



NDF v9.6i

© (a) Nuno Pessoa Barradas, 30 de Setembro de 2016

CONTENTS

1	NDF	7
1.1	INTRODUCTION	7
1.2	INSTALLATION	7
1.3	SHORT VERSION HISTORY	7
2	GETTING STARTED	12
2.1	SETTING UP THE PROBLEM	12
2.1.1	<i>The batch file: SPC</i>	12
2.1.2	<i>The sample structure file: STR</i>	12
2.1.3	<i>The geometry files: GEO</i>	12
2.1.4	<i>The depth profile file: PRF</i>	13
2.2	RUNNING NDF	13
2.3	LOOKING AT THE SIMULATION OR FIT	13
2.4	UNITS IN NDF	13
3	THE BATCH SPC FILE: SPECIFYING WHICH FILES TO ANALYSE	14
3.1	THE BATCH FILE .SPC	14
3.2	MULTIPLE DETECTED IONS 1 - E.G. TOF-ERDA, FUNNY FILTERS	15
3.3	MULTIPLE DETECTED IONS 2 - SAME ENERGY CALIBRATION	15
3.4	SIMULTANEOUS DETECTORS	15
3.5	SAME ENERGY CALIBRATION	16
3.6	VERBOSITY	16
3.7	ADVANCED USERS	16
4	THE STR FILE: SPECIFYING THE ELEMENTS AND/OR MOLECULES	17
4.1	THE STRUCTURE FILE *.STR	17
4.1.1	<i>Minimum and maximum layer thickness used in the fit</i>	17
4.1.2	<i>Number of elements / molecules in the sample</i>	17
4.2	TELLING NDF WHAT THE SAMPLE IS MADE OF	17
4.2.1	<i>Defining an element</i>	17
4.2.2	<i>Defining a specific isotope</i>	17
4.2.3	<i>Defining a molecule or a compound of known composition</i>	17
4.2.4	<i>Defining a molecule of unknown composition</i>	18
4.2.5	<i>Defining compound molecules / sub-molecules</i>	18
4.2.6	<i>Pre-defined molecules and substances</i>	18
4.2.7	<i>Specifying the density of an element or molecule</i>	19
4.3	SPECIFYING DEPTH AND CONCENTRATION RANGES FOR A GIVEN ELEMENT	19
4.3.1	<i>Defining the depth range</i>	19
4.3.2	<i>Defining the concentration range</i>	19
4.4	SPECIFYING A CONCENTRATION FORMULA FOR ONE GIVEN ELEMENT OR MOLECULE	20
4.5	SPECIFYING THE PRESENCE OF ROUGHNESS, VOIDS, INCLUSIONS, QUANTUM DOTS	21
4.6	EXTRA "FAKE ELEMENTS"	21
5	THE GEO FILE: SPECIFYING THE EXPERIMENTAL CONDITIONS	22
5.1	THE GEOMETRY FILES *.GEO	22
5.2	DATA TYPE: LINE 1	22
5.2.1	<i>Data compression</i>	23
5.2.2	<i>Real time and live time</i>	23
5.3	ROI: LINE 2	23
5.4	EXPERIMENTAL TECHNIQUE: LINE 3	24
5.4.1	<i>RBS</i>	24
5.4.2	<i>ERDA</i>	24
5.4.3	<i>Non-resonant NRA</i>	24
5.4.4	<i>NRP (resonant NRA, PIGE, whatever)</i>	24
5.4.5	<i>PIXE</i>	24
5.4.6	<i>SIMS (actually, any real depth profiling technique)</i>	26
5.4.7	<i>NDP: Neutron Depth Profiling</i>	27
5.5	BEAM ENERGY: LINE 4	27
5.6	RESOLUTION: LINE 5	28
5.6.1	<i>RBS, ERDA, NRA</i>	28
5.6.2	<i>PIXE</i>	28
5.6.3	<i>NRP (PIGE, RNRA)</i>	28

5.6.4	<i>SIMS</i>	28
5.7	GEOMETRY TYPE: LINE 6	28
5.8	GEOMETRY ANGLES: LINE 7	28
5.8.1	<i>Definition of angles</i>	28
5.8.2	<i>General geometry</i>	29
5.8.3	<i>Transmission geometry</i>	29
5.9	SOLID ANGLE: LINE 8	29
5.10	ENERGY CALIBRATION: LINE 9	29
5.10.1	<i>RBS, ERDA, NRA</i>	29
5.10.2	<i>NRP</i>	29
5.10.3	<i>PIXE</i>	29
5.10.4	<i>SIMS</i>	30
5.11	DETECTOR EFFICIENCY	30
5.12	FOIL OR DELTA-E DETECTOR	30
5.12.1	<i>RBS, ERDA, NRA, NDP</i>	30
5.12.2	<i>Foil thickness inhomogeneity in RBS, ERDA, NRA, NDP</i>	30
5.12.3	<i>PIXE</i>	31
5.13	DEAD LAYER AND NON-IONISING ENERGY LOSS IN THE DETECTOR	31
5.14	THICKNESS OF THE DEPLETED (ACTIVE) LAYER OF THE DETECTOR	32
5.15	FOIL BETWEEN THE BEAM AND THE SAMPLE	33
5.16	READGEO OVERRIDE - CHANGING PARAMETERS IN MANY GEO FILES	33
6	THE GEO FILE PART 2: ADVANCED CALCULATION OPTIONS	35
6.1	STOPPING POWER	35
6.2	BEAM CHARGE STATE	35
6.3	TYPE OF STRAGGLING	35
6.4	TSCHALÄR EFFECT	35
6.5	SHAPE OF THE STRAGGLING	35
6.6	NON-RUTHERFORD CROSS SECTIONS	35
6.7	ACCURATE CALCULATION OF RESONANCES	36
6.8	ELECTRON SCREENING	36
6.9	DOUBLE SCATTERING	36
6.10	PULSE PILE-UP	36
6.11	LOW ENERGY YIELD CORRECTION	36
6.12	AD-HOC PLURAL SCATTERING/CHANNELLING CORRECTIONS	36
6.13	GENERATE APPARENT CROSS SECTIONS THAT LEAD TO A PERFECT FIT TO YOUR DATA	36
6.14	EFFECTS OF MULTIPLE SCATTERING	36
6.15	EXPONENTIAL BACKGROUND FOR NDP	36
6.16	EXTREMELY THIN LAYERS	36
7	RUNNING NDF	37
7.1	SIMULATE A THEORETICAL SPECTRUM FROM A GIVEN DEPTH PROFILE	37
7.2	RUN A LOCAL OPTIMISATION STARTING FROM A GIVEN DEPTH PROFILE	37
7.3	MAIN FIT OPTIONS	37
7.3.1	<i>Speed</i>	38
7.3.2	<i>Compression</i>	38
7.3.3	<i>Convolute FWHM</i>	38
7.3.4	<i>Isotopes</i>	38
7.3.5	<i>Smooth</i>	38
7.3.6	<i>Output point by point profiles</i>	38
7.3.7	<i>Recalculate</i>	38
7.3.8	<i>Normalise χ^2</i>	39
7.3.9	<i>Form of χ^2 function - standard or robustified</i>	39
7.4	EXAMPLES OF POSSIBLE COMMANDS	39
7.5	CONTROLLING WHICH PARAMETERS CHANGE DURING THE FIT	40
7.5.1	<i>Beam energy, angle of scattering, incidence and exit, FWHM</i>	40
7.5.2	<i>Energy calibration and charge</i>	40
7.5.3	<i>Multiple scattering ad-hoc polynomial</i>	41
7.5.4	<i>Number of layers, layer composition, depth profile</i>	41
7.5.5	<i>Roughness parameters</i>	41
7.5.6	<i>Volume fraction of inclusions</i>	41
7.5.7	<i>Fine control of the fitting process in a local search</i>	41
7.6	REPEATING ONLY SOME OF THE FITS IN THE BATCH FILE	42
7.7	REPEAT SIMULATION	42

