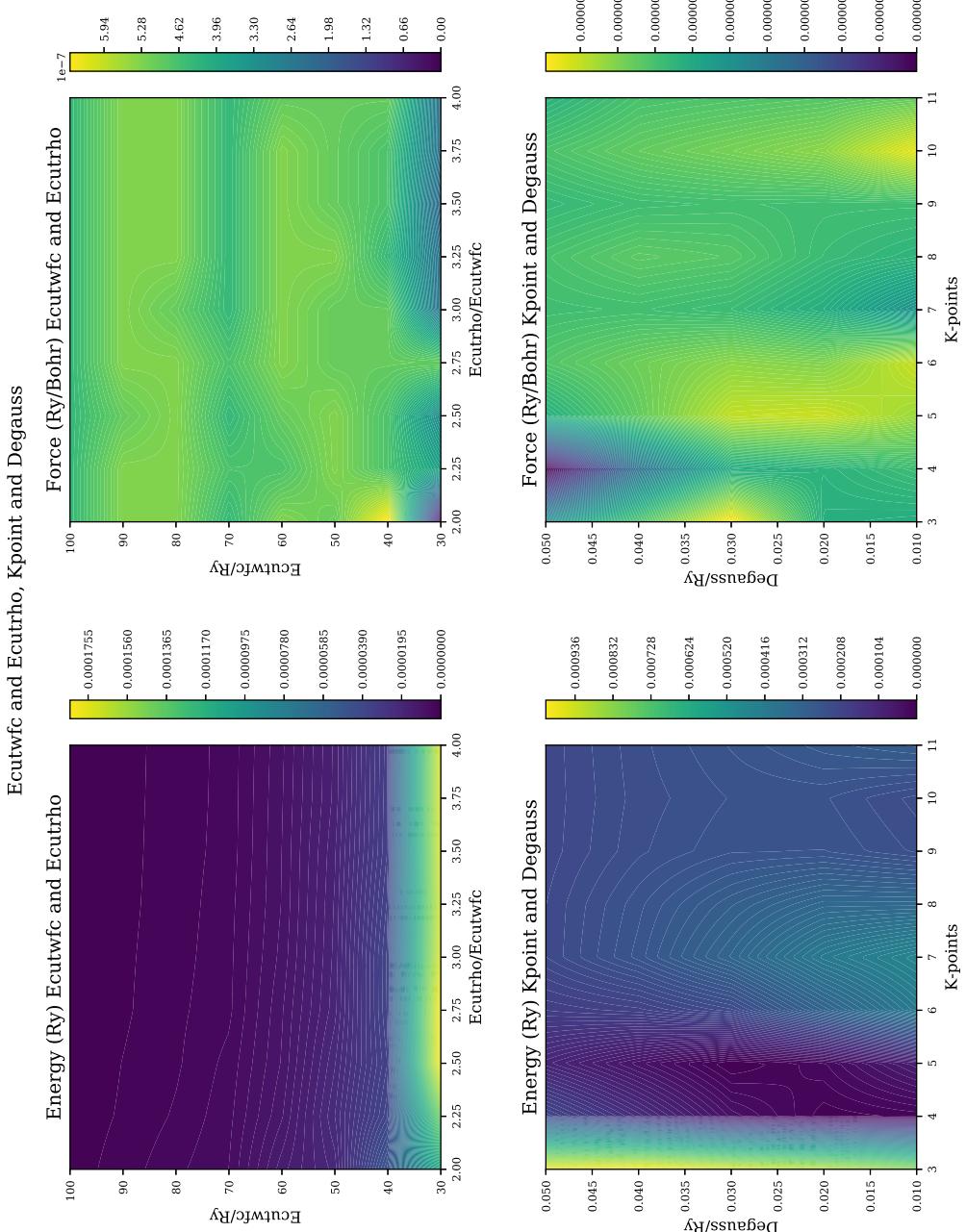


Figure I.2: 2D density map - force and energy convergence with k-points and smearing - aluminium

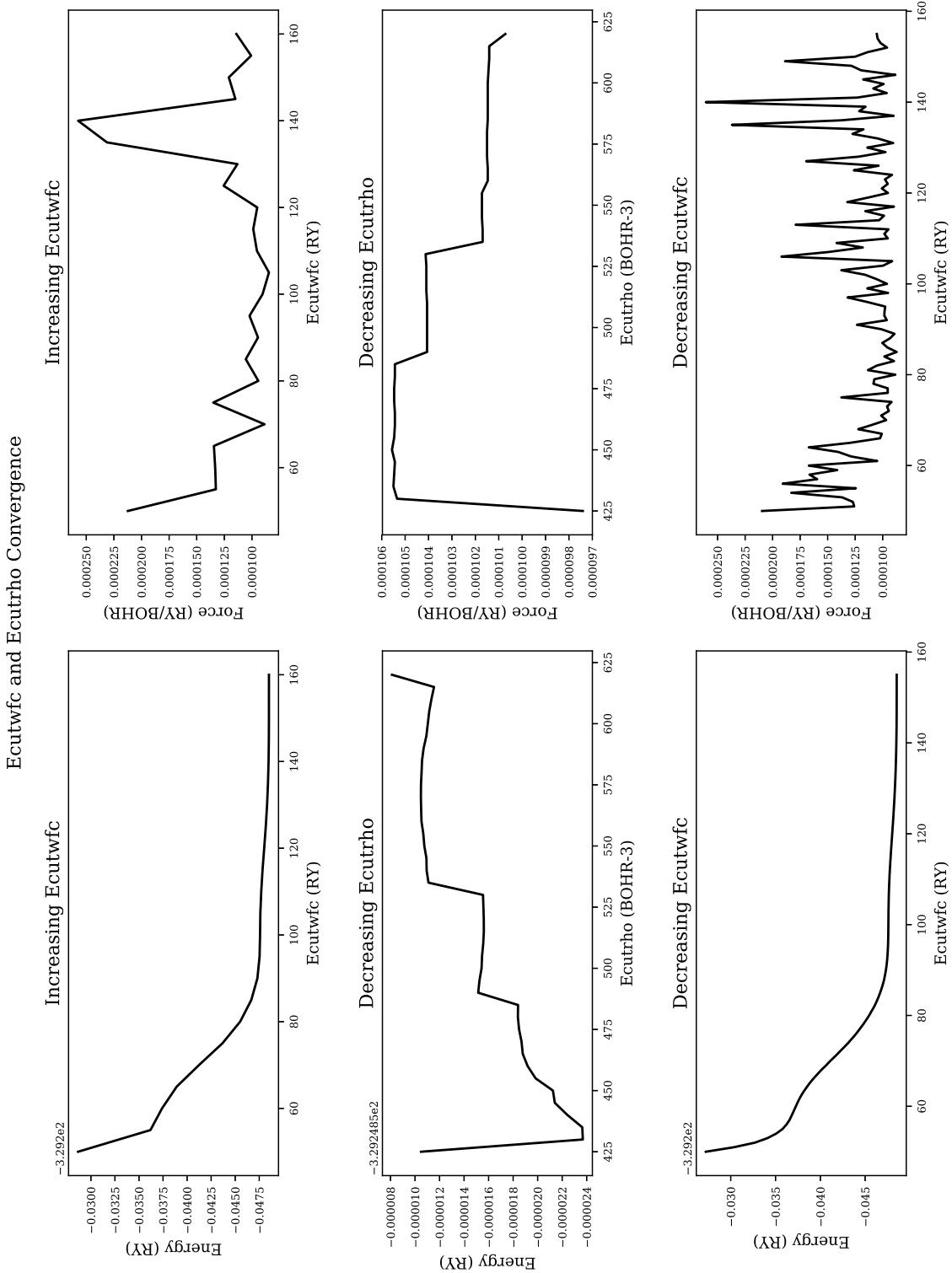


Figure I.3: Ecutwfc and Ecutfro convergence for iron (energy and force)

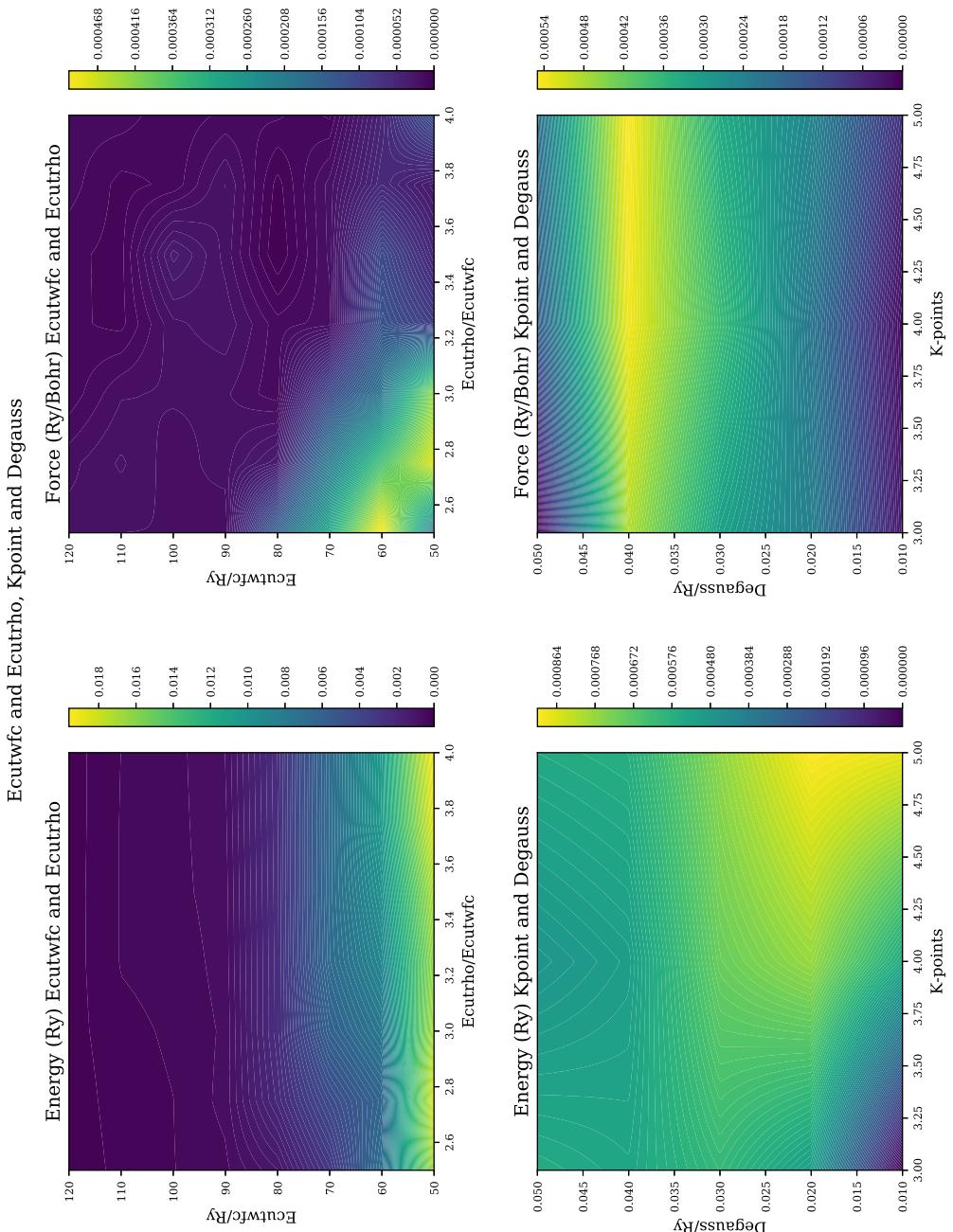


Figure I.4: 2D density map - force and energy convergence with k-points and smearing - iron

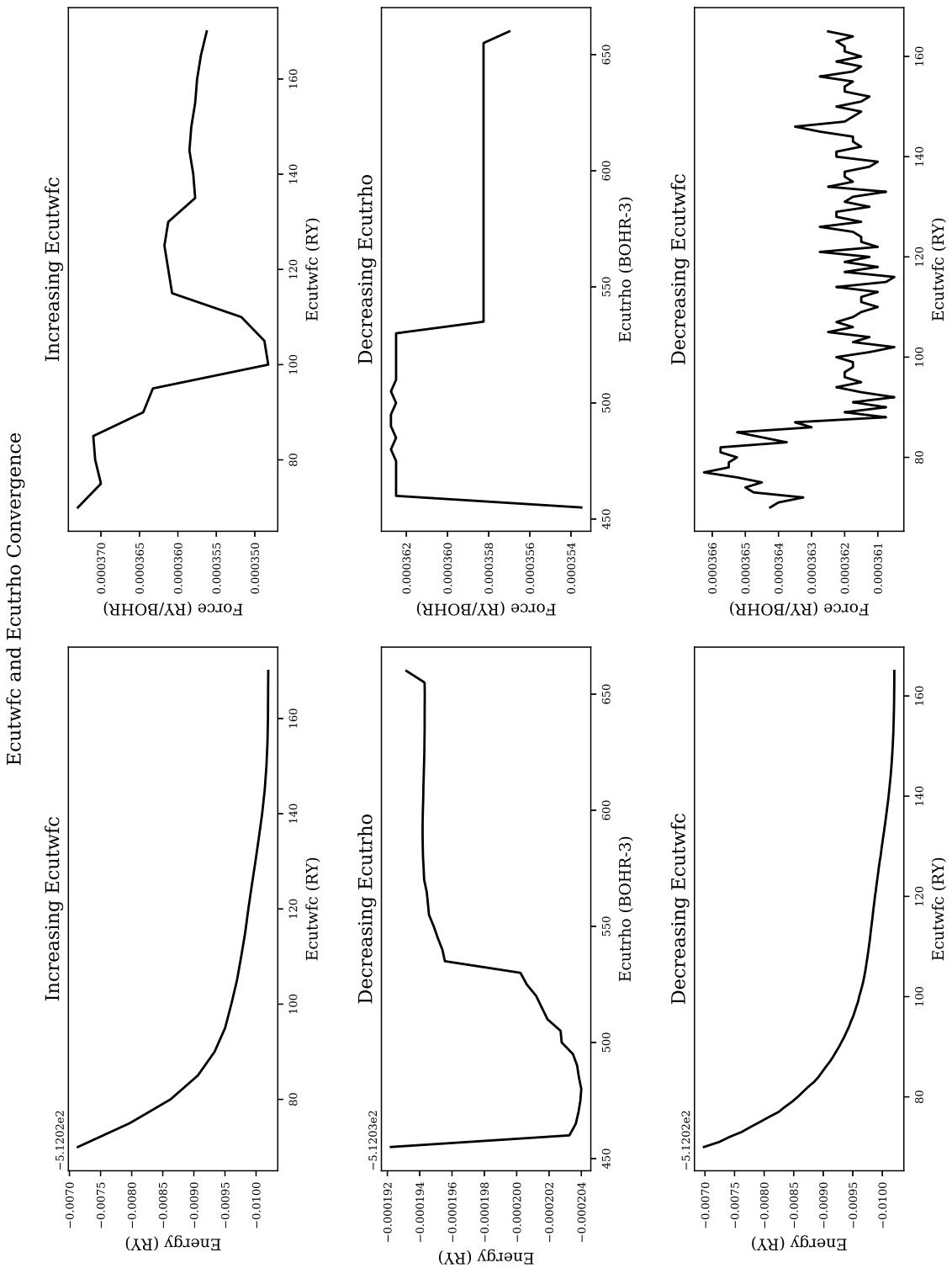


Figure I.5: Ecutwfc and Ecutfro convergence for palladium (energy and force)