7.8	AUTOMATED FITTING: SIMANN, BI/MCMC	42
7.8.1	<i>Run a simulated annealing search</i>	43
7.8.2	<i>Local SimAnn</i>	43
7.8.3	<i>Running Bayesian inference</i>	43
7.9	FIT TWEAKING TOOLS	43
7.9.1	<i>normalise χ^2</i>	43
7.9.2	<i>form of χ^2 function - standard or not</i>	43
7.9.3	<i>Errors for χ^2</i>	43
7.9.4	<i>Include Poisson noise in the simulation</i>	43
7.9.5	<i>Random number sequence</i>	43
7.10	RE-USE FIT RESULTS	44
8	OUTPUTS	45
8.1	VERBOSITY	45
8.2	VERBOSITY 0: DEFAULT OUTPUT	45
8.2.1	<i>Result file: bs.RES</i>	45
8.2.2	<i>Calculated spectra: bFsf.DAT</i>	45
8.2.3	<i>Calculated elemental spectra: bXsf.DAT</i>	45
8.2.4	<i>Best layer structure: bPs.DAT</i>	45
8.2.5	<i>Best layer structure: bPEs.DAT</i>	45
8.2.6	<i>Best layer structure: bs.PRF</i>	46
8.2.7	<i>Log file: bs.LOG</i>	46
8.2.8	<i>XML output</i>	46
8.2.9	<i>Calculated double scattering: DSsf.DAT</i>	46
8.2.10	<i>Calculated Pileup: bUsf.DAT</i>	46
8.2.11	<i>Stopping powers and cross sections</i>	46
8.3	VERBOSITY 1	46
8.3.1	<i>Point by point profiles: bsfe.DAT, and signal areas</i>	46
8.3.2	<i>Peaks and edges: bPEAKs.PE</i>	46
8.4	VERBOSITY 2	47
8.4.1	<i>Calculated layer elemental spectra: bEsfel.DAT</i>	47
8.5	VERBOSITY 3	47
8.5.1	<i>Extended intermediate calculations: bCsfei.DAT</i>	47
9	FINE-CONTROLLING THE FITTING PROCESS: NDF.TCN	48
9.1	WHAT IT IS FOR	48
9.2	DETAILED DESCRIPTION	48
10	STOPPING POWER DATA	50
10.1	ZBL2000 STOPPING	50
10.2	ZBL1990 STOPPING	50
10.3	MSTAR STOPPING	50
10.4	SRIM2003 (2006, 2010, ETC) STOPPING	50
10.5	SCALING THE STOPPING POWERS	50
10.6	USER-DEFINED ELEMENTAL AND MOLECULAR STOPPING POWERS	51
10.6.1	<i>Elemental stopping powers</i>	51
10.6.2	<i>Molecular stopping powers</i>	51
10.6.3	<i>Using an input file with (energy,stopping) values</i>	52
10.6.4	<i>Using ZBL90 a1-a8 parameters</i>	52
10.7	HOW TO FIT STOPPING POWERS	52
10.8	HOW TO GET ZBL90 A1-A8 COEFFICIENTS FROM AN INPUT TABLE	52
10.9	CHARGE STATE	52
11	CORRECTIONS TO SINGLE RUTHERFORD SCATTERING	54
11.1	NON-RUTHERFORD CROSS SECTIONS	54
11.1.1	<i>Angular dependent non-Rutherford cross sections from SigmaCalc</i>	54
11.1.2	<i>Inputting non-Rutherford cross sections from a file</i>	55
11.1.3	<i>Internally calculated non-Rutherford cross sections</i>	55
11.1.4	<i>Inputting all non-Rutherford cross sections for a given angle</i>	56
11.1.5	<i>Format and units of user-input non-Rutherford cross sections - R33</i>	56
11.1.6	<i>Format and units of user-input non-Rutherford cross sections - two column ascii</i>	56
11.1.7	<i>Scaling user-input non-Rutherford cross sections - two column ascii</i>	57
11.1.8	<i>Internal handling of non-Rutherford cross sections</i>	57
11.1.9	<i>Automatic division of internal layers with non-Rutherford cross sections</i>	57
11.1.10	<i>Cross section at 0 energy</i>	57

11.2	ACCURATE SIMULATION OF RESONANCES	57
11.3	SCREENING	58
11.4	DOUBLE SCATTERING	58
11.4.1	<i>Calculation of double scattering - automatic control</i>	59
11.4.2	<i>User-input double scattering contribution</i>	60
11.4.3	<i>Recommended usage</i>	60
11.4.4	<i>Calculation of double scattering - manual control</i>	61
11.4.5	<i>Further options</i>	61
11.5	PILE-UP CORRECTION	62
11.5.1	<i>Algorithm of Molodstov and Gurbich</i>	63
11.5.2	<i>Algorithm of Wielopolski and Gardner</i>	63
11.5.3	<i>Two-event pile-up: pure autocorrelation algorithm</i>	63
11.5.4	<i>Two-event pile-up: algorithm of Amsel et al.</i>	63
11.5.5	<i>Automatic pile-up correction on your data</i>	63
11.5.6	<i>Low level discrimination and pile-up</i>	64
11.5.7	<i>Pileup output file and recalculation from file</i>	64
11.5.8	<i>Number of channels in calculation</i>	64
11.5.9	<i>Livetime vs realtime</i>	65
11.6	LOW ENERGY YIELD CORRECTION	65
11.7	AD-HOC SUBSTRATE PLURAL SCATTERING / CHANNELLING CORRECTION.	65
11.7.1	<i>Basics</i>	65
11.7.2	<i>The problem</i>	65
11.7.3	<i>A (relatively general) ad-hoc solution</i>	65
11.7.4	<i>What you have to do</i>	66
11.7.5	<i>Dirtier trick</i>	66
11.7.6	<i>Wim trick</i>	66
11.8	AD-HOC THIN FILM PLURAL SCATTERING CORRECTION	67
11.9	GENERATE APPARENT CROSS SECTIONS THAT LEAD TO A PERFECT FIT TO YOUR DATA	68
11.10	EFFECTS OF MULTIPLE SCATTERING ON ENERGY, ANGULAR AND PATHLENGTH SPREAD	68
11.10.1	<i>Effect on energy spread</i>	68
11.10.2	<i>Effect on scattering angle spread and pathlength spread</i>	68
11.11	NON-RUTHERFORD CROSS SECTIONS: ⁴ HE ON H	69
11.12	NON-RUTHERFORD CROSS SECTIONS: ⁴ HE ON D	69
12	STRAGGLING CALCULATIONS	70
12.1	WDEPTH	70
12.2	MDEPTH	71
12.3	DEPTH	72
12.4	CHU AND BOHR STRAGGLING	72
12.5	SHAPE OF STRAGGLING FUNCTION	73
12.6	TSCHALÄR EFFECT	73
13	ROUGHNESS, POROSITY, INCLUSIONS, DENSITY VARIATION	74
13.1	ROUGHNESS: FAST VS. SLOW CALCULATIONS	74
13.2	GAMMA ROUGHNESS - LAYER THICKNESS INHOMOGENEITY	74
13.3	USER INPUT LAYER THICKNESS DISTRIBUTION	75
13.4	APPROXIMATE FAST ROUGHNESS MODELS	76
13.4.1	<i>Include roughness in a simulation or fit</i>	78
13.5	VOIDS AND INCLUSIONS - APPROXIMATE FAST MODEL	78
13.5.1	<i>Spherical</i>	79
13.5.2	<i>Cylindrical</i>	79
13.5.3	<i>The PRF file</i>	79
13.5.4	<i>Examples</i>	80
13.5.5	<i>Energy spreading and self-organisation</i>	81
13.6	LOCAL VARIATIONS OF DENSITY	81
13.7	EXTRA PATHLENGTH DUE TO SECONDARY CROSSINGS	81
13.7.1	<i>Correlations between models</i>	81
14	CALCULATION OF ERROR BARS	83
14.1	CALCULATE ERROR BARS ON YOUR DEPTH PROFILES USING BAYESIAN INFERENCE	83
14.2	OUTPUT FILES	83
14.3	ERROR BARS ON ROUGHNESS PARAMETERS	83
14.4	MODEL LIMITATIONS	83
14.5	USE THE NDF.TCN FILE	84

14.6	OUTPUT FILES	84
14.7	OTHER PARAMETERS	84
15	DETERMINATION OF STOPPING POWER VALUES	85
15.1	HOW IT WORKS	85
15.2	SIMPLE FITTING	85
15.3	BAYESIAN INFERENCE	85
15.4	RESTRICTIONS AND USAGE	85
15.5	OUTPUT FILES	86
16	DETERMINATION OF NON-RUTHERFORD CROSS SECTION VALUES	87
16.1	HOW IT WORKS	87
16.2	RESTRICTIONS AND USAGE	87
16.3	OUTPUT FILES	88
17	TRADING-OFF ACCURACY AND SPEED	89
18	THE IAEA INTERCOMPARISON OF IBA SOFTWARE: NDF	90

1 NDF

1.1 Introduction

NDF a DOS code dedicated to the analysis of RBS, ERDA, PIXE, non-resonant NRA and NDP data for any ion, any target, any geometry, and number of spectra. The free version of NDF is a simulator only. It does not implement fitting or Bayesian inference. It also does not implement PIXE. It is called ALP internally just to distinguish it from the regular NDF.

The general philosophy of NDF is to put accuracy before calculation speed. In some cases, this leads to calculations that can be orders of magnitude slower. Faster calculations can be made by turning the appropriate options on or off.

NDF is a DOS program which reads input files and creates output files. No graphical interface or output is supplied. The users must use their own graphics package to visually inspect the fits and depth profiles obtained.

NDF is distributed “as is”. Bug reports and suggestions are very welcome. Email me at nunoni@itn.pt

The NDF web page, including downloads of the most recent executable and manual, is http://www.itn.pt/facilities/lfi/ndf/uk_lfi_ndf.htm

The executable distributed in the web page is called ALP_xxx.exe, where xxx can be for instance 90a for version 9.0a. You have to rename this to ndf.exe.