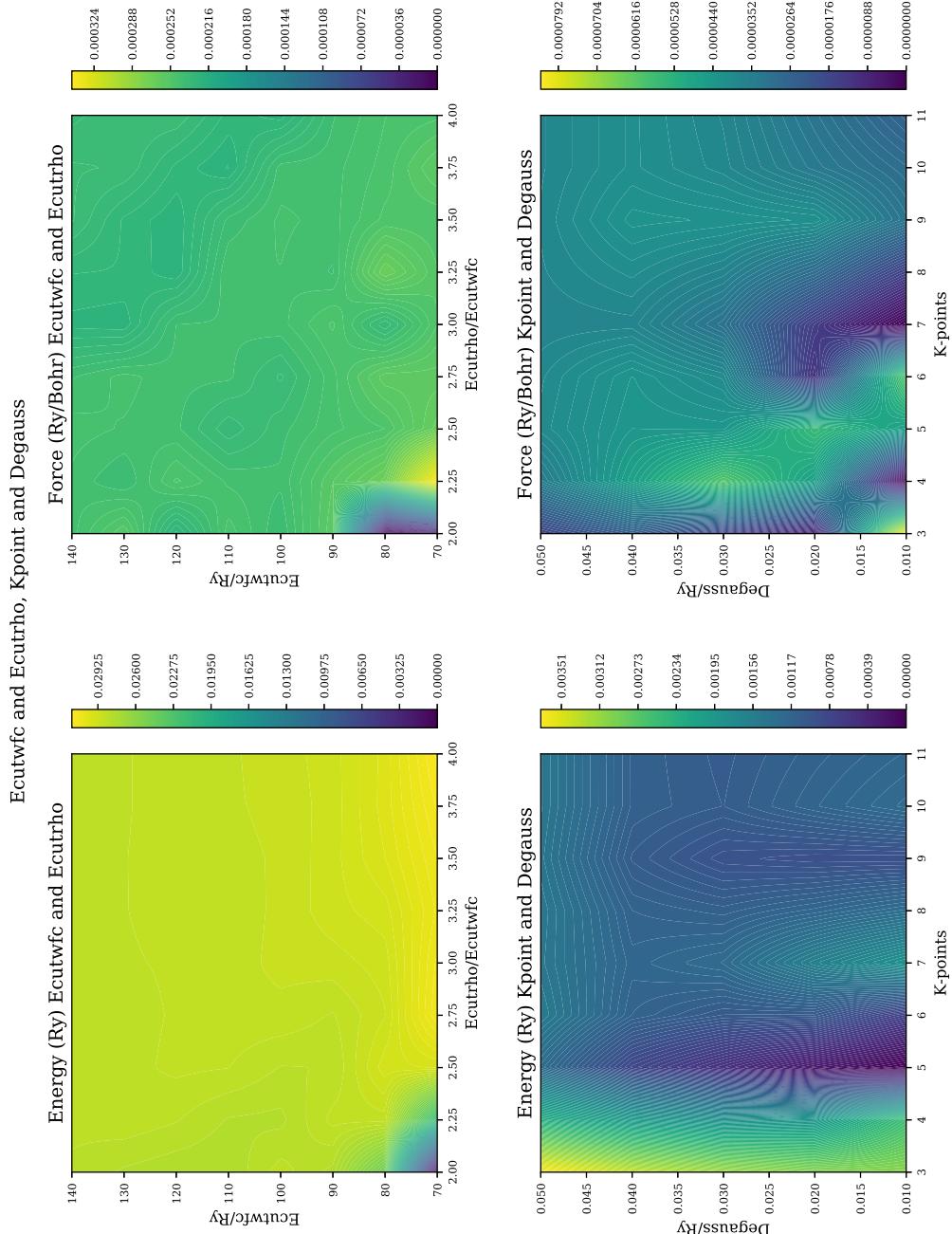


Figure I.6: 2D density map - force and energy convergence with k-points and smearing - palladium

Appendix J

DFT Calculated Properties

J.1 QE EoS

These results were calculated using the QE EoS python program that automatically creates and runs PWscf input files, collects the output files and processes the data.

J.2 FCC Aluminium

J.2.1 DFT Settings for PWscf

The calculations were performed with the Quantum Espresso PWscf package along with the qeconverge python program.

Ecutwfc: 50

Ecutrho: 200

Smearing: 0.04

K-points: 11 11 11 1 1 1

Nspin: 1 (non-polarized calculation)

Pseudopotential: Al.pbe-nl-kjpaw-psl.1.0.0.UPF

J.2.2 Equation of State and Elastic Constants

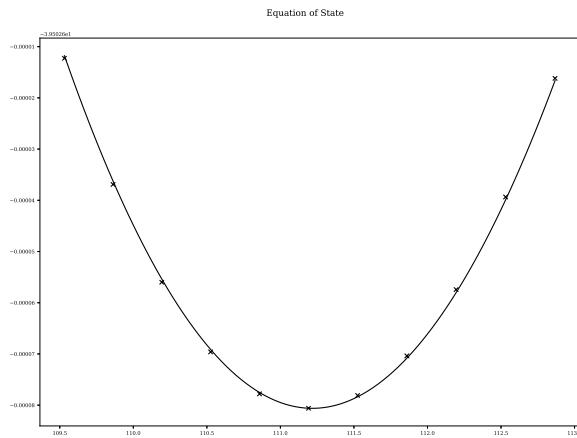


Figure J.1: Aluminium FCC equation of state plot

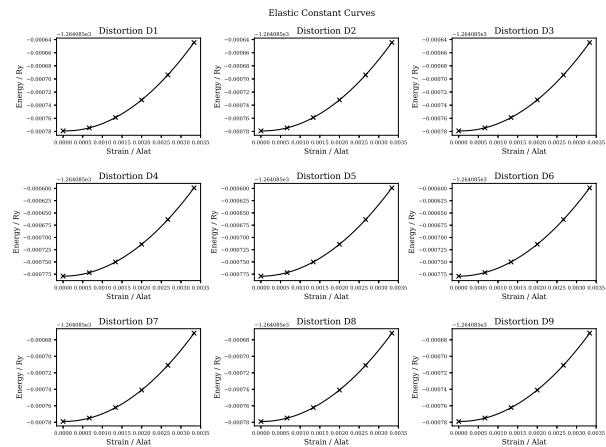


Figure J.2: Aluminium FCC elastic constants plot

J.2.3 Aluminium FCC Results File

Listing J.1: Aluminium FCC DFT Calculated Properties

```
1 ######
2 #          RESULTS          #
3 #####
4 Thu, 26 Mar 2020 13:32:45 +0000
5
6 INPUT SETTINGS
7
8 Structure:           fcc
9 Size:                2
10 Alat (Bohr):        6.9
11 Alat (Bohr) (exp.): 13.8
12 Alat (Ang):         3.6501000000000006
13 Alat (Ang) (exp.):  7.300200000000001
14 Atoms in crystal:   32
15 Atoms in crystal (exp.): 32
16 AMU per crystal:    863.423999999994
17
18 Ecutfwfc:           50
19 Ecutrho:             200
20 Degauss:             0.04
21 Kpoints:             automatic 11 11 11 1 1 1
22
23 EOS Points:          11
24 Total Strain:         0.005
25
26 V0 (Bohr^3):         111.192295309
27 E0:                  -39.5027342866
28 B0 (RY/BOHR3):        0.00531912635008
29 B0 (GPA):              77.572208693
30 BOP:                 2.0
31
32 EC Points:            9
33 Total Strain:          0.005
34
35
36 Relaxed alat (Bohr):  7.6325799
37 Relaxed alat (Bohr) (exp): 15.2651598
38 Relaxed alat (Ang):     4.0376347671
39 Relaxed alat (Ang) (exp): 8.0752695342
40 Relaxed cp:             1.0   0.0   0.0
41                      0.0   1.0   0.0
42                      0.0   0.0   1.0
43 Relaxed volume (Bohr^3): 3557.16544569
44 Relaxed density (kgm^-3): 2722.71416748
45
46 Stiffness (RY/BOHR3):   0.0076028 0.0039542 0.0039563 0.0   0.0   0.0
47                      0.0039542 0.0075973 0.003945 0.0   0.0   0.0
48                      0.0039563 0.003945 0.0076068 0.0   0.0   0.0
49                      0.0   0.0   0.0   0.0023326 0.0   0.0
50                      0.0   0.0   0.0   0.0   0.0023349 0.0
51                      0.0   0.0   0.0   0.0   0.0   0.0023339
52
53 Stiffness (GPA):       110.877131 57.6666837 57.6978857 0.0   0.0   0.0
54                      57.6666837 110.796070 57.5331112 0.0   0.0   0.0
55                      57.6978857 57.5331112 110.934314 0.0   0.0   0.0
56                      0.0   0.0   0.0   34.0184756 0.0   0.0
57                      0.0   0.0   0.0   0.0   34.0513488 0.0
58                      0.0   0.0   0.0   0.0   0.0   34.037456
```

59
60 Compliance (1/GPA): 0.0140133 -0.0048022 -0.0047979 0.0 0.0 0.0
61 -0.0048022 0.0139977 -0.0047619 0.0 0.0 0.0
62 -0.0047979 -0.0047619 0.0139794 0.0 0.0 0.0
63 0.0 -0.0 -0.0 0.0293958 -0.0 -0.0
64 0.0 0.0 0.0 0.0 0.0293674 0.0
65 0.0 0.0 0.0 0.0 0.0 0.0293794
66
67 Bulk Modulus B (GPA): 75.378064563
68 BR (GPA): 75.3780315039
69 BV (GPA): 75.378097622
70
71 Shear Modulus G (GPA): 30.8455908955
72 GR: 30.6224032645
73 GV: 31.0687785264
74
75 Young's Modulus E (GPA): 81.4294731854
76
77 Poisson's Ratio v: 0.319953205977
78
79
80 L Elastic Wave V: 0.206857953612
81 T Elastic Wave V: 0.106437709845
82 M Elastic Wave V: 0.119193708075
83
84 Debye Temperature: 0.00139515659606
85
86 Melting Point: 1300.0K

J.3 BCC Iron [No Magnetism]

J.3.1 Major DFT Settings

Ecutwfc: 71

Ecutrho: 430

Smearing: 0.04

K-points: 9 9 9 1 1 1

Nspin: 1 (non-polarized calculation)