After you read the manual, we advise you to run the examples coming from the International Atomic Agency intercomparison of IBA software, as detailed in chapter 18.

1.2 Installation

The code can be run under DOS or in a DOS shell.

Create a directory (say, C:\NDF) on your hard disc. Copy the distribution files to this directory and unzip them with the Unzip command. Further, you have to configure your computer (assuming C:\NDF to be the directory where you have the executable). You MUST include two lines in AUTOEXEC.BAT:

```
SET PATH=%PATH%;C:\NDF
SET PATHNDF=C:\NDF
```

1.3 Short version history

6Apr98	v2.2	Corrected a few bugs (like not recognising properly e.g. 109Ag as a target species or projectile).
7Apr98	v2.3	Included He on D ERDA cross sections, changed presentation of .RES files to include more info.
8Apr98	v3.0	Included non-resonant NRA.
21Jul98	v3.1	Included the possibility of using DEPTH output.
25Jul98	v4.0	Included Bayesian Inference with the MCMC algorithm.
27Jul9	v4.1	Changed the MCMC algorithm to make it adaptive.
20Aug98	v5.x	Hopelessly bugged first Genetic Algorithm version.
20Aug98	v5.y	Included thin film plural scattering correction – developed on a different machine as v5.x.
18Dec98	v5.3x	Good GenAlg. Not as efficient as SimAnn, dumped it.
5Jan99	v6.0	Introduced molecules, eliminated genetic algorithm
8Jan99	v6.1	Delta layers, substrate layers
22Jan99	v6.2	Introduced element search
8Feb99	v6.3	Debugged NRA
25Feb99	v6.4x	Took SQUEAKIE out, wrote own routine that works for ERDA and NRA as well. New foil calculation. Old ROI still in. Played with iFFT deconvolution of resolution (does not work).
23Mar99	v6.4	Debugged new foil calculation
21Jun99	v6.44	Eliminated (hopefully) inconsistencies between delta-layers and other user-defined constraints.
6Jul99	v6.5x	Introduced fitting of the polynomial corrections. Debugged here and there.
4Aug99	v6.5	Fitting of calibration, charge, etc during SimAnn as well.
30Nov99	v7.0c	New MCMC thickness&stoichiometry calculation.
14Dec99	v7.1c	64 spectra.
03Mar00	v7.1h	Revamped forward function algorithm, introduction of user-input densities.
20Mar00	v7.2b	Introduction of straggling and of possibility of very fine non-Ruth cross section files
23Mar00	v7.2c	32 spectra, other internal changes
27Mar00	v7.3	NDP introduced, bugged version

17Apr00 v7.3c Size of internal layer, cross section and stopping power arrays user-definable in NDF.TCN

Jul00 7.4 BI/MCMC of stopping powers (quite nice)

20Sep00 7.5a BI/MCMC of non-Rutherford cross sections (pretty bad, unusable)

29Nov00 7.5d Local SimAnn from NDF.PRF

29Jan01 7.6a Roughness models 1-3 seem to be debugged and working. Other smaller changes. 16 spectra only

26Jul01 7.6e BI/MCMC on roughness

8Aug01 7.6f linked constant ratio charge for simultaneous multiple detectors

20Sep01 7.6g Roughness model (4)

22Oct01 7.6h Introduced ZBL00 and MSTAR stopping powers

24Jan02 7.6j Fixed error in the He on D scattering cross-sections - thanks to Joana Vaz Pinto

13Feb02 7.6k Introduced data types 21 (Garching) and 22 (2048 channel Ortec)

15Feb02 7.7a Quadratic energy calibration

1Jul02 7.7c Molecules with varying stoichiometry

Oct02 7.7d Better roughness model (4), new model (5)

6Nov02 7.7e Tschalar scaling, bug in element search, streamlined routines, better roughness model (4)

1Dec02 7.8ax Double scattering routine introduced, hopelessly bugged

27Mar03 7.8a Stopping power of 1 molecule can be read and/or fitted, also in BI/MCMC. DS still with bug.

8Aug03 7.8c Bug in factor and constant in DEPTH files calculations fixed. DS still with bug

3Sep03 7.8d Introduced formula in one given element

8Sep03 7.8e Double scattering seems to be working (even for grazing angle of incidence) - not really

30Oct03 7.8f Introduced initial guess and BI/MCMC for molecules of unknown composition. DS works really well, although there is some room for improvement on the MS side of things

19Nov03 7.8g Automatic call to MDEPTH - many thanks to Edit Szilágyi; DS about finished.

20Jan04 7.8h Refinements in the DS routine, some new data types introduced, quad factor in the energy calibration may now be fitted.

12Mar04 7.8i A few small bugs were corrected - thanks to Carlos Pascual-Izarra

19Apr04 7.9a Bug in D(3He,4He)p cross section corrected - thanks to Karl Krieger

21Apr04 7.9b Eliminated Z-search, it led to major code structure complexity and no-one ever used it afak

31Aug04 7.9d First version of latent variable for thickness. Sigmoid taken out, but code not deleted

31Aug04 7.9e Sigmoid code deleted

8Nov04 7.9f Complete change of variable structure, to dynamic variables. Introduction of effect of energy spread before interaction in the shape of resonances. MSTAR fortran code included in NDF.

6Jan05 7.9g Small changes and corrections. Introduction of \ in GEO file to comment out lines.

10Feb05 7.9h Fairly ad-hoc correction to low energy signal when using MDEPTH or straggling and counts are lost to negative energies. New pile-up algorithm implemented.

25Feb05 7.9i Bug on NRA d(3He,p)4He cross section corrected. Wielopolski and Gardner pile-up algorithm introduced, thanks to Matej Mayer. Pileup algorithms no longer assume linear calibration.

3Mar05 7.9i Effect of multiple scattering - scattering angle spread and pathlength spread introduced. Spherical and cylindrical inclusions programmed.

22Mar05 8.0a Introduction of PIXE - i.e. link to the LibCPIXE library rewritten by Carlos Pascual-Izarra based on the DATPIXE code written by Miguel Reis. LibCPIXE is free code under a GNU license. NDF has a special license to use it. Some bugs remain in CPIXE.

23May05 Corrected bug in smoothing data when doing BI - thanks to Thomas Lauf. New Wielopolski and Gardner pileup routine including pileup rejection being introduced.

30May05 8.0b CPIXE seems to be working well, without bugs. Corrected bug in convdept routine loss of counts to negative energy. Approximate triple pileup, and PUR introduced in WG pileup algorithm.

11Jul05 8.0b Removed ad-hoc correction to low energy signal when using MDEPTH or straggling and counts are lost to negative energies, since it was not good enough.

20Jul05 8.0c Improved Gaussian description in convolution algorithms: now follows Gaussian down to 0.001% of maximum, Simpson integration of area in each bin.

25Jul05 8.0d Found and corrected a bug in NRA with straggling, introduced in 7.9f - only appeared when NRA was used with straggling.

24Aug05 8.0d Wrapped up the three pile-up routines. Abandoned back-compatibility of pile-up syntax with previous NDF versions.

26Aug05 8.0d Up to 8.0d, when using more than 1000 stopping powers via NDF.TCN, the stopping was not correctly calculated. Introduced transmission geometry for RBS, ERDA and NRA, but still untested. Straggling still not working for transmission geometry.

01Sep05 8.0e User can control random number sequence.

19Sep05 8.0f Mitsuo Tosaki's model for local variations of density introduced.

21Sep05 8.0g Transmission geometry calculations conclusively tested for ERDA and RBS. Only L'Ecuyer screening available for ERDA. Improved low energy yield calculation.

27Sep05 8.0h Low energy yield calculation seems to be working correctly, for Bohr and Chu straggling. Corrected a bug in NRA introduced in v8.0d - not the total yield, but the exact energy distribution was incorrect.

- 29Sep05 8.0i Introduced integration of cross section over energy spread for low energy yield calculation.
- 11Oct05 8.0j Supports WDEPTH, including larger path names. MDEPTH not completely backcompatible (from now on user must input e.g. C: instead of just C) (but user can also input e.g. D:\CODES\iba).
- 18Oct05 8.0m NDF had an internal constrain to max of 16 geometries per sample, now removed.
- 29Nov05 8.1a First version of new code for fitting cross section
- 07Dec05 8.1a non-Ruth cross section: on energies larger than max in file, value is taken as for max energy in file. For energies lower than min in file, 1 is taken. Autolayer option is enforced when effect of energy spread before scattering on resonance is calculated.
- 15Feb06 8.1c The routine to impose constraints was improved (a couple of loopholes were closed); fitting cross sections seems to be working very well.
- 16Feb06 8.1d Andersen screening re-introduced in ERDA. The previous version was very slightly incorrect (effects mostly not noticeable on calculated spectra).
- 23Feb06 8.1e Various bug fixes
- 07Apr06 8.1f Introduced non-Gaussian straggling (gamma function)
- 19Apr06 8.1g Various fixes, slightly updated manual, autol off introduced
- 04May06 8.1g Various fixes, Chu straggling is now default
- 16May06 8.1h Gauss default for NRA and NDP. Straggling calculated for beamfoil.
- 17Jul06 8.1i Small improvement of gamma straggling for large straggling values. Default is now Gaussian, except if the FWHM is smaller than 3 times the calibration gain in RBS and ERDA.
- 13Sep06 8.1j Various fixes, BI of molecular concentration with ? elements in order with new algorithm.
- 12Dec06 8.1l Introduced general geometry, introduced GUPIX output format as input file for PIXE.
- 12Jan07 8.1m Some interlocks introduced (e.g. no low yield correction when roughness is used).
- 17Jan07 8.1o Compound molecules / sub-molecules were iontroduced. First version of new χ^2 .
- 19Jan07 8.2a χ^2 fairly insensitive to forward model problems; and normalised. Can be changed as user option.
- 22Jan07 8.2b Local SimAnn working for both normalised and non-normalised cases. First version of repeat calculation.
- 01Mar07 8.2c Non ionising losses with a good default.
- 13Apr07 8.2d Restrictions to number of channels removed; repeat calculation from file for pile-up implemented.
- 23May07 8.2e Corrected a bug in submolecules/PIXE, introduced flagvar_ in geo files.
- 06Jun07 8.2f Maximum number of samples and of geometries per sample increased from 99 to 359
- 13Jul07 8.2i First version with experimental parameters changing
- 19Jul07 8.2j Reasonably debugged version with experimental parameters changing
- 27Aug07 8.2k Allows funny filters in PIXE - changed slightly meaning of (1) syntax in SPC file
- 30Aug07 8.2l Introduced foil inhomogeneity.
- 26Oct07 9.0a Introduced detector efficiency, corrected some bugs, including in NRA (due to recent introduction of tables for straggling). Better description of large columnar inclusions. Free version as simulator only and without PIXE is implemented, and it is called internally "ALP".
- 31Oct07 9.0b Corrected several bugs, including in the autol and NRA routines.
- 03Dec07 9.0c Introduced verbosity, changed the previously "unused" speed option. Introduced GEO changes in the repeat calculation. Corrected a few bugs, including in ROIs when fitting with a foil present. Readstop and stop codewords were removed. Molstop and elstop now can also be used for reading in a file with (energy, stopping) values.
- 14Dec07 9.0d Handling of user-input molecular and elemental stopping completely changed. Now, not only ZBL90 a1-a8 parameters can be read, but also (energy,stopping) files.
- 23Jan08 9.1a Changed handling of DEPTH output at large energy loss, and corrected some minor bugs.
- 03Apr08 9.1e First version with NRP.
- 08Apr08 9.1f User input charge state of incident beam introduced.
- 14May08 9.1g More or less solid version before changing handling of non-Rutherford cross sections
- 02Jul08 9.1h Removed input of multiple angle non-Rutherford cross section. Completely changed handling of cross sections, to explicitly get rid of interpolating tables. Bugged versions.
- 02Jul08 9.1i First version that seems to be working more or less correctly for all IAEA examples plus NRP.
- 09Jul08 9.1j DS forgot to take into account energy loss in foil and detector dead layer, corrected this.
- 12Sep08 9.1l A few small bugs corrected in DS with dead layer (again!), and cross sections (again).
- 18Sep08 9.1l Introduced the "fast" option for DS, eliminated the "ord" option as it became redundant.
- 09Out08 9.2a First version of DS for ERDA. Only events involving the beam particle and the recoil in the target are considered, e.g. 1) ion scatters off target element then produces a recoil, and 2) ion produces a recoil which then scatters off a target element. Events involving intermediate recoils are not considered, e.g. ion produces recoil which produces a second detected recoil. Recoil events in the absorbing foil are also not calculated.
- 06Nov08 9.2b Second version of DS for ERDA, where recoils produced in the stopping foil are also considered, but only those produced at less than 20° scattering angle.
- 10Nov08 9.2c All directions up to 80° recoils are considered. However, it is assumed that the foil is part of the detector, i.e. all transmitted recoils are detected.

- 02Dec08 9.2c New treatment of angular spread effects in yield, energy loss and kinematic factor.
- 12Mar09 9.2g Corrected some bug in DS, that made NDF “forget” some trajectories in extremely thin self-supporting layers. Such layers can’t exist in real life. Changed default stopping to ZBL00. All file names now can have up to 132 characters including the path. Created repository for SRIM03 calculated stopping powers. This MUST exist before SRIM03 is called, NDF does not check and does not create it.
- 11May09 9.2j Corrected a bug that showed up when the low energy yield option was used in conjunction with resonance calculation with very fine energy step. NDF now supports R33 cross section format
- 15May09 9.2k NRP seems to be working for the most demanding high resolution narrow resonance cases. Introduced Tschalar OFF.
- 15Jun09 9.2l Corrected some bug when reading R33 cross sections.
- 20Jul09 9.2m Introduced SigmaCalc cross sections via a file for a-ERDA of 1H. Also included the angular dependence of this cross sections for the double scattering algorithm, but only in the range 1-3 MeV.
- 18Set09 9.2o At some point, the inbuilt 4He on 2H ERDA cross sections went wrong. Don't know which versions are affected. This is now corrected. Also, SigmaCalc cross sections (or from other source, e.g. experimental) via a file, with the angular dependence, is implemented for any system.
- 24Feb10 9.2q Small changes, including syntax of single-element sub-molecules.
- 25Feb10 9.3a First version of SIMS implemented. Sensitivity not yet a fitting parameter.
- 22Mar10 9.3b BI with SIMS seems to be working. Sensitivity not yet a fitting parameter.
- 18May10 9.3c New option: simultaneously detected different particles with same energy calibration. DS calculation improved for non-grazing incidence of very thin films.
- 31May10 9.3c First implementation of thickness of depleted region of Si detectors. Also, NIL is now always calculated as long as a detector dead layer or an active (depleted) layer is given.
- 02Jun10 9.3c Corrected a bug that made NDF not calculate the effective charge correction to stopping power when SRIM03 was used.
- 22Sep10 9.3e Handling of stopping power much more general. User can input molecular and elemental stopping powers, via a file with data, or via a1-a8 ZBL85 parameters, and including the elements (but not molecules) in the beamfoil, foil, dead layer, and active layer of the detector (assumed to be Si). One single elemental or molecular stopping power can be fitted in RBS. This changed dramatically in v9.3e! It is completely not back-compatible and it is going to work very differently from previous versions.
- 10Nov10 9.3e Peaks and edges are output (for verbosity 1).
- 20Dec10 9.3e Bug in LLD for pile-up corrected.
- 14Jan11 9.3f SigmaCalc files changed definition. This is not back-compatible with previous versions. New code word ebsfiles introduced to define user-preferred non-Rutherford cross section files for given detectors. Handling of user-defined ion charge state changed, also not back-compatible. This may change soon (to introduce charge state of the detected ion) so it is not documented in this manual for this version. See sections 10.8 and 11.1.
- 24Jan11 9.3f LLD is now automatically determined. This automatic determination is overridden either by a use input LLD or by the GEO command lld off. See section 11.5.6.
- 03Feb11 9.3g Reintroduced the point by point profiles for NRA; NDF now checks whether there is an energy inversion or not, and if there isn't, it calculates the pbp profiles.
- 18Mar11 9.3h Introduced the gamma function for roughness, but this is still in tests, it is not active in the released version.
- 09May11 9.3i Output of IDF files.
- 14Jun11 9.4a First version with gamma roughness.
- 16Jun11 9.4b Formula now also works for molecules
- 27Sep11 9.4c Gamma roughness seems to be working, including for PIXE and fitting. New data format 35 that should read any single column format no matter how many channls (being power of 2 might help)
- 20Oct11 9.4c Small correction to ndf.tcn constraints, see section 9.2.
- 30Oct11 9.4d NDP was not working at least since the handling of cross sections changed in 9.1h. It is now working again, see section 5.4.7. Also, the exponential background for NDP can be input by the user, see section 5.4.7 and 6.15.
- 08Nov11 9.4d Pileup changed, now it is recalculated each time the energy calibration changes enough to warrant it. Also, it can deal with negative values of the offset. Also, the first number in the data file should now be the livetime, not the realtime. the realtime can be the second number.
- 29Nov11 9.4d Corrected bug in formula in conjunction with NRA/ERDA
- 16Dez11 9.4e Corrected bug in the pbp profiles with non-Rutherford cross section: it was wrong before. Also, expanded the pbp algorithm to NRA whenever possible, and to cases where more than one reaction is detected. Also now calculates, within the pbp algorithm, the area of an elemental signal in the spectrum. See section 8.3.1. Introduced the new data format 36. See section 5.2.
- 26Apr12 9.4f Function erf now available in the function profile facility (already in previous version).
- 29Apr12 9.4g Moliere, Universal, Lenz-Jensen and Bohr potential screening available.