Pseudopotential: Fe.pbe-spn-kjpaw-psl.1.0.0.UPF

J.3.2 Equation of State and Elastic Constants

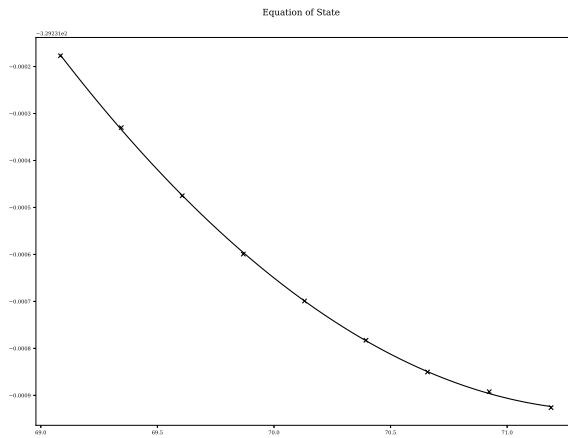


Figure J.3: Iron BCC equation of state plot

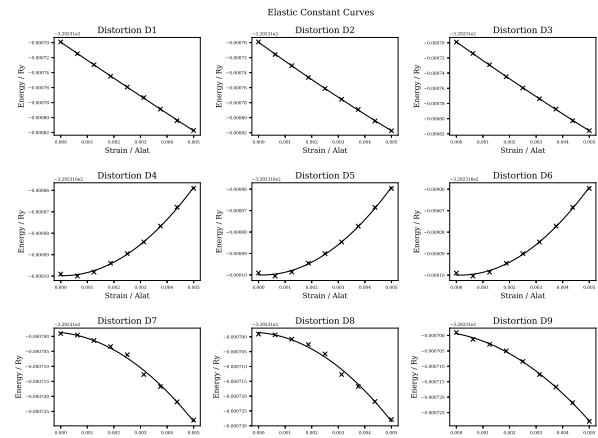


Figure J.4: Iron BCC elastic constants plot

J.3.3 Iron BCC Results File

Listing J.2: Iron BCC DFT Calculated Properties

```
1 #####  
2 # RESULTS #  
3 #####  
4 Tue, 03 Nov 2020 06:01:22 +0000  
5  
6 INPUT SETTINGS  
7 #####  
8  
9 Structure: bcc  
10 Size: 2  
11 AO: 2.8  
12 AO Units: ang  
13 AO (Bohr): 10.584  
14 AO (Angs): 5.598936  
15  
16 Ecutwfc: 71  
17 Ecutrho: 430  
18 Degauss: 0.04  
19 Kpoints: 9 9 9 1 1 1  
20 Kpoints Type: automatic  
21 EOS Points: 9  
22 EOS Strain: 0.005  
23 EC Points: 9  
24 EC Strain: 0.005  
25  
26  
27 RELAXED  
28 #####  
29  
30 a0 (Bohr): 10.391466456  
31 CP: 1.0 0.0 0.0  
32 0.0 1.0 0.0  
33 0.0 0.0 1.0  
34 Volume (Bohr^3/unit cell): 1122.09730705  
35 Density KG/m^3: 8932.14485799  
36 AMU per crystal unit: 893.5200000000003  
37 Atoms per crystal unit: 16  
38 Total Energy/Ry: -5267.70721439  
39 Energy per Atom/Ry: -329.231700899375  
40 Total Force Ry/Bohr: 1.7e-05  
41 Force per atom Ry/Bohr: 1.0625e-06  
42  
43  
44 a0 primitive (Bohr): 5.195733228  
45 a0 primitive (Ang): 2.748542877612  
46  
47  
48 EOS  
49 #####  
50  
51 V0 (Bohr^3 / atom): 71.4368199046  
52 E0 (Ry / atom): -329.231932186  
53 B0 (RY/BOHR3): 0.0195284174344  
54 B0 (GPA): 287.272784669  
55 BOP: 4.98933790197  
56  
57  
58 EC
```

```

59 #####
60
61 Stiffness (RY/BOHR3):   0.0043229 0.0225755 0.0216226 0.0      0.0      0.0
62                      0.0225755 0.0121066 0.0217004 0.0      0.0      0.0
63                      0.0216226 0.0217004 0.0083439 0.0      0.0      0.0
64                      0.0      0.0      0.0      0.0121324 0.0      0.0
65                      0.0      0.0      0.0      0.0      0.0123943 0.0
66                      0.0      0.0      0.0      0.0      0.0      0.0123441
67
68 Stiffness (GPA):      63.04417 329.233427 315.336007 0.0      0.0      0.0
69                      329.233427 176.558337 316.470988 0.0      0.0      0.0
70                      315.336007 316.470988 121.684505 0.0      0.0      0.0
71                      0.0      0.0      0.0      176.934632 0.0      0.0
72                      0.0      0.0      0.0      0.0      180.753497 0.0
73                      0.0      0.0      0.0      0.0      0.0      180.022531
74
75 Compliance (1/GPA): -0.0026218 0.0019907 0.001617 0.0      0.0      0.0
76                      0.0019907 -0.0030583 0.0027951 0.0      0.0      0.0
77                      0.001617 0.0027951 -0.0032415 0.0      0.0      0.0
78                      0.0      0.0      0.0      0.0056518 0.0      0.0
79                      0.0      0.0      0.0      0.0      0.0055324 0.0
80                      0.0      0.0      0.0      0.0      0.0      0.0055549
81
82 Stability:
83 C11:                  63.044169998 (Stable)
84 C11C22 - C12C12:     -97263.6756949 (Unstable)
85 C11*C22*C33+2*C12*C13*C23-C11*C23*C23-C33*C12*C12: 47561899.9206 (Stable)
86 C44:                  176.934632292 (Stable)
87 C55:                  180.753497554 (Stable)
88 C66:                  180.022531471 (Stable)
89
90 Bulk Modulus B (GPA): 255.592980231
91 BR (GPA):             257.478420517
92 BV (GPA):             253.707539944
93
94 Shear Modulus G (GPA): -643.096162505
95 GR:                   -1353.7508966
96 GV:                   67.5585715904
97
98 Young's Modulus E (GPA): -11960.7418627
99
100 Poisson's Ratio v:    8.29934165372
101
102
103 L Elastic Wave V:    nan
104 T Elastic Wave V:    nan
105 M Elastic Wave V:    nan
106
107 Debye Temperature:   nan
108
109 Melting Point:       1357.0K
110
111
112
113
114
115
116
117
118 References
119 =====

```

120
121 First Principles Calculations of Elastic Properties of Metals
122 M. J. Mehl, B. M. Klein, D. A. Papaconstantopoulos
123 1994
124
125 Ab Initio Study of the Elastic and Mechanical Properties of B19 TiAl
126 Y. Wen, L. Wang, H. Liu and L. Song
127 Crystals
128 2017
129
130 Density functional theory for calculation of elastic properties of orthorhombic crystals - applications to TiSi₂
131 P. Ravindran, Lars Fast, P. A. Korzhavyi, B. Johansson
132 Journal of Applied Physics
133 1998

J.4 BCC Iron [Ferromagnetic]

J.4.1 Major DFT Settings

Ecutwfc: 71

Ecutrho: 430

Smearing: 0.04

K-points: 9 9 9 1 1 1

Nspin: 2 (Collinear Spin)

Pseudopotential: Fe.pbe-spn-kjpaw-psl.1.0.0.UPF

J.4.2 Equation of State and Elastic Constants

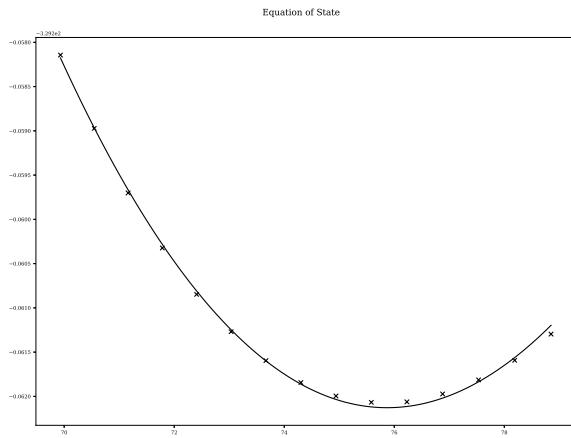


Figure J.5: Iron BCC (magnetic) equation of state plot

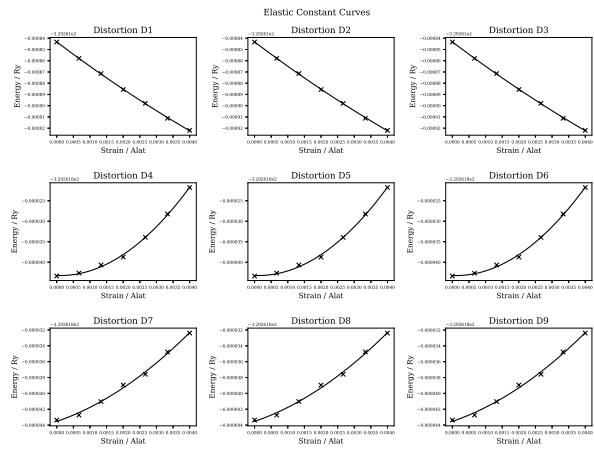


Figure J.6: Iron BCC (magnetic) elastic constants plot

J.4.3 Iron Ferromagnetic BCC Results File

Listing J.3: Iron BCC DFT calculated properties

```
1 ######
2 #          RESULTS          #
3 #####
4
5 INPUT SETTINGS
6 #####
7
8 Structure:           bcc
9 Size:                2
10 AO:                 2.8
11 AO Units:           ang
12 AO (Bohr):          10.584
13 AO (Angs):          5.598936
14
15 Ecutfc:              71
16 Ecutrho:             430
17 Degauss:              0.04
18 Kpoints:              9 9 9 1 1 1
19 Kpoints Type:         automatic
20 EOS Points:           15
21 EOS Strain:            0.02
22 EC Points:             7
23 EC Strain:             0.004
24
25
26 RELAXED
27 #####
28
29 a0 (Bohr):           10.5935195777
30 CP:                  1.0   0.0   0.0
31                      0.0   1.0   0.0
32                      0.0   0.0   1.0
33 Volume (Bohr^3/unit cell): 1188.83291445
34 Density KG/m^3:        8430.73536197
35 AMU per crystal unit:  893.5200000000003
36 Atoms per crystal unit: 16
37 Total Energy/Ry:       -5268.18846365
38 Energy per Atom/Ry:    -329.261778978125
39 Total Force Ry/Bohr:   2.2e-05
40 Force per atom Ry/Bohr: 1.375e-06
41
42
43 a0 primitive (Bohr):   5.2967597888519995
44 a0 primitive (Ang):     2.8019859283027078
45
46
47 EOS
48 #####
49
50 V0 (Bohr^3 / atom):    75.8702665574
51 E0 (Ry / atom):         -329.262127846
52 B0 (RY/BOHR3):          0.0162627705061
53 B0 (GPA):               239.23348553
54 BOP:                   3.31889649178
55
56
57 EC
58 ######
```

59
 60 Stiffness (RY/BOHR3): 0.0171516 0.0124976 0.0125676 0.0 0.0 0.0
 61 0.0124976 0.0170822 0.0125444 0.0 0.0 0.0
 62 0.0125676 0.0125444 0.0172259 0.0 0.0 0.0
 63 0.0 0.0 0.0 0.0095442 0.0 0.0
 64 0.0 0.0 0.0 0.0 0.0095793 0.0
 65 0.0 0.0 0.0 0.0 0.0 0.0095537
 66
 67 Stiffness (GPA): 250.132354 182.259753 183.281653 0.0 0.0 0.0
 68 182.259753 249.120311 182.943207 0.0 0.0 0.0
 69 183.281653 182.943207 251.216687 0.0 0.0 0.0
 70 0.0 0.0 0.0 139.188774 0.0 0.0
 71 0.0 0.0 0.0 0.0 139.701368 0.0
 72 0.0 0.0 0.0 0.0 0.0 139.328420
 73
 74 Compliance (1/GPA): 0.0104302 -0.0043908 -0.0044121 0.0 0.0 0.0
 75 -0.0043908 0.0104768 -0.0044261 0.0 0.0 0.0
 76 -0.0044121 -0.0044261 0.0104228 0.0 0.0 0.0
 77 0.0 0.0 0.0 0.0071845 0.0 0.0
 78 0.0 0.0 0.0 0.0 0.0071581 0.0
 79 0.0 0.0 0.0 0.0 0.0 0.0071773
 80
 81 Stability:
 82 C11: 250.132354807 (Stable)
 83 C11C22 - C12C12: 29094.4322943 (Stable)
 84 C11*C22*C33+2*C12*C13*C23-C11*C23*C23-C33*C12*C12: 11159910.6986 (Stable)
 85 C44: 139.188774163 (Stable)
 86 C55: 139.701368243 (Stable)
 87 C66: 139.328420919 (Stable)
 88
 89 Bulk Modulus B (GPA): 205.266912094
 90 BR (GPA): 205.262870468
 91 BV (GPA): 205.270953721
 92
 93 Shear Modulus G (GPA): 79.4449473647
 94 GR: 61.7805328168
 95 GV: 97.1093619125
 96
 97 Young's Modulus E (GPA): 211.100586016
 98
 99 Poisson's Ratio v: 0.328596676183
 100
 101
 102 L Elastic Wave V: 0.192124405208
 103 T Elastic Wave V: 0.0970734382597
 104 M Elastic Wave V: 0.108830261337
 105
 106 Debye Temperature: 0.000728462251696
 107
 108 Melting Point: 2076.0K
 109
 110
 111
 112
 113
 114
 115
 116
 117 References
 118 ======
 119

120 First Principles Calculations of Elastic Properties of Metals
121 M. J. Mehl, B. M. Klein, D. A. Papaconstantopoulos
122 1994
123
124 Ab Initio Study of the Elastic and Mechanical Properties of B19 TiAl
125 Y. Wen, L. Wang, H. Liu and L. Song
126 Crystals
127 2017
128
129 Density functional theory for calculation of elastic properties of orthorhombic crystals - applications to TiSi₂
130 P. Ravindran, Lars Fast, P. A. Korzhavyi, B. Johansson
131 Journal of Applied Physics
132 1998

J.5 FCC Iron [Antiferromagnetic]

J.5.1 Major DFT Settings

Ecutwfc: 71

Ecutrho: 430

Smearing: 0.04

K-points: 9 9 9 1 1 1

Nspin: 2 (Collinear Spin)

Pseudopotential: Fe.pbe-spn-kjpaw-psl.1.0.0.UPF

J.5.2 Equation of State and Elastic Constants

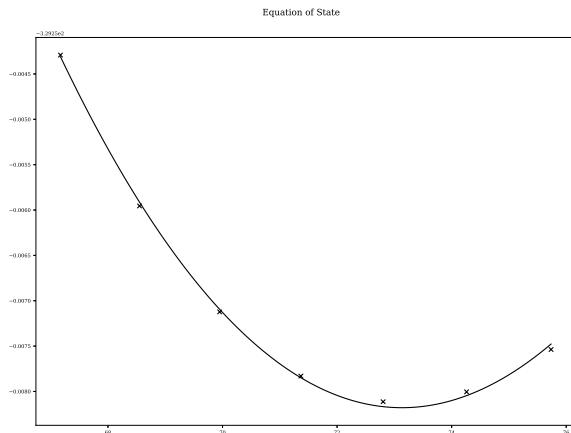


Figure J.7: Iron FCC (magnetic) equation of state plot

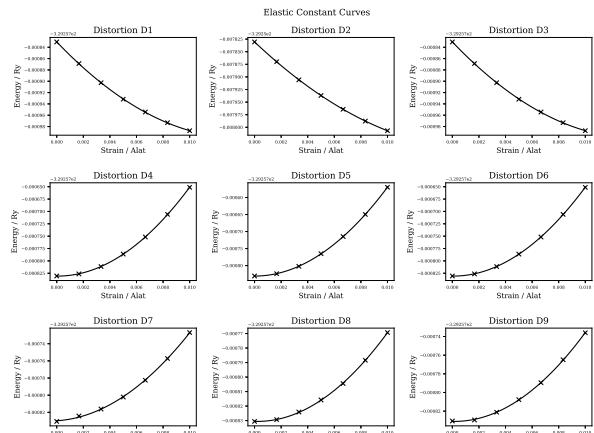


Figure J.8: Iron FCC (magnetic) elastic constants plot

J.5.3 Iron FCC Results File

Listing J.4: Iron FCC DFT Calculated Properties

```
1 ######
2 #          RESULTS          #
3 ######
4
5 INPUT SETTINGS
6 #####
7
8 Structure:           fcc
9 Size:                2
10 AO:                 3.4278156
11 AO Units:           ang
12 AO (Bohr):         12.957142968
13 AO (Angs):          6.854328630072
14
15 Ecutwfc:            71
16 Ecutrho:             430
17 Degauss:             0.04
18 Kpoints:             9 9 9 1 1 1
19 Kpoints Type:        automatic
20 EOS Points:          7
21 EOS Strain:          0.02
22 EC Points:            7
23 EC Strain:            0.01
24
25
26 RELAXED
27 #####
28
29 a0 (Bohr):          12.9381866998
30 CP:                  1.0    0.0    0.0
31                      0.0    1.05447369502   0.0
32                      0.0    0.0    0.999986980953
33 Volume (Bohr^3/unit cell): 2283.75934496
34 Density KG/m^3:        8777.4009231
35 AMU per crystal unit: 1787.040000000006
36 Atoms per crystal unit: 32
37 Total Energy/Ry:       -10536.25057837
38 Energy per Atom/Ry:     -329.2578305740625
39 Total Force Ry/Bohr:   5.1e-05
40 Force per atom Ry/Bohr: 1.59375e-06
41
42
43 a0 primitive (Bohr):   6.4690933498955
44 a0 primitive (Ang):     3.4221503820947197
45
46
47 EOS
48 #####
49
50 V0 (Bohr^3 / atom):   73.1321918692
51 E0 (Ry / atom):        -329.258178984
52 B0 (RY/BOHR3):         0.0153732026949
53 B0 (GPA):              226.147498244
54 BOP:                  3.74220549488
55
56
57 EC
58 ######
```

59
 60 Stiffness (RY/BOHR3): 0.0250005 0.009711 0.0160292 0.0 0.0 0.0
 61 0.009711 0.0204815 0.0089411 0.0 0.0 0.0
 62 0.0160292 0.0089411 0.0250019 0.0 0.0 0.0
 63 0.0 0.0 0.0 0.0127715 0.0 0.0
 64 0.0 0.0 0.0 0.0 0.0182966 0.0
 65 0.0 0.0 0.0 0.0 0.0 0.0127778
 66
 67 Stiffness (GPA): 364.598330 141.621966 233.764448 0.0 0.0 0.0
 68 141.621966 298.695202 130.394083 0.0 0.0 0.0
 69 233.764448 130.394083 364.619131 0.0 0.0 0.0
 70 0.0 0.0 0.0 186.254349 0.0 0.0
 71 0.0 0.0 0.0 0.0 266.830684 0.0
 72 0.0 0.0 0.0 0.0 0.0 186.346168
 73
 74 Compliance (1/GPA): 0.004966 -0.0011431 -0.002775 0.0 0.0 0.0
 75 -0.0011431 0.0042304 -0.00078 0.0 0.0 0.0
 76 -0.002775 -0.00078 0.0048006 0.0 0.0 0.0
 77 0.0 0.0 0.0 0.005369 0.0 0.0
 78 0.0 0.0 0.0 0.0 0.0037477 0.0
 79 0.0 0.0 0.0 0.0 0.0 0.0053664
 80
 81 Stability:
 82 C11: 364.598330439 (Stable)
 83 C11C22 - C12C12: 88846.9908456 (Stable)
 84 C11*C22*C33+2*C12*C13*C23-C11*C23*C23-C33*C12*C12: 34829887.0518 (Stable)
 85 C44: 186.254349116 (Stable)
 86 C55: 266.830684551 (Stable)
 87 C66: 186.346168623 (Stable)
 88
 89 Bulk Modulus B (GPA): 221.980853006
 90 BR (GPA): 217.35352142
 91 BV (GPA): 226.608184591
 92
 93 Shear Modulus G (GPA): 144.783544731
 94 GR: 126.872037967
 95 GV: 162.695051494
 96
 97 Young's Modulus E (GPA): 356.782113599
 98
 99 Poisson's Ratio v: 0.232122456533
 100
 101
 102 L Elastic Wave V: 0.217447522685
 103 T Elastic Wave V: 0.128433002219
 104 M Elastic Wave V: 0.142291445915
 105
 106 Debye Temperature: 0.0009653163572
 107
 108 Melting Point: 2528.0K
 109
 110
 111
 112
 113
 114
 115
 116
 117 References
 118 ======
 119

120 First Principles Calculations of Elastic Properties of Metals
121 M. J. Mehl, B. M. Klein, D. A. Papaconstantopoulos
122 1994
123
124 Ab Initio Study of the Elastic and Mechanical Properties of B19 TiAl
125 Y. Wen, L. Wang, H. Liu and L. Song
126 Crystals
127 2017
128
129 Density functional theory for calculation of elastic properties of orthorhombic crystals - applications to TiSi₂
130 P. Ravindran, Lars Fast, P. A. Korzhavyi, B. Johansson
131 Journal of Applied Physics
132 1998

J.6 FCC Palladium

J.6.1 Major DFT Settings

Ecutwfc: 71

Ecutrho: 430

Smearing: 0.04

K-points: 9 9 9 1 1 1

Nspin: 1 (non-polarized calculation)

Pseudopotential: Pd 106.42 Pd.pbe-spn-kjpaw_psl.1.0.0.UPF

J.6.2 Equation of State and Elastic Constants

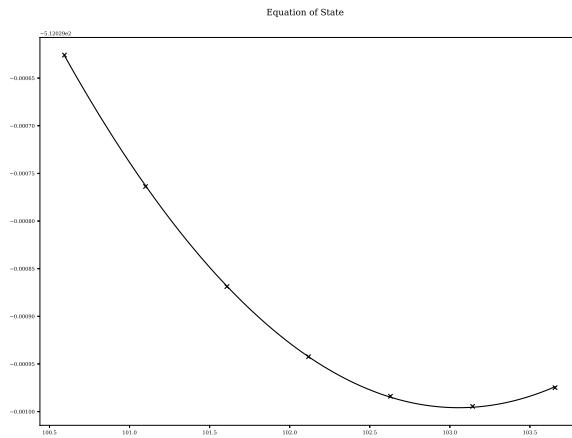


Figure J.9: Palladium FCC equation of state plot

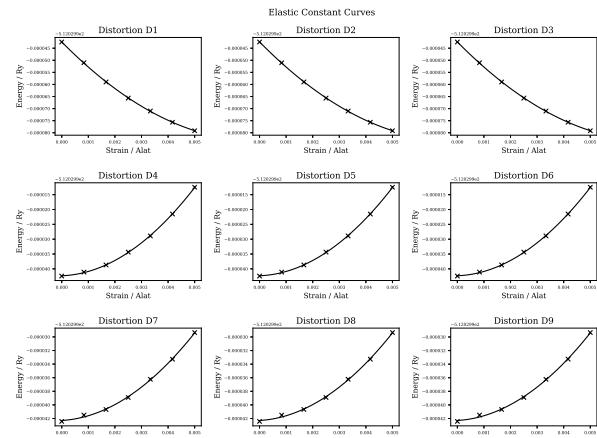


Figure J.10: Palladium FCC elastic constants plot

J.6.3 Palladium FCC Results File

Listing J.5: Palladium FCC DFT Calculated Properties

```
#####
# RESULTS #
#####

INPUT SETTINGS
#####
1 Structure: fcc
2 Size: 2
3 AO: 3.89
4 AO Units: ang
5 AO (Bohr): 14.7042
6 AO (Angs): 7.7785218
7
8 Ecutfc: 71
9 Ecutrho: 430
10 Degauss: 0.04
11 Kpoints: 9 9 9 1 1 1
12 Kpoints Type: automatic
13 EOS Points: 7
14 EOS Strain: 0.005
15 EC Points: 7
16 EC Strain: 0.005
17
18 RELAXED
19 #####
20
21 a0 (Bohr): 14.8394198232
22 CP: 1.0 0.0 0.0
23 0.0 1.0 0.0
24 0.0 0.0 1.0
25
26 Volume (Bohr^3/unit cell): 3267.76460963
27 Density KG/m^3: 11689.7296625
28 AMU per crystal unit: 3405.4400000000014
29 Atoms per crystal unit: 32
30 Total Energy/Ry: -16384.95810846
31 Energy per Atom/Ry: -512.029940889375
32 Total Force Ry/Bohr: 0.000188
33 Force per atom Ry/Bohr: 5.875e-06
34
35 a0 primitive (Bohr): 7.4197099116
36 a0 primitive (Ang): 3.9250265432364
37
38 EOS
39 #####
40
41
42
43
44
45
46
47
48
49
50 V0 (Bohr^3 / atom): 103.055133214
51 E0 (Ry / atom): -512.029995887
52 B0 (RY/BOHR3): 0.012538146479
53 B0 (GPA): 184.442403779
54 BOP: 4.7014295893
55
56
57 EC
58 #####
```

59
 60 Stiffness (RY/BOHR3): 0.0149806 0.010383 0.0103812 0.0 0.0 0.0
 61 0.010383 0.0149801 0.0103722 0.0 0.0 0.0
 62 0.0103812 0.0103722 0.0149778 0.0 0.0 0.0
 63 0.0 0.0 0.0 0.0055032 0.0 0.0
 64 0.0 0.0 0.0 0.0 0.0055036 0.0
 65 0.0 0.0 0.0 0.0 0.0 0.0055024
 66
 67 Stiffness (GPA): 218.471929 151.422406 151.395627 0.0 0.0 0.0
 68 151.422406 218.464281 151.264808 0.0 0.0 0.0
 69 151.395627 151.264808 218.430622 0.0 0.0 0.0
 70 0.0 0.0 0.0 80.2564574 0.0 0.0
 71 0.0 0.0 0.0 0.0 80.2629605 0.0
 72 0.0 0.0 0.0 0.0 0.0 80.2449828
 73
 74 Compliance (1/GPA): 0.0105867 -0.0043366 -0.0043346 0.0 0.0 0.0
 75 -0.0043366 0.0105705 -0.0043144 0.0 0.0 0.0
 76 -0.0043346 -0.0043144 0.0105702 0.0 0.0 0.0
 77 0.0 0.0 0.0 0.0124601 0.0 0.0
 78 0.0 0.0 0.0 0.0 0.012459 0.0
 79 0.0 0.0 0.0 0.0 0.0 0.0124618
 80
 81 Stability:
 82 C11: 218.471929458 (Stable)
 83 C11C22 - C12C12: 24799.5680651 (Stable)
 84 C11*C22*C33+2*C12*C13*C23-C11*C23*C23-C33*C12*C12: 7353517.38556 (Stable)
 85 C44: 80.256457365 (Stable)
 86 C55: 80.262960524 (Stable)
 87 C66: 80.244982752 (Stable)
 88
 89 Bulk Modulus B (GPA): 173.725818131
 90 BR (GPA): 173.725800894
 91 BV (GPA): 173.725835369
 92
 93 Shear Modulus G (GPA): 56.5595512391
 94 GR: 51.5472895795
 95 GV: 61.5718128987
 96
 97 Young's Modulus E (GPA): 153.067378318
 98
 99 Poisson's Ratio v: 0.35315234089
 100
 101
 102 L Elastic Wave V: 0.145988361768
 103 T Elastic Wave V: 0.0695585857774
 104 M Elastic Wave V: 0.0782389075958
 105
 106 Debye Temperature: 0.000471028127273
 107
 108 Melting Point: 1910.0K
 109
 110
 111
 112
 113
 114
 115
 116
 117 References
 118 =====
 119

120 First Principles Calculations of Elastic Properties of Metals
121 M. J. Mehl, B. M. Klein, D. A. Papaconstantopoulos
122 1994
123
124 Ab Initio Study of the Elastic and Mechanical Properties of B19 TiAl
125 Y. Wen, L. Wang, H. Liu and L. Song
126 Crystals
127 2017
128
129 Density functional theory for calculation of elastic properties of orthorhombic crystals - applications to TiSi₂
130 P. Ravindran, Lars Fast, P. A. Korzhavyi, B. Johansson
131 Journal of Applied Physics
132 1998

Appendix K

Potential Functions

K.1 Introduction

This section of the appendix covers the types of potential function and common choices of function used and details on how to use these in the potential fitting code.

K.2 Functions

K.2.1 Ackland-Mendelev Pair

$$v(r) = \sum_{i=1}^N (a_i(r - r_i)^3 H(r_i - r)) \quad (\text{K.1})$$

Listing K.1: Ackland Mendelev Pair

```
1 #TYPE ackland_mendelev_pair
2 #P 1.0
3 #PF 0.0
```

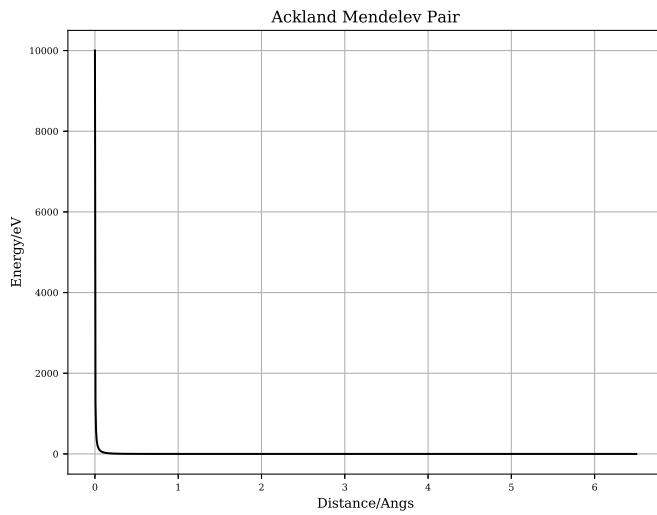


Figure K.1: Ackland Mendelev Pair

K.2.2 Buckingham

$$V(r) = A * \exp(-B * r) - \frac{C}{r * 6} \quad (\text{K.2})$$

Listing K.2: Buckingham

```
1 #TYPE buckingham
2 #P 6.0 0.5 12.0
```

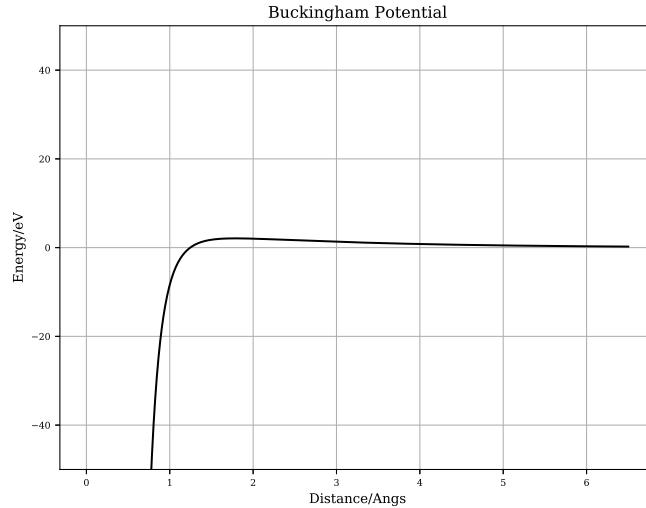


Figure K.2: Buckingham Potential

K.2.3 Cubic Knot Spline

$$V(r) = \sum_i^N a_i (r - r_i)^3 H(r_i - r)$$

where

$$H(x) = \begin{cases} 0 & x < 0 \\ 1 & x \geq 0 \end{cases} \quad (K.3)$$

Listing K.3: Cubic Knot Spline

```
1 #TYPE cubic_knot_spline
2 #P 0.0 1.0 0.02 -0.01 0.002 0.0
3 #PF 0.0 1.0 2.0 3.0 4.0 5.0
```

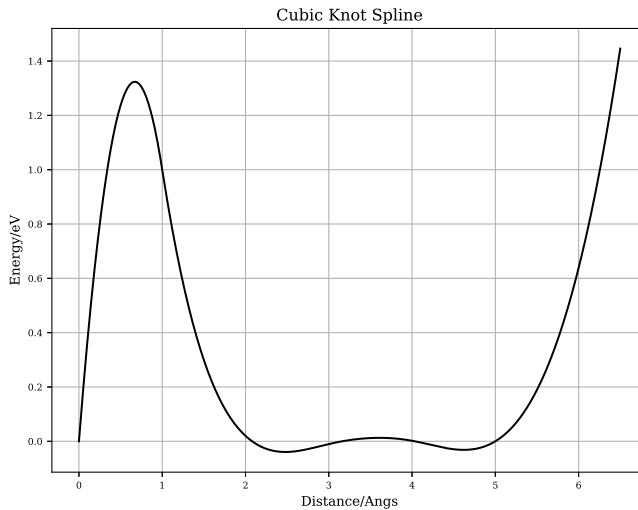


Figure K.3: Cubic Knot Spline

K.2.4 Cubic Knot Spline Fixed End

$$V(r) = \sum_i^N a_i (r - r_i)^3 H(r_i - r)$$

where

$$H(x) = \begin{cases} 0 & x < 0 \\ 1 & x \geq 0 \end{cases} \quad (K.4)$$

Listing K.4: Cubic Knot Spline Fixed End

```
1 #TYPE cubic_knot_spline_fixed_end
2 #P 0.0 1.0 0.02 -0.01 0.002 0.0
3 #PF 0.0 1.0 2.0 3.0 4.0 5.0 6.5 0.0 0.0
```