- 15Jun12 9.4g MSTAR was not working, corrected. WDEPTH and foil had a problem that showed up on rare occasions, corrected.
- 28Jun12 9.4h Introduced keyword "force" in the ROIs (see section 5.3), and also the possibility to input the error in PIXE data (see section 5.10.3).
- 22Aug12 9.4h Small correction in rep calculation with SigmaCalc, clarified GUPIX format 26 is Pixconc.txt. Corrected small bug in NRA cross sections (almost never had effects)
- 19Oct12 9.4i Introduced new option in concentration limits, see section 4.3.2
- 15Nov12 9.4j Now volume fraction of inclusions can be fitted (see section 13.5). The PRF files with inclusions needed to change format, and are not back-compatible with previous NDF versions. Also introduced a first version of fine control of the local search fitting process (see section 7.5.7).
- 14Dec12 9.5a Introduced second version of fine control of local search via file NDFPRF.ORD (see section 7.5.7). This introduces strong changes in how the fitting of volume fraction of inclusions and layer roughness is controlled (see sections 13.4.1 and 13.5.3). The changes are back compatible in terms of existing PRF files, but behaviour changes.
- 31Jan13 9.5b Introduced the READGEO directive in the GEO files, whereby the user can override some of the parameters in the GEO file. This is useful to change the parameters for several spectra at the same time. See section 5.16.
- 23Apr13 9.5e Introduced inhomogeneous active layer thickness, see section 5.15 below.
- 26Apr13 Corrected a small bug in inhomogeneous active layer thickness, and introduced user input maximum number of points in gamma roughness, see section 13.2.
- 30Apr13 Changed pre-defined molecules, see section 4.2.6.
- 20May13 In SimAnn followed by local search, layers of equal concentration are merged before the local search. this does not happen if there is layer roughness models 1, 4.
- 04Jul13 Made subtle changes in the fine control of local search via file NDFPRF.ORD (see section 7.5.7). Now, when the file does not exist, NDF creates one automatically (as it did before) but elements with concentration equal to zero in a given layer do not change in that layer (so the concentration remains 0 is a local search). Also, in the first line, besides the number of layers NDF writes the code "auto", so next time it knows it was an automatically produced file and it will ignore it, looking again at the concentrations of the elements in each layer.
- 15Jul13 9.5e Introduced M lines in PIXE. May be bugged, needs testing with real data.
- 27Sep13 9.5g Corrected bug in autol (when same cross section file input for more than one element or isotope)
- 12Oct13 9.5h Corrected small bugs in M lines (would not fit), and in adding Poisson noise to the simulation. Introduced user input thickness distribution, see section 13.3. Introduced PILESPEED option, see section 11.5.8.
- 19Oct13 9.5h Corrected small bugs here and there, including in NDFPRF.ORD. Importantly, foil thickness inhomogeneity was not well implemented in WDEPTH; and NDF by default was not calculating the signal from layers so thin that the energy loss was lower than around 10^{-8} of the beam energy, see section 6.16.
- 30Oct13 9.6a First version of Valentina Paneta's implementation of the roughness algorithm by Gurbich et al, including local search and BI/MCMC. See section 13.7.
- 10Jan14 BI/MCMC of formula parameters introduced, see sections 4.4 and 14.7. Introduced SPX. See section 7.10 below.
- 10Mar14 Corrected small bug in output messages concerning stopping powers.
- 29May14 9.6b Possibility to scale cross sections given in two column ascii files. See section 11.1.7.
- 09Jun14 9.6c User can now choose equal errors for all channels. See section 7.9.3. Updated the table of isotopes to reflect latest values, including even very minor isotopes.
- 25Aug14 9.6d Effective charge state on the way in introduced again, see section 10.9.
- 03Nov14 Renamed double scattering files, see section 11.4; changed behaviour of SPX, see section 7.10.
- 07Jan15 Bug in non Rutherford cross sections corrected (when given in mbarn in r33 files, depending on the isotopes present; seldom occurring).
- 09Jul15 9.6f Pilespeed in pileup now can be used to make pileup in a user input number of channels. See section 11.5.8.
- 31Aug15 9.6g MS correction for Wim Arnoldbik. See section 11.7.6.
- 02Sep15 MS correction for Wim Arnoldbik with option without cut-off value. See section 11.7.6.
- 30Dez15 Introduced extra elements with different corrections (cross sections, MS, PS, etc). See section 4.6.
- 07Jan16 Corrected small bug in the extra elements feature.
- 30Sep16 9.6i Corrected bug that prevented NDF from using the open source routine (it was not called). Bug introduced at unknown previous version.

2 Getting started

This section is intended to be a quick and dirty reference to the bare essentials. Most, or all, of what is here is repeated and extended elsewhere in the manual. We strive to cross-reference as much as possible. A lot more detail is given in the manual. Please note that this manual refers to the DOS code NDF.

2.1 *Setting up the problem*

You must:

1. Tell the computer which elements are present in your sample.
2. Tell the computer the experimental parameters, ROI for the fit etc.
3. Tell the computer any other information you might have - e.g. non-Rutherford cross sections, corrections to stopping powers, or any info you might have on your samples.
4. Tell the computer which files you want to analyse.

This is done via a series of files described below. The batch file SPC is at the heart of everything.

2.1.1 The batch file: SPC

The batch file controls everything. This can have any name, but has extension spc. We call this “the spc file”. It can be for instance

```
datafile 5.      geofile  structurefile
```

where

datafile is the name of the data file
 5. is the collected charge (see below for the units)
 geofile is the name of a file that contains all the information on the geometry, as well as on calculation method. This can be any name, but the extension must be geo. We call these “the geo files”
 structurefile is the name of a file that contains all information on the sample, such as elements present. This can be any name, but the extension must be str. We call this “the str file”.

In this simple example, one single spectrum was collected from the sample and is analysed. If e.g. three spectra are to be analysed, the SPC file could be for instance

```
datarbs 5.      georbs1  structurefile
dataerda 3.      geoerda
datarbs2 6.56   georbs2
```

So one different GEO file must be defined for each geometry/experiment made. A single structure file means that all data will be analysed simultaneously, with the same sample definition and depth profile.

See chapter 3 for a full description.

2.1.2 The sample structure file: STR

This file defines what your sample is made of. It includes all the elements and/or molecules present, and also the presence of roughness, voids, inclusions, quantum dots, etc.

One STR file must be prepared for each different type of sample. However, if you have many similar samples, all with the same elements present, and only the depth profile is different, the same STR file can be used for all.

See chapters 4 and 13.

2.1.3 The geometry files: GEO

Each GEO file describes one set of experimental conditions, as well as different calculation options. So for simultaneous analysis of multiple spectra from the same sample, you prepare one GEO file per spectrum, as necessary.

See chapters 5 and 6.

2.1.4 The depth profile file: PRF

To make a simulation or a local search starting from an initial guess, you must prepare a file with the depth profile. This is normally called `ndf.prf` (but not always, some advanced functions lead to other names) - see the relevant section. The file looks e.g.:

```
4
1000 1 0 3
200 0.3 8
300 1 1 3
30000 0 0 1
```

Which is a 4 layer target; the first number defines the number of layers; then one line per layer, stating first the thickness (in $1e15$ at/cm²), and then (unnormalised) atomic concentrations for each element/molecule defined in the STR file.

In the current version a maximum of 250 layers are allowed for simulation and local search, and 50 in simulated annealing and Bayesian inference.

See section 7.1.

2.2 Running NDF

You can:

1. Simulate a theoretical spectrum from a given depth profile. This is the main recommended use for NDF.
2. Make a local search optimisation starting from a given depth profile.
3. Run a simulated annealing search. This is advised for advanced users only, as the results may need interpretation.

You can also:

4. Calculate error bars on your depth profiles using Bayesian Inference; this will be done at the end of a fit (not after a simulation).

2.3 Looking at the simulation or fit

NDF generates output files, you must look at them with any graphics routine you like.

2.4 Units in NDF

NDF uses as units of depth 10^{15} at/cm². Values in nm provided in some output files use weighted averages of atomic densities and are meaningless except for pure layers. The user can also specify densities (10^{22} at/cm³) (see section 4.1 below).

From version 9.3a on, NDF can treat some techniques where real depth is measured, such as SIMS or profiling with a lateral microbeam. The depth units used are nm.

NDF uses as units of concentration atomic fraction (from 0 to 1). When the user inputs e.g. Si 1 O 2, NDF will automatically normalise to 1 at.fraction.

NDF uses as units of beam fluence particle- μ C. That is, it is proportional to the number of beam particles hitting the sample and not to the charge. This makes it independent of the charge state of the particles. Suppose you collected 10 μ C with a He⁺⁺ beam. Then the fluence you have to introduce in NDF is $10/2 = 5$ particle- μ C.