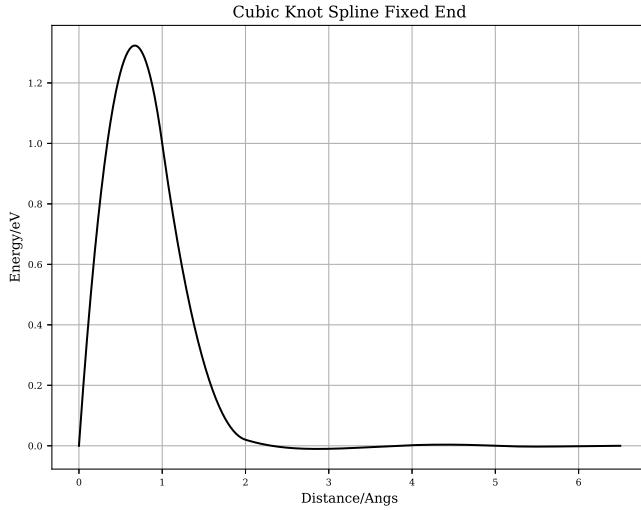


Figure K.4: Cubic Knot Spline Fixed End

K.2.5 Cubic Knot Spline Fixed End Pair

$$V(r) = \sum_i^N a_i (r - r_i)^3 H(r_i - r)$$

where

$$H(x) = \begin{cases} 0 & x < 0 \\ 1 & x \geq 0 \end{cases} \quad (K.5)$$

Listing K.5: Cubic Knot Spline Fixed End Pair

```

1 #TYPE cubic_knot_spline_fixed_end_pair
2 #P 0.5 0.007 -0.3 0.000001 0.002 0.0
3 #PF 2.0 2.5 3.0 3.5 4.0 5.0 26.0 26.0 1.0 6.5 0.0 0.0

```

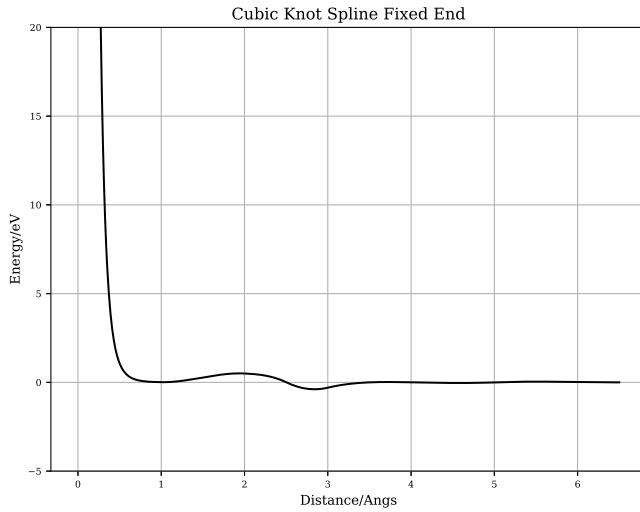


Figure K.5: Cubic Knot Spline Fixed End Pair

K.2.6 Cubic Splines

$$V(r) = \sum_i^N a_i (r - r_i)^3 H(r_i - r) \quad \text{where} \quad (K.6)$$

$$H(x) = \begin{cases} 0 & x < 0 \\ 1 & x \geq 0 \end{cases}$$

This function requires two sets of parameters. P is a list of N coefficients and PF is a list of N cutoffs. The cubic polynomials are summed and individually scaled by the coefficients, and by virtue of its form and the heaviside step function, they cut off at the desired radius.

Listing K.6: Cubic Splines

```

1 #TYPE cubic_spline
2 #P -165.0 -78.5 -78.15 1.868
3 #PF 0.976 1.15 1.216 1.650

```

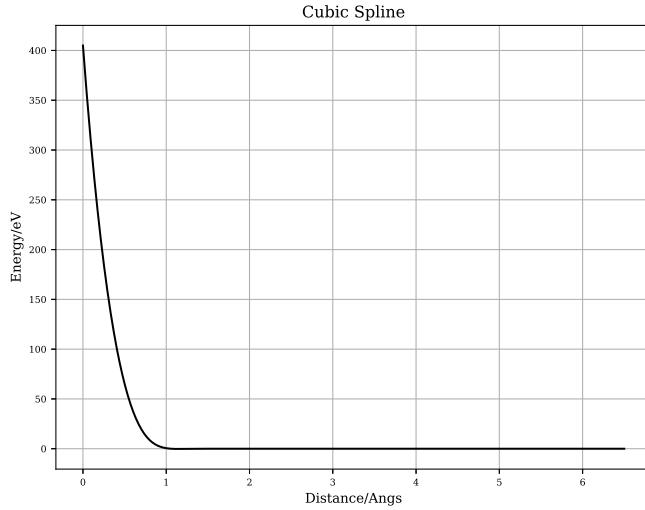


Figure K.6: Ackland Embedding

K.2.7 Cubic Spline Plus ZBL

$$V(r) = \begin{cases} z(r) & r < r_1 \\ a_0 + a_1r + a_2r^2 + a_3r^3 & r_1 \leq r \leq r_2 \text{ where} \\ \sum_i^N a_i(r - r_i)^3 H(r_i - r) & r > r_2 \end{cases} \quad (K.7)$$

$$H(x) = \begin{cases} 0 & x < 0 \\ 1 & x \geq 0 \end{cases}$$

This function requires two sets of parameters. P is a list of N coefficients and PF is a list of N cutoffs. The cubic polynomials are summed and individually scaled by the coefficients, and by virtue of its form and the heaviside step function, they cut off at the desired radius.

Listing K.7: Cubic Splines Plus ZBL

```

1 #TYPE cubic_spline_zbl
2 #P 0.707937643181 1.00089008639 0.978759224081 -0.0460365116314 -0.00762125172653 -0.0209035950064
   0.00938859724754 0.000854708875904 0.0230647235533 0.011459696349 0.023419434017 0.0220932107774
   0.0325043051929 -0.0425828941427 -0.0147289840464 0.0138435254515 0.000880298224106 -0.0134658842396
   0.00102524605929
3 #PF 1.7 1.8 2.0 2.1 2.2 2.3 2.4 2.5 2.6 2.7 2.8 3.0 3.3 3.7 4.2 4.7 5.3 6.0 6.5 26.0 26.0 0.55 1.7 1.0

```

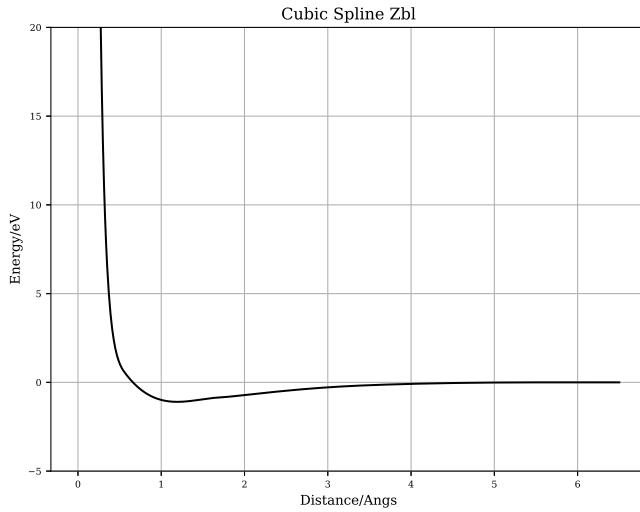


Figure K.7: Ackland Embedding

Listing K.8: Cubic Splines Plus ZBL

```

1 #TYPE cubic_spline_zbl
2 #P 1.00048374974 0.999953842306 0.817335736705 -0.0469506463728 -0.118042905174 0.00429308591859 0.024332744974
   0.0454260983505 0.0205392979832 0.415064378029 0.649740384042 0.0433627017609 0.047890256076 0.026097144285
   0.0106186840232 0.0138597059993 -0.0172453611171 -0.00144350314717 -0.00170969310596
3 #PF 1.7 1.8 2.0 2.1 2.2 2.3 2.4 2.5 2.6 2.7 2.8 3.0 3.3 3.7 4.2 4.7 5.3 6.0 6.5 26.0 26.0 0.6 1.6 1.0

```

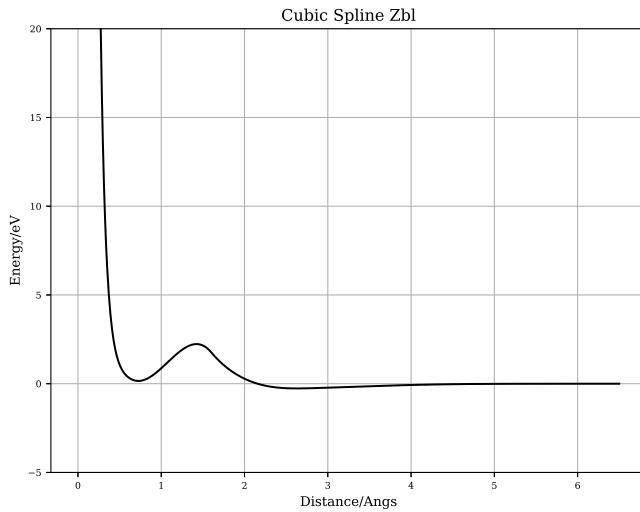


Figure K.8: Ackland Embedding

K.2.8 Embedding A - Finnis Sinclair

$$F(\rho) = -a * \sqrt{\rho} \quad (\text{K.8})$$

This function requires one parameter. Finnis MW and Sinclair J E 1984 Phil. Mag. A 50 45

Listing K.9: Embedding A

```
1 #TYPE embedding_a
2 #P 0.1
```

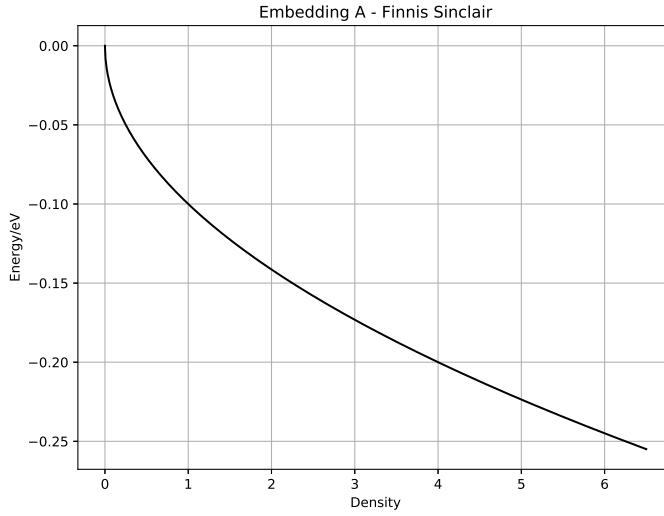


Figure K.9: Embedding A

K.2.9 Embedding B - Mendelev

$$F(\rho) = -\sqrt{\rho} + a\rho^2 \quad (\text{K.9})$$

This function requires one parameter. Development of new interatomic potentials appropriate for crystalline and liquid iron, 2003.

Listing K.10: Embedding B

```
1 #TYPE embedding_b
2 #P 0.1
```

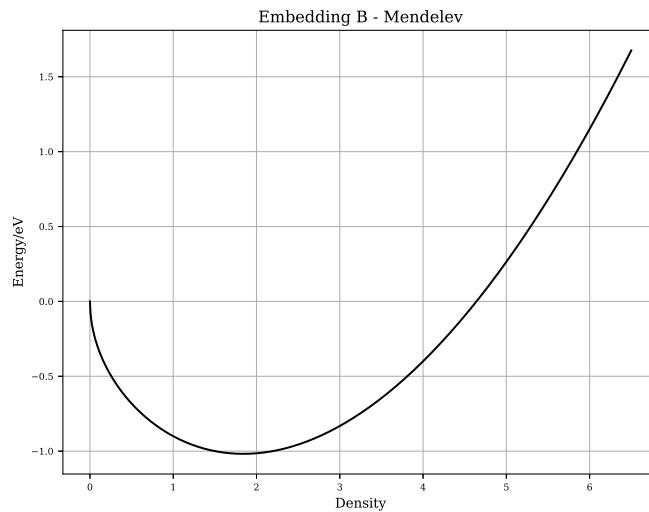


Figure K.10: Embedding B

K.2.10 Embedding C

$$F(\rho) = a\sqrt{\rho} + b\rho^2 \quad (\text{K.10})$$

Listing K.11: Embedding C

```
1 #TYPE embedding_c
2 #P 0.1 0.1
```

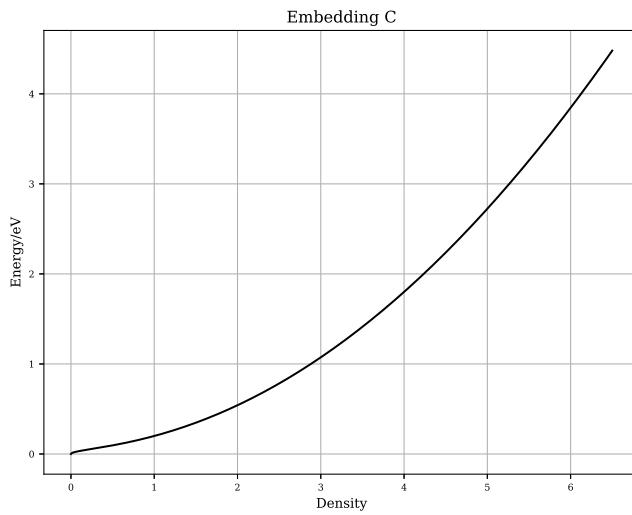


Figure K.11: Embedding C

K.2.11 Embedding D - Ackland-Mendelev

$$F(\rho) = -\sqrt{\rho + a\rho^2 + b\rho^4} \quad (\text{K.11})$$

This function requires one parameter. Development of an interatomic potential for phosphorus impurities in -iron, 2004.

Listing K.12: Embedding D

```
1 #TYPE embedding_d
2 #P 0.05 -0.00001
```

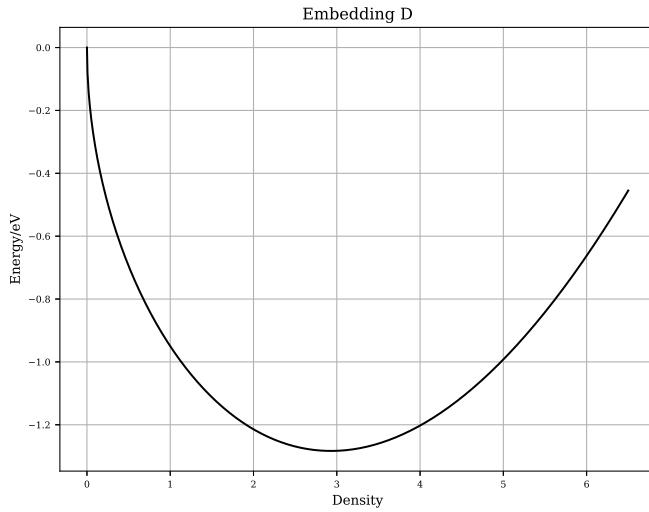


Figure K.12: Embedding D

K.2.12 Embedding E

$$F(\rho) = a * \text{sqrt}\rho + b\rho^2 + c\rho^4 \quad (\text{K.12})$$

Listing K.13: Embedding E

```

1 #TYPE embedding_e
2 #P -0.8 0.05 -0.00001

```

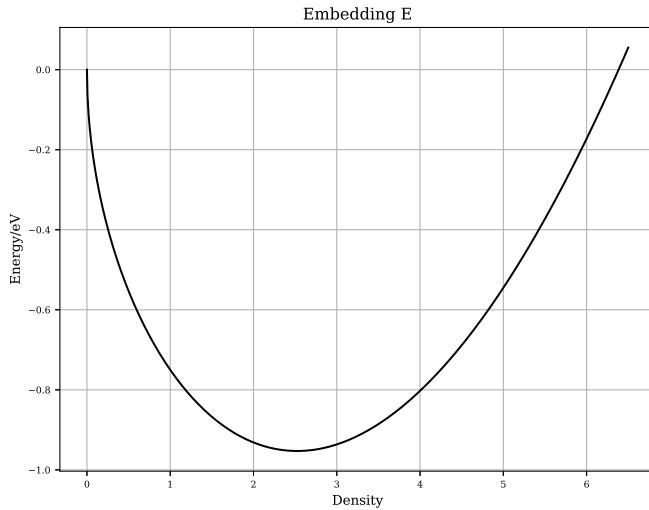


Figure K.13: Ackland Embedding

K.2.13 Embedding F

$$F(\rho) = a * \text{sqrt}\rho + b\rho^2 + \rho^4 \quad (\text{K.13})$$

Listing K.14: Embedding F

```
1 #TYPE embedding_f
2 #P -0.8 0.05
```

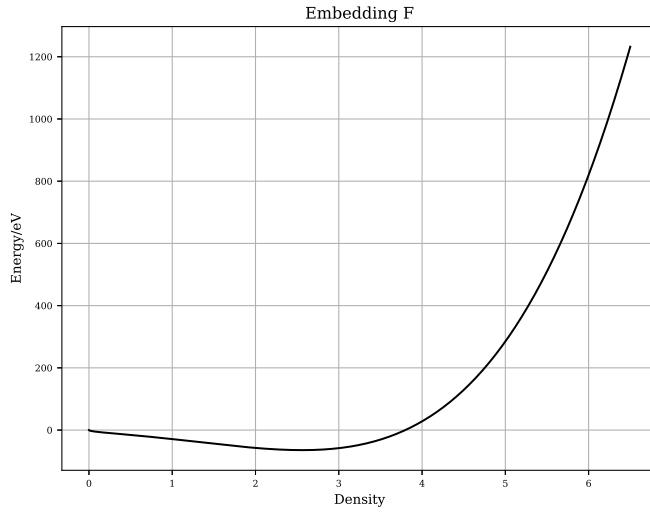


Figure K.14: Embedding F Function Plot

K.2.14 Embedding G

$$F(\rho) = a * \sqrt{\rho} + b\rho^2 + f\rho^4 \quad (\text{K.14})$$

Listing K.15: Embedding F

```
1 #TYPE embedding_g
2 #P -9.15 5.57
3 #PF 0.01
```

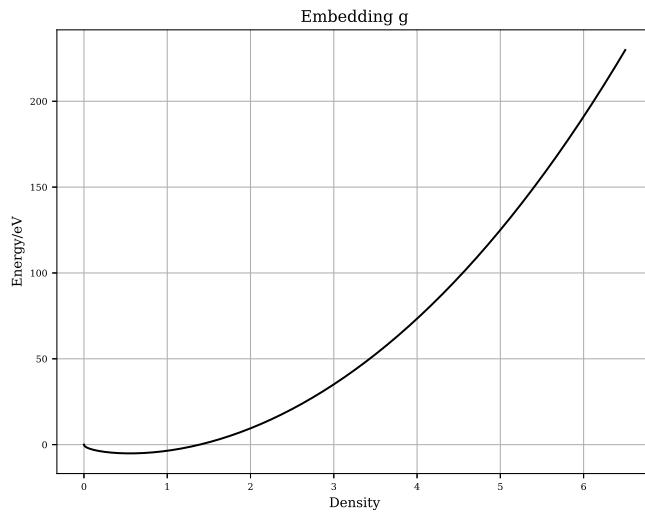


Figure K.15: Embedding G Function Plot

K.2.15 Lennard-Jones

$$V(r) = e \left(\left(\frac{r_m}{r} \right)^1 2 - 2 \left(\frac{r_m}{r} \right)^6 \right) \quad (\text{K.15})$$

Listing K.16: Lennard-Jones

```
1 #TYPE lennard_jones
2 #P 2.3 3.5
```

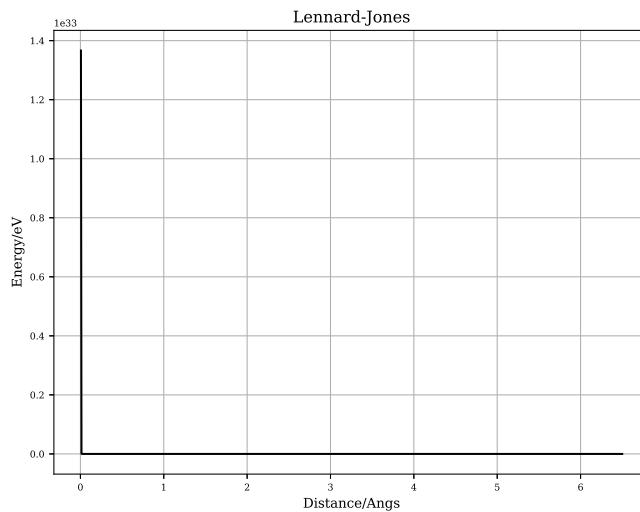


Figure K.16: Lennard-Jones

K.2.16 Morse

$$V(r) = \exp(-2a(r - re)) - 2\exp(-a \times (r - re)) \quad (\text{K.16})$$

Listing K.17: Morse

```
1 #TYPE morse
2 #P 4.669 1.256 2.8
```

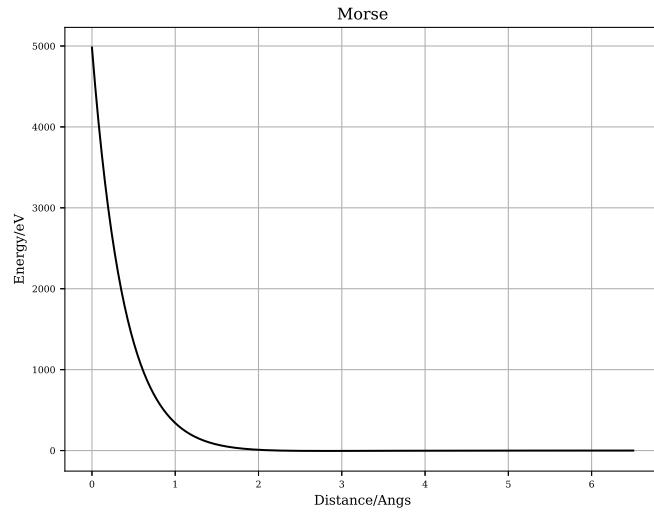


Figure K.17: Morse

K.2.17 Slater 4S (Squared)

$$\rho(r) = (N_s r^3 \exp(-\eta r))^2 \quad (\text{K.17})$$

Listing K.18: Slater 4S

```
1 #TYPE slater_4s
2 #P 5.0 1.323
```

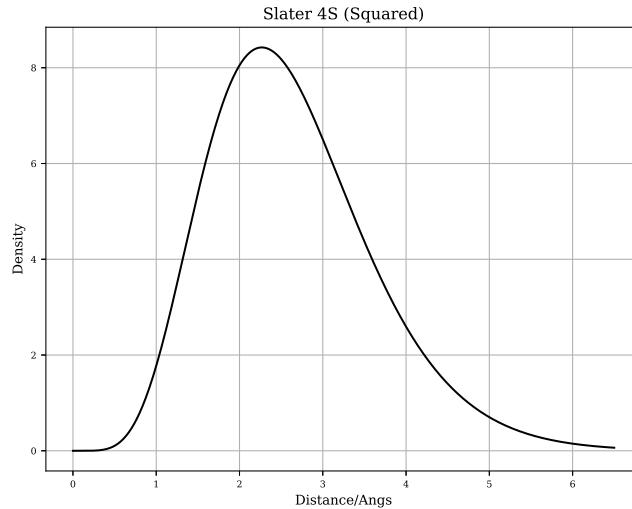


Figure K.18: Quadratic Density

K.2.18 ZBL

$$\phi(x) = 0.181e^{-3.2x} + 0.5099e^{-0.9423x} + 0.2802e^{-0.4029x} + 0.02817e^{-0.2016x}$$

$$\text{where } a_{ij} = \frac{0.8854a_0}{Z_i^{0.23} + Z_j^{0.23}} \quad (\text{K.18})$$

and $a_0 = 0.529\text{angstrom}$

Listing K.19: ZBL

```
1 #TYPE zbl
2 #P 26.0 26.0
```

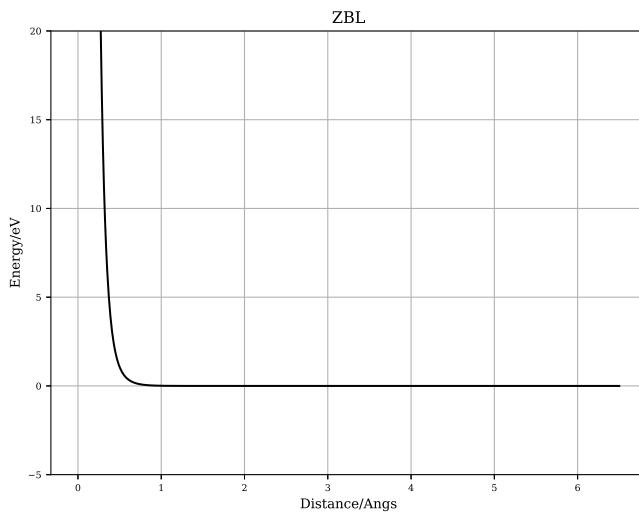


Figure K.19: ZBL

K.2.19 Zero

$$f(x) = 0 \quad (\text{K.19})$$

Listing K.20: Zero

```
1 #TYPE zero
2 #P 0.0
```

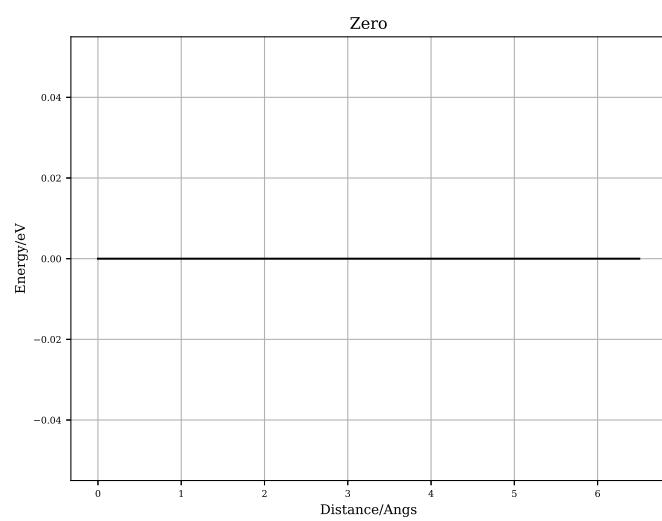


Figure K.20: Zero

Appendix L

FePd Fit: Version 1

L.1 Potential Files

Listing L.1: Potential Index File fepd.pot

```
1 POTNAME fe_pot
2
3 START
4 F_ON true
5 FILE fe_pair.pot
6 LABEL FE FE
7 F_TYPE PAIR
8 END
9
10 START
11 F_ON true
12 FILE fe_dens.pot
13 LABEL FE
14 F_TYPE DENS
15 F_GROUP Group1
16 END
17
18 START
19 F_ON true
20 FILE fe_embe.pot
21 LABEL FE
22 F_TYPE EMBE
23 F_GROUP Group1
24 END
25
26 START
27 F_ON true
28 FILE pd_pair.pot
29 LABEL PD PD
30 F_TYPE PAIR
31 END
32
33 START
34 F_ON true
35 FILE pd_dens.pot
36 LABEL PD
37 F_TYPE DENS
38 F_GROUP Group1
39 END
```

```

40
41 START
42 F_ON true
43 FILE pd_embe.pot
44 LABEL PD
45 F_TYPE EMBE
46 F_GROUP Group1
47 END
48
49 START
50 F_ON true
51 FILE fepd_pair.pot
52 LABEL FE PD
53 F_TYPE PAIR
54 END

```

Listing L.2: fe_pair.pot

```

1 #TYPE cubic_knot_spline_fixed_end
2 #P 50089.4914436 1495.0173568 41.8077313048 2.62868971288 0.884035467244 0.518520725075 0.310525323002
   0.151317904619 0.0457231898041 -0.00199232081514 -0.0285815828726 -0.044141420039 -0.0826842455756
   -0.0800699021737 -0.0749035806431 -0.0483139551314 -0.0271622770015 -0.0244614853927 -0.0276516107094
   -0.0215826770276 -0.0100294465383 -0.00704320230389 -0.00983027553784 -0.00397339407663 -0.00187138774473
   0.000692137630225 0.00191634452235 0.00292127387636 0.00236805599954 0.000626427605252 -0.000652560615754
   -0.0011085461043 -0.0012092545989 -0.000773644125613 -0.000205249951821 -2.49686593374e-08 0.0
3 #PF 0.0 0.1472 1.0304 1.9328 2.2272 2.3744 2.5216 2.6688 2.816 2.9632 3.1104 3.2576 3.4048 3.4944 3.6416 3.7888
   3.936 4.0832 4.2304 4.3776 4.5248 4.672 4.8192 4.9664 5.1136 5.2608 5.408 5.4848 5.6128 5.7408 5.8688 5.9968
   6.0096 6.1376 6.2656 6.3936 6.4 6.5 0.0 0.0
4 #VR 1.0