This gets slightly weird when analysing NDP data. The units in NDF are still particle- μ C in that case (although the neutron is not known to have any electric charge at all). The user must do the following conversion. Suppose N is the total number of neutrons incident on the sample. $e=1.60217733\times 10^{-13}$ μ C is the electron charge magnitude. Then the user must input (in the .SPC file, see section) $N\times e$. For instance, for 5×10^{10} neutrons, we have 8.011×10^{-3} particle- μ C.

3 The batch SPC file: specifying which files to analyse

3.1 The batch file .SPC

Once you have your structure and geometry files ready, you have to prepare what we call the batch file, that basically tells the computer which files it must fit. You can give this file any name you want, with at least three characters (DOS restrictions apply), but the extension must be .SPC.

The example of a real life .SPC file follows. It is a batch file to analyse four files, all with roughly similar structures:

```
sifeco1. 9.947 g1 s1
sifeco2. 10.506 g1 s1
sifeco3. 10.535 g1 s1
sifeco4. 10.450 g1 s1
```

The charge (in particle- μC - see section 2.4 above) follows the data file name; after that comes the name of the geometry file and of the structure file, without extension: so in this example, the geometry file will be g1.geo, and the structure file will be s1.str. **WARNING:** The input charge value will be changed during the fit. You can turn this off by using NDF.TCN (see section 9 below).

Notice that all the four files in the example above were collected at the same experimental conditions, so the same *geometry* file is given for all of them. The same *structure* file is given to all of them because they are all similar samples, composed of the same elements. Now imagine a more complicated situation. For example, you measured two Si-Fe silicide samples in the same experimental conditions, and three silicon carbide samples, two of which were measured with the same geometry, and the third at a different angle of incidence. The resulting batch file would be

```
sifeco1. 9.947 g1 s1
sifeco2. 10.506 g1 s1
sic1. 10.02 sc1 s2
sic2. 10.13 sc1 s2
sic3. 10.093 sc2 s2
```

In this case, s1.str defines samples with Fe, Co, and Si, and s2.str defines samples with Si and C. Further, g1.geo defines the experimental conditions at which the Si-Fe silicide samples were measured, sc1.geo and sc2.geo the experimental conditions at which the first two (sic1. and sic2.) and the last (sic3.) silicon carbide samples were measured.

Now suppose you measured two similar samples, each in three different sets of experimental conditions (thus collecting three separate data files for each sample), for which the geometry files are g1.geo, g2.geo, and g3.geo. The .SPC file will be

```
s1file1. 5.2 g1 s1
s1file2. 17.3 g2
s1file3. 8.03 g3
s2file1. 10 g1 s1
s2file2. 6.41 g2
s2file3. 9.3 g3
```

Notice that the three files corresponding to sample1 (s1file1., s1file2., s1file3.), all have different geometry files, but only the first one has a structure file. NDF then automatically thinks that the files without an explicit sample structure file are to be fitted together, simultaneously, that is, that s1file1., s1file2., s1file3. all correspond to the same sample as defined in s1.str. As s2file1. again has a sample definition file attached, NDF automatically thinks it is a new sample.

Basically, each line with a sample definition filename attached defines a new fit (new sample). Up to 359 fits are allowed. Each fit can simultaneously analyse several spectra (up to 359) corresponding to the same sample.

We now give a still more complicated example, and there will be more about this in section 3.2 below. Suppose you measured two spectra of one sample, one just normal RBS, the other He-ERDA on a sample with H and deuterium, and the H and D signals come out superimposed. That means you have two particles detected on the second spectrum. The

```
rbsfile 5. rbsgeo s1
erdafile 10. Hgeo
erdafile 10. Dgeo (2)
```

.SPC file must then be

The (2) on the third line indicates that the theoretical spectrum calculated with basis on the geometry file Dgeo is to be added to that of line (2), that is, to the Hgeo one, before calculating the chisquared. This is what we want. And, the

collected charge will automatically be the same for both Hgeo and Dgeo; the value that you input for Dgeo is simply ignored.

3.2 *Multiple detected ions 1 - e.g. TOF-ERDA, funny filters*

Suppose more than one different ion (for instance in TOF-ERDA or NRA) is detected (see sections 5.4.2 and 5.4.3 below), their spectra are superimposed, and you want to analyse the total spectrum. This is what you must do: First, create one GEO file for each detected particle. E.g. in HI ERDA of a Si/TaN sample using 35 MeV ^{35}Cl as incident beam, you need to define four GEO files. One for the backscattered ^{35}Cl (RBS), plus three for the Si, N and Ta (yes, it will be there!) recoils. Then the batch file will look e.g. like

```
sample.rbs    10.    rbsgeo    sample
sample.si     9.8    sigeo (1)
sample.n      9.9    ngeo (1)
sample.ta     10.    tageo (1)
```

The (1) means that the corresponding spectrum will be added to spectrum number 1 for that sample (in this case, sample.rbs - rbsgeo). So, lines 2 to 4 will lead to the ERDA simulated spectra being added to line 1, that is the RBS simulated spectrum, and then a single spectrum will be fitted.

Suppose that you were using ToF ERDA and you could separate the N recoils but not the Si or Ta ones from the backscattered spectrum. Then the batch file could look like

```
sample.n      10.    ngeo      sample
sample.rbs    10.    rbsgeo
sample.si     9.8    sigeo (2)
sample.ta     10.    tageo (2)
```

Whereby the RBS and the Si and Ta ERDA simulated spectra would be added up, and it would be fitted simultaneously with the N ERDA spectrum.

Note that different charges are allowed (from 8.2k on; in earlier versions, only the charge of the first one was used, and the other values were simply ignored). This is because the detector may have different efficiencies for different ions, and also for funny filters in PIXE (see section 5.12.3 below). During the fit, the ratio of the charges is kept constant at the value given in the SPC files. That is, the ratio will not be changed.

Note that each ion can have a different energy calibration. This is because different species behave differently in the detecting system.

WARNING: A spectrum cannot be attached to another one with a higher number. So e.g. the following is not allowed:

```
sample.n      10.    ngeo      sample
sample.si     10.    sigeo (4)
sample.ta     10.    tageo (4)
sample.rbs    10.    rbsgeo
```

3.3 *Multiple detected ions 2 - same energy calibration*

In some cases you may wish to do the same as in the previous section, but enforcing the same energy calibration for the different detected ions. This will be the case if a very well determined energy calibration is available, and if the pulse height defect is correctly calculated (see section 5.13 below). This is done e.g.

```
sample.rbs    10.    rbsgeo    sample
sample.si     9.8    sigeo <1>
sample.n      9.9    ngeo <1>
sample.ta     10.    tageo <1>
```

which will do exactly the same as was described in the previous section for (1) instead of <1>, the only difference being that the energy calibrations given in sigeo.geo, ngeo.geo and tageo.geo are ignored and the one given in rbsgeo.geo is used instead.

3.4 *Simultaneous detectors*

Furthermore, suppose you measured HI-ERDA of a Ti C O H sample. The backscattered particles plus Ti recoils aren't separated, and you analyse them as a single spectrum. But, the O, C and H spectra are treated separately. You can do all this in NDF with one of the SPC files (which work in a completely equivalent way):

```
ti_a.rbs 0.84 rbs15 ti
ti_a.rbs 0.83 ti15 (1)
ti_a.o 0.84 o15 [1]
ti_a.c 0.84 c15 [3]
ti_a.h 0.86 h15 [3]
```

```
ti_a.rbs 0.84 rbs15 ti
ti_a.rbs 0.83 ti15 (1)
ti_a.o 0.84 o15 [1]
ti_a.c 0.84 c15 [1]
ti_a.h 0.86 h15 [1]
```

The difference between "ti_a.rbs 0.84 ti15 (1)" and "ti_a.o 0.84 o15 [1]" is that (1) means the spectrum is superimposed, while [1] is treated independently, with its own χ^2 . In both cases, the charge ratio will be kept throughout the fit.

WARNING: A spectrum cannot be attached to another one with a higher number. So e.g. the following is not allowed:

```
sample.n      10.      ngeo      sample
sample.si     10.      sigeo [3]
sample.rbs    10.      rbsgeo
```

3.5 Same energy calibration

Suppose you measured several spectra of the same sample the same day, you didn't touch the electronics, and you simply have complete certainty that the energy calibration is the same in all of them. You want to let NDF fit the energy calibration, but you want it to remain the same for several of the spectra. You can do this in NDF:

```
f_rbs1 5.00 rbs15 ti
f_rbs2 6.30 rbs2 {1}
f_erda 4.8 erd (1)
```

In this case, the {1} in the second line means that the energy calibration in the file rbs2.geo will be ignored, and it is that of rbs1.geo that will be used. (As for the third line, this would mean that an ERDA spectrum was collected simultaneously with the first RBS spectrum. Since the detected particle is different, the energy calibration will of course be different.)

WARNING: A spectrum cannot be attached to another one with a higher number. So e.g. the following is not allowed:

```
sample.n      10.      ngeo      sample
sample.si     10.      sigeo {3}
sample.rbs    10.      rbsgeo
```

3.6 Verbosity

The verbosity flag defines what output are generated, as detailed in chapter 8. The default, which is verbosity=0, requires no action by the user. Higher verbosity leads to more outputs being generated. This is made by including a fifth element in the first line defining a sample, just after the STR file. For instance,

```
sample.rbs    10.      rbsgeo      sample 2
sample.si     9.8      sigeo (1)
sample.n      9.9      ngeo (1)
sample.ta     10.      tageo (1)
```

Means that verbosity=2 is defined.

3.7 Advanced users

See section 7.7 below.

4 The STR file: specifying the elements and/or molecules

4.1 The structure file *.STR

You do this by means of a file, that we call the structure file. You can give this file any name you want (DOS restrictions apply), but the extension must be .STR. Here is one example of one such structure file:

```

100 5000
5
Au
0 500
0 1
58Ni
0 0
0 .5
Si 1 O 2 6.6
0 0
d
Si
0 0
s

```

4.1.1 Minimum and maximum layer thickness used in the fit

The number in the first line is the minimum and maximum layer thickness that NDF will use in the fit (in 10^{15} at/cm²). NDF will never try to introduce layers outside these limits. The second value (maximum layer thickness) can (and often should) be omitted altogether, and generally it is not necessary to introduce it at all. Note that small values will mean NDF might have to introduce lots of thin layers to make up e.g. a thick layer.

4.1.2 Number of elements / molecules in the sample

The second line is the number of elements or molecules present. In this case, we have 5. Maximum is 92

4.2 Telling NDF what the sample is made of

After the first two lines, come blocks of three lines each, each block corresponding to one element or molecule:

The first line in a block defines the element or molecule. The second line defines a depth range where that element can exist. And the third line defines the concentration range in which it can exist. These are explained in detail in the next subsections.

4.2.1 Defining an element

If you want to have a given element in the fit, the first line in a block is the chemical symbol of the element. If you want to have say gold in your sample, just type in AU, Au, or au. The natural isotopic composition will be used.

4.2.2 Defining a specific isotope

You can also introduce specific isotopes, such as ²⁹Si or ⁵⁸Ni. If you are analysing a sample where an element as an isotopic ratio different than the natural one, you can specify e.g. two isotopes separately (so each one will have its own three-line block).

4.2.3 Defining a molecule or a compound of known composition

Suppose you know that you have a given molecule in your sample. E.g. you could have a silicon dioxide layer. You can introduce this in NDF by typing "si 1 o 2" (or "Si 1 O 2" etc). In this way, NDF fit the molecular composition as a whole, instead of fitting the separate Si and O amounts. This ensures you have a physically meaningful molecule in your fit.

You can have more complicated molecules as well: C 2.97 O 1 N 2.3 or whatever else you fancy (with a maximum of 78 characters including blanks in the molecule's name, and a maximum of 20 elements).

You can also define the density of the molecule, in 10^{22} at/cm³ units, for instance in
Si 1 O 2 6.6

The density of SiO₂ is specified as 6.6×10^{22} at/cm³.

WARNING: A maximum of 132 characters is allowed for the length of the molecule definition string.

4.2.4 Defining a molecule of unknown composition

You can also have a molecule of unknown composition, such as $\text{Al} \text{ ?=1 } \text{N} \text{ ?=1 } \text{O} \text{ ?=1}$, in which case NDF will fit the molecule's composition. You may know the ration between some elements, and $\text{Al} \text{ 2 } \text{O} \text{ 3 } \text{N} \text{ ?=1}$ is also allowed (the ration of Al:O will be fixed).

The values given are taken as initial guess.

BI/MCMC (see section 14 below) will also change the composition in this case, and the result will be given in the res file.

WARNING: A maximum of 132 characters is allowed for the length of the molecule definition string.

4.2.5 Defining compound molecules / sub-molecules

You can have compound molecules such as

$(\text{Si} \text{ 1 } \text{O} \text{ 2 } \text{6.6}) \text{ 2 } (\text{Co} \text{ 20 } \text{Fe} \text{ 80 } \text{O} \text{ 90}) \text{ 3}$

Which will be, in terms of atomic composition:

Si 0.133 Co 0.063 Fe 0.253 O 0.551

The densities of each sub-molecule can also be input. In the example above, SiO₂ has a density of 6.6×10^{22} at/cm³. The total density of the compound molecule cannot be input. It will be calculated as the average (weighted by the amount of each sub-molecule, in the example above 0.4 for SiO₂ and 0.6 for Co₂Fe₈O₉) of the densities of the constituting sub-molecules.

For sub-molecules that are one single element, no concentration is given:

$(\text{Zn} \text{ 1 } \text{O} \text{ 1}) \text{ 99 } (\text{Pb}) \text{ 1}$

or

$(\text{Zn} \text{ 1 } \text{O} \text{ 1}) \text{ 99 } (\text{Pb} \text{ 3.296}) \text{ 1}$

To make explicit the Pb density.

WARNING: This changed in version 9.2q! In previous versions, the concentration was mandatory. So NDF is not back-compatible in cases where sub-molecules with a single element are defined!

Both the atomic concentration within each sub-molecule, and the concentration of each sub-molecule as a whole, can be fitted independently:

$(\text{Si} \text{ 1 } \text{O} \text{ ?=2 } \text{6.6}) \text{ ?=2 } (\text{Co} \text{ 20 } \text{Fe} \text{ 80 } \text{O} \text{ ?=90}) \text{ ?=3}$

Note that in the second sub-molecule, the Co:Fe ratio is fixed as 1:4. BI will also be done on these parameters, and the result will be given in the res file.

WARNING: A maximum of 132 characters is allowed for the length of the molecule definition string. And a maximum of 5 sub-molecules is allowed in each molecule. Also, a space must always be given between the commas and the elements and their concentrations.

4.2.6 Pre-defined molecules and substances

The file compound.dat has a list of substances with their common name. For instance,

"Silicon Dioxide"

2

"Si" 1

"O" 2

2.32

You can use this name in the STR file. So, instead of a logical element called e.g.

Si 1 O 2 6.6

0 0

0 0

(the 6.6 is the density in $1e22$ at/cm³ units)

you can specify

silicon dioxide

0 0

0 0

You can also specify pre-defined molecules as sub-molecules, but each sub-molecule may have only one pre-defined

molecule, and nothing more. Also, the density of the pre-defined molecule cannot be changed in the str file. For instance, this is valid:

```
( silicon dioxide ) ?=1 ( Bakerlite ) 2 ( Al 2 O 3 8.3 ) ?=1
```

but the following are not:

```
( silicon dioxide 6.6 )
( silicon dioxide ?=1 Bakerlite 3 ) 1
( silicon dioxide ?=1 Al 3 )
```

4.2.7 Specifying the density of an element or molecule

You can specify a certain density for an element or molecule. Just the density value, in 10^{22} at/cm³, after the molecule description, e.g. Si 1 O 2 6.6 for silicon oxide (with $\rho=6.6 \times 10^{22}$ at/cm³), or C 17.6 for diamond.

the density is actually used in real depth profiling techniques such as SIMS or lateral microbeam PIXE (see section 5.4.6). In this case, if used together with some traditionally IBA technique, the density can actually be determined. Then, it can be a fit parameter. This leads to the syntax e.g.

```
Si 1 O 2 ?=6.6
or even in sub-molecules
( Si 1 O 2 6.6 ) 2 ( Co 20 Fe 80 O 90 ?=8 ) 3
```

Note that NDF will only actually try to fit the density if SIMS+some other IBA are fitted simultaneously!

4.3 Specifying depth and concentration ranges for a given element

Depth and concentration ranges where each given element may exist can be given to NDF. They don't have to be given, but if you have extra information (for instance, you know there is a surface oxide in a Si sample), then you should introduce it as it will make NDF converge faster and more accurately.

Do not make the constraints too rigid - if you think a given element exists only in the depth range 200 - 5000, put as limits for instance 0 - 10000. Often the samples are not exactly what you expect them to be.

Also, make sure you don't define impossible samples. If you have two elements and specify that one exists in range 0-500 and the other in range 600-1e10, then nothing is allowed in range 500-600, and the fit is going to fail miserably.

4.3.1 Defining the depth range

The second line defines the depth range (in 10^{15} at/cm²) within which the element can exist. The fit will never try to introduce the element or molecule specified outside this range. Typing "0 0" means no limit. Please note that if you type in tight, surface-near, limits for all elements present, NDF will still have to find something to put at high depths, with consequent silly results.

4.3.2 Defining the concentration range

The third line defines the concentration range (from 0 to 1) within which the element can exist. The fit will never try to change the concentration of the element or molecule specified outside this range. Typing "0 0" or "0 1" means no limit.

You can also specify that one given element or molecule must exist in delta layers only, that is, with either 0 or 1 concentration. This will mean this element will always be in sharp layers. To do this, instead of typing in a concentration range, just type in a "d".