```

Listing L.3: pd_pair.pot

```

1 #TYPE cubic_knot_spline_fixed_end
2 #P 57433.8196425 1246.54498259 372.86926226 186.530902261 90.9601078094 64.4722319917 44.7955309107 30.9422713797
   19.7661678393 22.7730631807 17.8712893446 20.5301963435 15.7532904958 8.14434737667 3.57750340968
   2.24283276474 1.29878073656 0.681897587232 0.3413406266 0.139633674104 0.0415441088092 -0.00157891985762
   -0.0265766263265 -0.0519228478909 -0.0713202575903 -0.0818422199207 -0.0763981760836 -0.0408735005309
   -0.0337434510853 -0.0340454413917 -0.0221603780251 -0.0140921270967 -0.0112860782378 -0.0075565994614
   -0.00650714812082 -0.00492928641989 -0.00224835545732 0.000551830630351 0.00190602741869 0.00239930313018
   0.00185508157381 0.000683532974593 -0.000618830916973 -0.00106264430655 -0.00107459874027 -0.000816500845506
   -0.000221711741725 -2.71267919382e-08 0.0
3 #PF 0.0 0.1472 0.2944 0.4416 0.5888 0.736 0.8832 1.0304 1.1776 1.1968 1.344 1.4912 1.6384 1.7856 1.9328 2.08
   2.2272 2.3744 2.5216 2.6688 2.816 2.9632 3.1104 3.2576 3.4048 3.4944 3.6416 3.7888 3.936 4.0832 4.2304 4.3776
   4.5248 4.672 4.8192 4.9664 5.1136 5.2608 5.408 5.4848 5.6128 5.7408 5.8688 5.9968 6.0096 6.1376 6.2656
   6.3936 6.4 6.5 0.0 0.0
4 #VR 0.2

```

Listing L.4: fepd_pair.pot

```

1 #TYPE cubic_knot_spline_fixed_end
2 #P 50509.5071771 1510.58202206 41.6181984732 2.73482804796 0.877876472889 0.515874627808 0.308018317521
   0.157372115281 0.0464758155843 -0.00201060627526 -0.0293229989679 -0.0438255499706 -0.0818154118837
   -0.079548485098 -0.0743015444599 -0.0497182011604 -0.027687237496 -0.0241247837683 -0.0288217312248
   -0.0218938512359 -0.00984505409127 -0.00702367568202 -0.00995935352427 -0.00396441140107 -0.00186009860295
   0.000684536237639 0.00191771085336 0.0028941005267 0.00229363524352 0.000624566932013 -0.000659755922205
   -0.00111343976897 -0.00119282373519 -0.000778770104568 -0.000201643446637 -2.50582829498e-08 0.0
3 #PF 0.0 0.1472 1.0304 1.9328 2.2272 2.3744 2.5216 2.6688 2.816 2.9632 3.1104 3.2576 3.4048 3.4944 3.6416 3.7888
   3.936 4.0832 4.2304 4.3776 4.5248 4.672 4.8192 4.9664 5.1136 5.2608 5.408 5.4848 5.6128 5.7408 5.8688 5.9968
   6.0096 6.1376 6.2656 6.3936 6.4 6.5 0.0 0.0
4 #VR 1.0

```

Listing L.5: fe_dens.pot

```

1 #TYPE cubic_knot_spline
2 #P 0.0 1.11185118593 1.07405655632 0.501301076432 0.157569861684 0.0558612376686 0.0331469688897 0.0185595436281
   0.0100054223074 0.00599263304398 0.0031448455987 0.00203980413638 0.00194732163608 0.00155166455135
   0.00107322938655 0.000574828582083 0.000328728725463 0.000104765166684 2.21153619775e-05 7.18575750428e-07
   0.0
3 #PF 0.0 1.184 1.1904 1.4912 1.792 2.0928 2.3936 2.6944 2.9952 3.296 3.5968 3.8976 4.1984 4.4992 4.8 5.1008 5.4016
   5.7024 6.0032 6.304 6.4
4 #VR 0.2

```

Listing L.6: pd_dens.pot

```

1 #TYPE cubic_knot_spline_fixed_end
2 #P 0.0 0.0575871210036 0.0349621319115 0.0175098696686 0.0102461350982 0.00571266625643 0.00297811468455
   0.00210702585433 0.00197274218575 0.00158627909383 0.00107976661552 0.0005622208493 0.000317467916465
   0.000111251939791 2.24374877178e-05 6.933985749e-07 0.0
3 #PF 0.0 2.0928 2.3936 2.6944 2.9952 3.296 3.5968 3.8976 4.1984 4.4992 4.8 5.1008 5.4016 5.7024 6.0032 6.304 6.4
   6.5 0.0 0.0
4 #VR 0.2

```

Listing L.7: fe_embe.pot

```

1 #TYPE embedding_g
2 #P -8.8990305293 -0.656860026472
3 #PF 0.1
4 #VR 2.0

```

Listing L.8: fe_embe.pot

```

1 #TYPE embedding_g
2 #P -9.50412589256 8.00275039263
3 #PF 0.1
4 #VR 0.5

```

L.2 Potential Plots

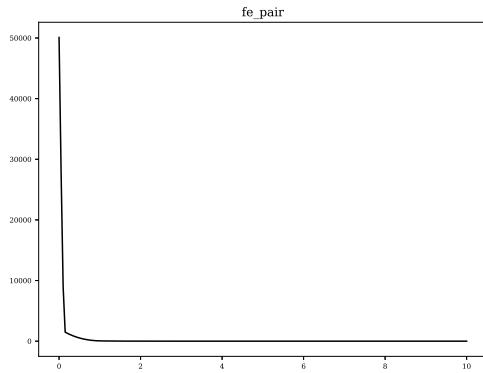


Figure L.1: Fe-Fe Pair Function

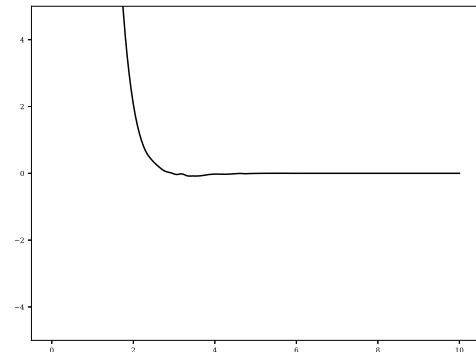


Figure L.2: Fe-Fe Pair Function (Zoomed In)

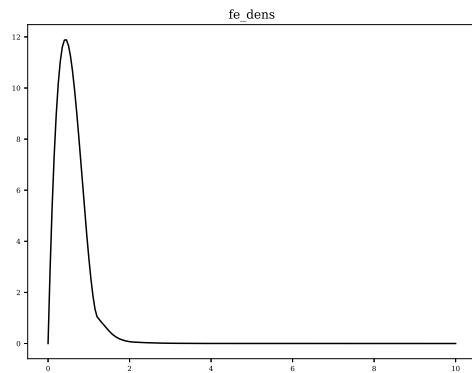


Figure L.3: Fe Density Function

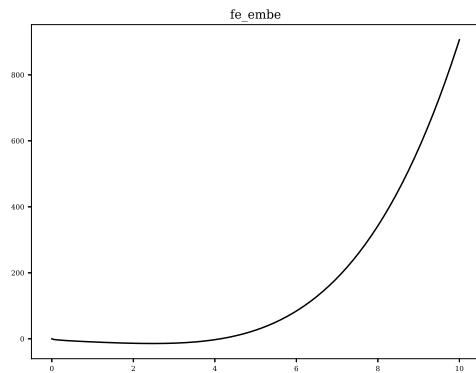


Figure L.4: Fe Embedding Function

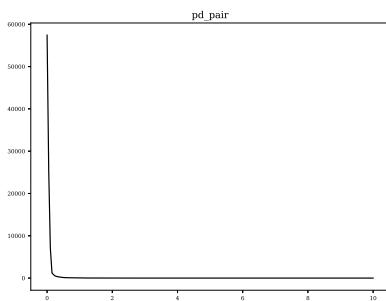


Figure L.5: Pd-Pd Pair Function

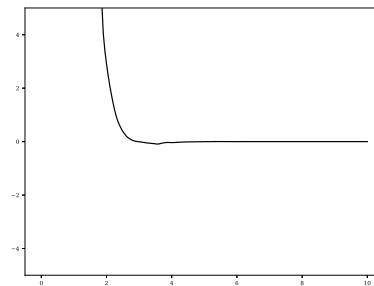


Figure L.6: Pd-Pd Pair Function (Zoomed In)

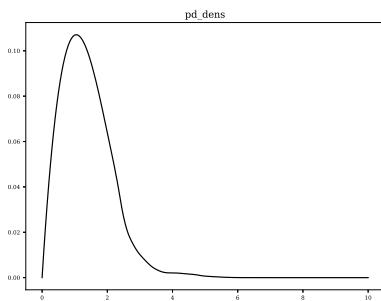


Figure L.7: Pd Density Function

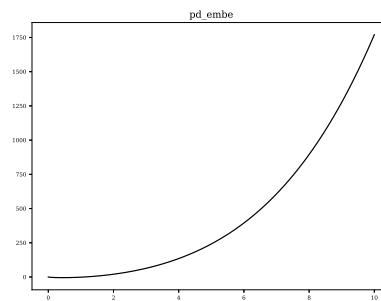


Figure L.8: Pd Embedding Function

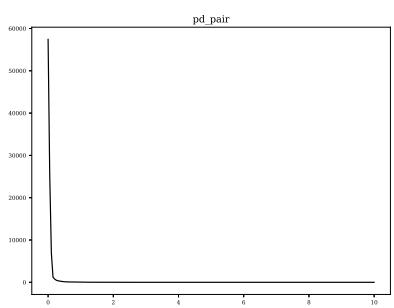


Figure L.9: Fe-Pd Pair Function

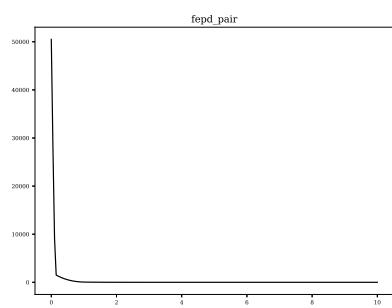


Figure L.10: Fe-Pd Pair Function (Zoomed In)

L.3 Equation of State and Elastic Constant (Distortion) Plots

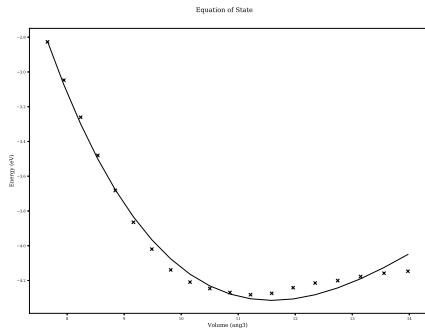


Figure L.11: Iron equation of state

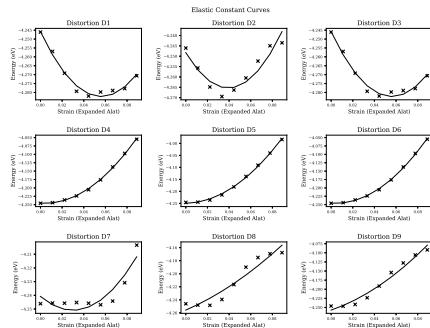


Figure L.12: Iron elastic constants

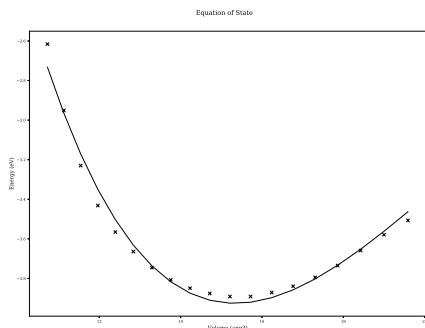


Figure L.13: Palladium equation of state

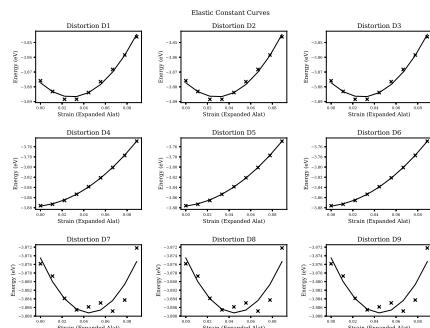


Figure L.14: Palladium elastic constants

L.4 Cohesive Energy Plots

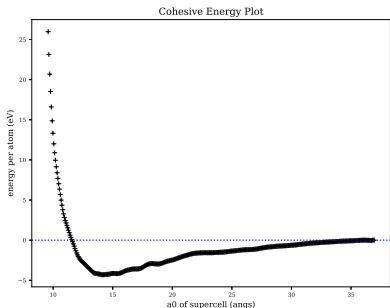


Figure L.15: Iron cohesive energy

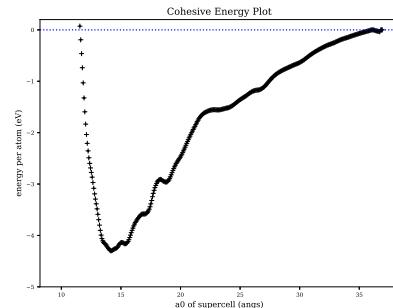


Figure L.16: Iron cohesive energy (zoomed in)

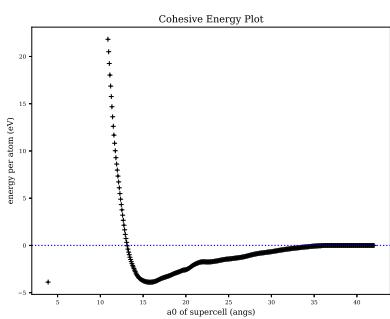


Figure L.17: Palladium cohesive energy

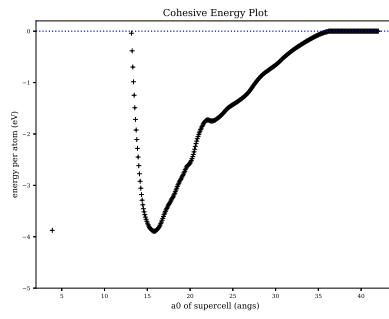


Figure L.18: Palladium cohesive energy (zoomed in)

L.5 Surface Energy Plots

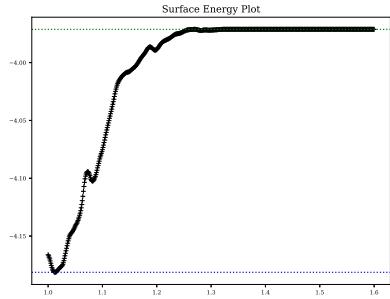


Figure L.19: Iron surface energy

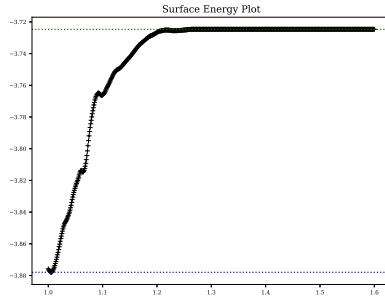


Figure L.20: Palladium surface energy

L.6 Fe-Pd V1 Potential Properties

Parameter	Experimental/DFT	This Potential
a_0	3.42	3.58
e_0	-4.27	-4.27
B_0	222.0	238.8
C_{11}	365.6	356.8
C_{22}	298.7	296.1
C_{33}	364.0	356.8
C_{44}	186.3	180.9
C_{55}	266.8	263.2
C_{66}	186.3	180.9
C_{12}	141.6	139.1
C_{13}	233.8	241.7
C_{23}	130.4	127.3

Table L.1

Parameter	Experimental/DFT	This Potential
a_0	3.89	3.95
e_0	-3.91	-3.91
B_0	180.0	171.3
C_{11}	234.0	248.9
C_{12}	176.0	167.2
C_{44}	71.2	67.6

Table L.2

Appendix M

FePd Fit: Version 2

M.1 Potential Files

Listing M.1: Potential Index File fepd.pot

```
1 POTNAME pdfe_pot
2
3 START
4 F_ON true
5 FILE fe_pair.pot
6 LABEL FE FE
7 F_TYPE PAIR
8 END
9
10 START
11 F_ON true
12 FILE pd_pair.pot
13 LABEL PD PD
14 F_TYPE PAIR
15 END
16
17 START
18 F_ON true
19 FILE fepd_pair.pot
20 LABEL FE PD
21 F_TYPE PAIR
22 END
23
24 START
25 F_ON true
26 FILE fe_dens_slater_4s.pot
27 LABEL FE
28 F_TYPE DENS
29 F_GROUP D1
30 END
31
32 START
33 F_ON true
34 FILE fe_embe_ackland.pot
35 LABEL FE
36 F_TYPE EMBE
37 F_GROUP D1
38 END
39
```

```

40 START
41 F_ON true
42 FILE pd_dens_slater_4s.pot
43 LABEL PD
44 F_TYPE DENS
45 F_GROUP D1
46 END
47
48 START
49 F_ON true
50 FILE pd_embe_ackland.pot
51 LABEL PD
52 F_TYPE EMBE
53 F_GROUP D1
54 END

```

Listing M.2: fe_pair.pot

```

1 #TYPE cubic_knot_spline_5
2 #P 0.534414176557 2.47158957018 3.62572960689 3.82910960636 5.19186569757 6.33755681101 -0.841594441898
   -0.0382741742048 -0.0294982066555 -0.128378635827 0.00520712319717 -0.00173340452865 0.0394106099537
   -0.0422374384814 -0.0212703972446 8.0576645813 0.0114178033157
3 #PF 26.0 26.0 1.0 0.0 6.5 0.0 0.0
4 #VR 0.01

```

Listing M.3: pd_pair.pot

```

1 #TYPE cubic_knot_spline_5
2 #P 0.621299299174 2.49026129701 3.70671760046 4.0464633078 5.12962452826 10.465167784 -0.531944848494
   0.00813465723736 0.017090970358 0.0234628754648 0.00109856040369 0.00122784782403 0.00107526425391
   0.00075123581898 0.000782865546721 38.7466665287 0.00147277281864
3 #PF 26.0 26.0 1.0 0.0 6.5 0.0 0.0
4 #VR 0.01

```

Listing M.4: fepd_pair.pot

```

1 #TYPE cubic_knot_spline_5
2 #P 0.602048570084 2.26359176313 3.43770789475 3.82473216113 6.2662084167 10.0200084349 -0.575534460106
   0.00701145112101 0.0163328650575 0.0209807243721 0.000935085295777 0.00104298283428 0.00113879012964
   0.00108097915755 0.000910022605946 47.310687291 0.00094153913992
3 #PF 26.0 26.0 1.0 0.0 6.5 0.0 0.0
4 #VR 0.01

```

Listing M.5: fe_dens.pot

```

1 #TYPE slater_4s
2 #P 0.112239220675 1.4062318087
3 #PF 0.0
4 #VR 0.01

```

Listing M.6: pd_dens.pot

```

1 #TYPE slater_4s
2 #P 0.841303167535 1.82754932548
3 #PF 0.0
4 #VR 0.01

```

Listing M.7: fe_embe.pot

```

1 #TYPE embedding_g
2 #P 17.3081279913 -42.0918942985
3 #PF 0.1
4 #VR 0.01

```

Listing M.8: fe_embe.pot

```
1 #TYPE embedding_g
2 #P -6.74702954151 41.2855552159
3 #PF 0.1
4 #VR 0.01
```

M.2 Potential Plots

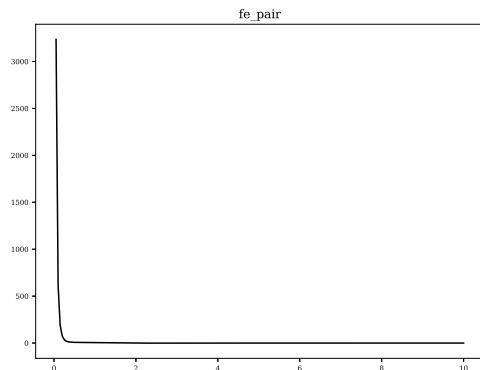


Figure M.1: Fe-Fe Pair Function

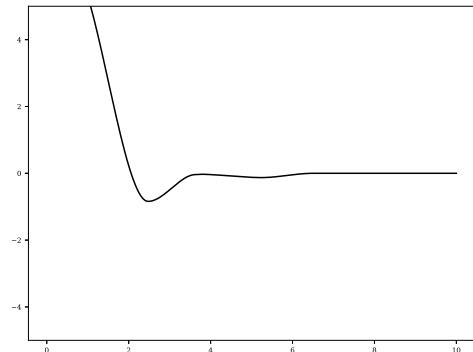


Figure M.2: Fe-Fe Pair Function (Zoomed In)

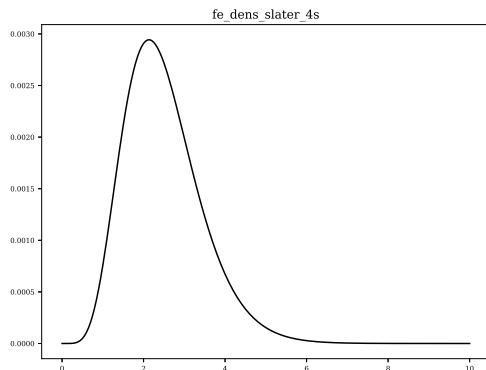


Figure M.3: Fe Density Function

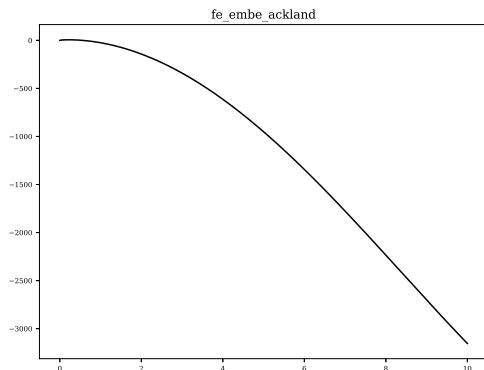


Figure M.4: Fe Embedding Function

M.3 Equation of State and Elastic Constant (Distortion) Plots

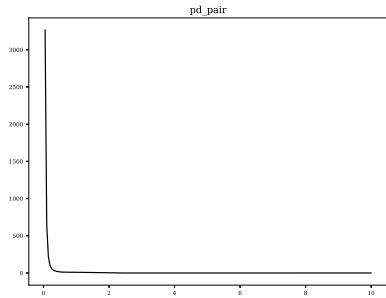


Figure M.5: Pd-Pd Pair Function

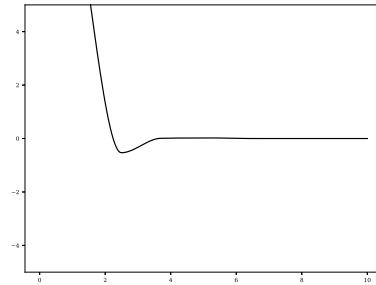


Figure M.6: Pd-Pd Pair Function (Zoomed In)

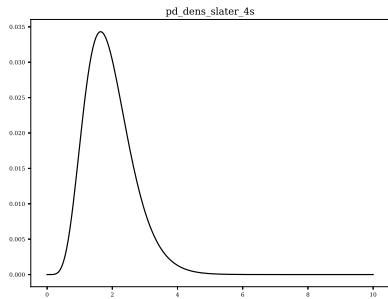


Figure M.7: Pd Density Function

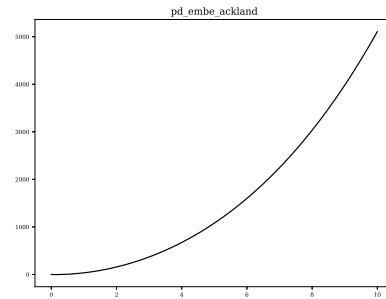


Figure M.8: Pd Embedding Function

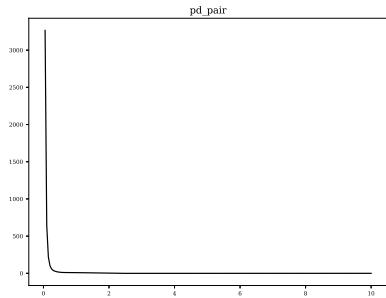


Figure M.9: Fe-Pd Pair Function

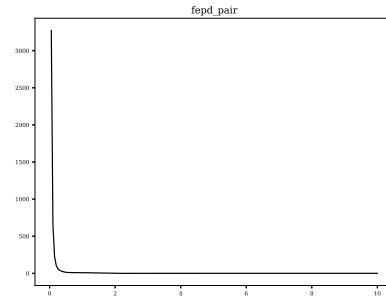


Figure M.10: Fe-Pd Pair Function (Zoomed In)

M.4 Cohesive Energy Plots

M.5 Surface Energy Plots

M.6 Fe-Pd V2 Potential Properties

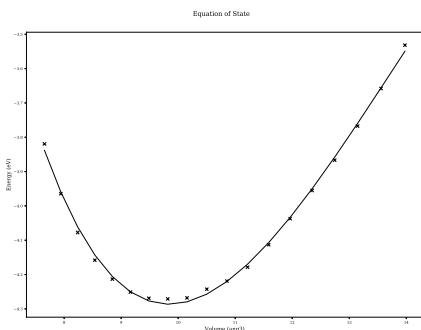


Figure M.11: Iron equation of state

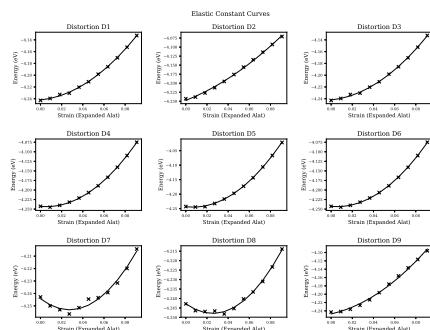


Figure M.12: Iron elastic constants

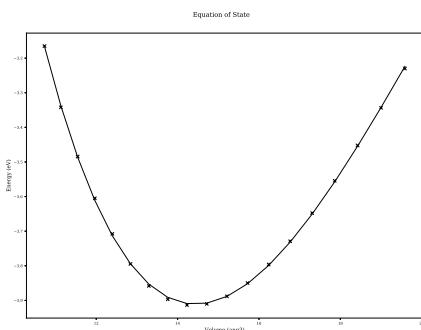


Figure M.13: Palladium equation of state

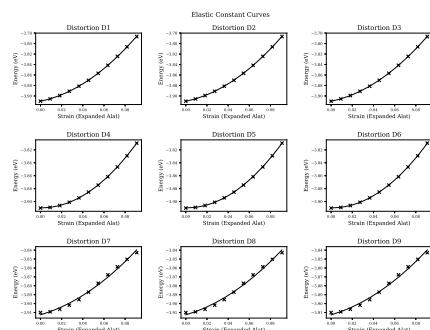


Figure M.14: Palladium elastic constants

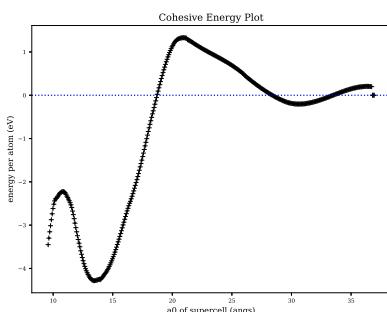


Figure M.15: Iron cohesive energy

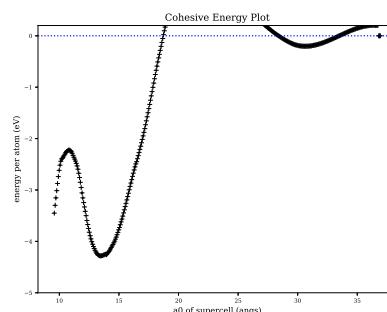


Figure M.16: Iron cohesive energy (zoomed in)

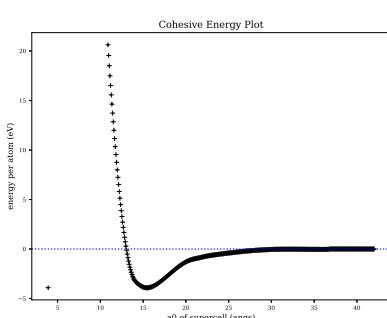


Figure M.17: Palladium cohesive energy

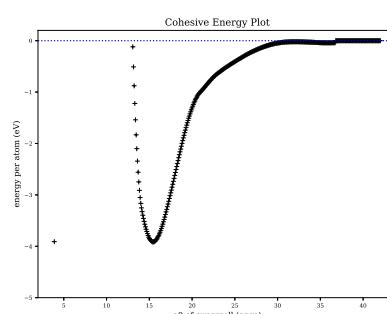


Figure M.18: Palladium cohesive energy (zoomed in)

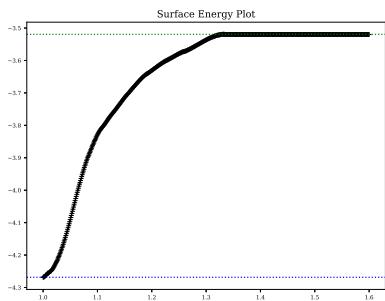


Figure M.19: Iron surface energy

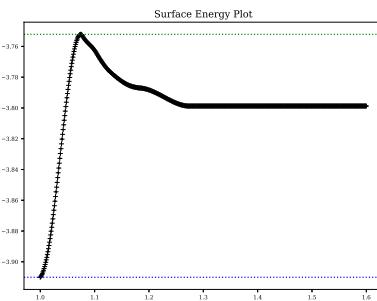


Figure M.20: Palladium surface energy

Parameter	Experimental/DFT	This Potential
a_0	3.42	3.40
e_0	-4.27	-4.29
B_0	222.0	229.1
C_{11}	365.6	365.9
C_{22}	298.7	298.6
C_{33}	364.0	365.9
C_{44}	186.3	187.6
C_{55}	266.8	258.4
C_{66}	186.3	187.6
C_{12}	141.6	141.1
C_{13}	233.8	238.1
C_{23}	130.4	124.8

Table M.1

Parameter	Experimental/DFT	This Potential
a_0	3.89	3.87
e_0	-3.91	-3.91
B_0	180.0	171.3
C_{11}	234.0	232.9
C_{12}	176.0	176.7
C_{44}	71.2	68.8

Table M.2