You can also tell NDF that you wish one particular element or molecule to be the substrate of the sample. To do this, instead of typing in a concentration range, just type in a "s". Then, NDF will automatically create a thick layer with this element or molecule only, at the bottom of the sample. In this case, you should give either no limit on the depth or a very large value.

During a local search, you may want to start from a given profile where an element only exists in some of the layers, but with a minimum and maximum concentration. If you specify as concentration limits e.g.

```
0.1 .2
```

NDF will force this through all the layers where the element can exist, including those where it was zero. You can specify instead

```
0.1 0.2 d
```

and NDF will only enforce these constraints where the concentration started as larger than zero; if it was zero from the beginning, it will continue zero.

WARNING: Sometimes conflicts between different concentration/depth limits can occur. In those cases, which are very rare, the user-defined limits may not be respected. This can also lead to unexpected increases in the chisquared.

4.4 *Specifying a concentration formula for one given element or molecule*

You can tell NDF that you wish one particular logical element to have its concentration defined by a formula. This will also work for molecules. To do this, you must type in the second line (that normally defines the depth range) "f" or "F". Then, in the third line (that normally defines the concentration range) you must type the name of the file that will contain the family or definition. An example is given below:

```
Co
f
gaussian.fr
```

The file will contain the formula, as well as initial values for the parameters, as in the example given below:

```
a0*exp(-(t-a1)^2./(2.*a2**2.))
0.5 0 1 1
1000.0 2000 1
200. 1. 500. 1
```

The parameters (that will also be fit parameters in SimAnn or local search) must be named a0, a1,... a9. A maximum of 10 parameters is thus allowed. The formula may also depend on t, which is the depth in 10¹⁵ at/cm² units. Besides the parameters, the family thus supports brackets as well as all the common operators. The maximum length of the formula is 250 characters. The operators supported are (consults a Fortran manual for the definitions):

+, -, *, /, ^ (equivalent to **), sin, cos, tan, asin, acos, atan, sinh, cosh, tanh, sind, cosd, tand, log, log10, nint, anint, aint, exp, sqrt, abs, floor, erf (error function)

After the formula as such, come lines, one per parameter (excluding t), starting with a0 and continuing. In each line, the first number is the initial value for the parameter. The two following numbers are the limits of variation in the fit. The final number is 0 if the parameter is not allowed to change during the fit, or 1 if it can change during the fit.

Please note that only one logical element can be defined through a formula. Also, at least one other element or molecule must be present in the sample definition. The composition of the element defined through the formula is directly the value taken by the formula, clipped to the interval [0, 1]. The composition of the other elements will be scaled accordingly. The composition of a given internal sublayer at depth t is then set to:

$$\text{composition} = \text{formula}(t) \times \text{composition}_{\text{element_from_formula}} + [1-\text{formula}(t)] \times \text{composition}_{\text{matrix}}$$

As an example, suppose that you have the following structure file:

```
100
2
Si 1 O 2
0 0
0 0
Co
F
gaussian.fr
```

Where the formula evaluates to zero, the composition of the sample will be Si 0.3333 O 0.6667 Co 0. Now take depth 900. The composition of cobalt is 0.441, because the formula evaluates to 0.441 at that depth. Thus, the composition of Si will be (1-0.441)×0.3333 = 0.1863, and the composition of O will be (1-0.441)×0.6667 = 0.3725.

WARNING: Please note that the concentration constraints for the matrix elements apply to the matrix composition, before the formula has been taken into account. This means that the effective concentration of a matrix element may, in some cases, be outside the constraints input by the user.

WARNING: Please note that in a simulation or local search (starting in ndf.prf, see section), the ndf.prf file must include a column for the element defined by the formula. This column should consist of zeros.

WARNING: Please note that the density values given in the result file are calculated for the matrix, not considering the

element defined by the formula. **WARNING:** Also, Bohr or Chu straggling (see section 12.4) will not take into account the element defined by the formula.

WARNING: Suppose you define a formula for logical element "Al 1 N 1", and the matrix is "Ga 1 N 1". NDF needs to distinguish between N in AlN, that has its concentration defined by the formula, and N in GaN, that has its concentration defined by the matrix in the normal way. Therefore, NDF will treat these two "N" separately, and in the result file, they will show up separately. Also there will be separate partial spectra for these two "N".

Finally: the uncertainty of the formula parameters can be calculated with Bayesian inference, see section 14.

4.5 Specifying the presence of roughness, voids, inclusions, quantum dots

This is also done in the str file. See chapter 13.

4.6 Extra "fake elements"

Introduced at the request of heavy duty users. If the same element is included in more than one logical element (molecule), NDF first gets the total concentration and then calculates the spectrum for that element. This means that any corrections, such as MS, PS, non-Rutherford cross sections, etc., are the same for all of them. The request was to be able to have different corrections for the same element. The purpose is when one element is used as substrate, where it is convenient to have e.g. a dedicated MS corrections (e.g. to fake channelling), but the same element is also present in the rest of the sample.

The solution was to introduce fake elements in the STR file. For instance, a logical element called "Wa", "Cj" or "Aw" can be introduced. This must then be defined in a new file (located in the NDF directory) called "element_extra.dat". This must include all extra fake elements, and it is a two column file: first column, the name of the fake element; second column is the actual element.

For instance, if you want an extra Si for the substrate, you could introduce the logical element Cj in the STR file, then have element_extra.dat with:

```
Cj Si
```

Then in the GEO file you could have for instance

```
Si crosssection1.r33
Cj crosssection2.r33
Cj
500 1.5 1
```

note: In the RES file, everything will come out as "Si". This is confusing and misleading. But the entire option is confusing so either you are an expert and know how to handle it and how to avoid being misled in the RES file, or you should not use this.

5 The GEO file: specifying the experimental conditions

5.1 The geometry files *.GEO

You specify the experimental conditions, as well as calculation options, by means of a file, that we call the geometry file. You can give this file any name you want (DOS restrictions apply), but the extension must be .GEO. Here is one example of one such file for helium RBS at 1.5 MeV:

2	(line 1: data format. See section Error! Reference source not found.)
50 400	(line 2: Region of Interest for fit - up to 5, only 1 for data types 11 and 12. No comma allowed)
4He 4He	(line 3: beam in, beam out. 35Cl Si would be heavy ion ERDA, detecting Si)
1500	(line 4: beam energy in keV; for NDP, cross section in milibarn)
17	(line 5: FWHM energy resolution in keV)
ibm	(line 6: geometry: ibm, cornell, or general.)
0 163	(line 7: angle of incidence - 0 means normal - and scattering angle; exit angle for general geo)
7.2	(line 8: detector solid angle in msr)
2.653 80.7	(line 9: energy calibration, gain in keV per channel and energy at channel 0 in keV) quadratic energy calibration is also possible.

This example is exactly the same, except for the compression of the data that will be made in the fit. Data format type 2 has 512 channels, but here they will be compressed to 256. Another difference is that the FWHM is not constant, it changes linearly with detected energy: $FWHM = 17 + 0.01 E_{\text{detected}}$

2 256	(line 1: data format. See section Error! Reference source not found. ; numer of channels after ompression (power of 2)
50 400	(line 2: Region of Interest for fit - up to 5, only 1 for data types 11 and 12. No comma allowed)
He He	(line 3: beam in, beam out. "He" means average mass will be used, which is not accurate!)
1500	(line 4: beam energy in keV; for NDP, cross section in milibarn)
17 0.01	(line 5: FWHM energy resolution in keV, and linear term)
ibm	(line 6: geometry: ibm, cornell, or general.)
0 163	(line 7: angle of incidence - 0 means normal - and scattering angle; exit angle for general geo)
7.2	(line 8: detector solid angle in msr)
2.653 80.7	(line 9: energy calibration, gain in keV per channel and energy at channel 0 in keV)

Both examples include only those things that are mandatory in the geo files. Many other optional codewords, specifying special experimental conditions or calculation/simulation options, can also be given. These come always after the mandatory 9 lines.

5.2 Data type: line 1

You have to specify which of the many data types supported by NDF that you are using. This is the number you have to put in the first line of the geometry files. We would like to introduce RUMP format, but we would need easily usable Fortran code for that.

0	512 channels in single column
1	512 raw XRBS output (Surrey format)
2	512 normalised XRBS output (Surrey format)
3	13 line header, 512 channels in single column
4	1024 channels as 128 lines of 8 columns
5	512 channels in 512 lines and 2 columns: second column is read
6	1024 channels in 1024 lines (single column)
7	1024 channels as 256 lines of 4 columns
8	13 line header, 1024 channels in single column
9	13 line header, 256 channels in 256 lines and 2 columns: second column is read.
10	256 channels as 64 lines of 4 columns.
11	2048 channels in two columns, second is read. Changed from v8.2d on.
12	Limited and unknown number of channels. Two columns, second is read.
13	13 line header, 256 channels in 256 lines of two columns: second is read
14	Ortec binary data. 12 bytes, live time in 20 ms increments (4 bytes), 16 bytes, 512 4 byte integers
15	Ortec binary data. 12 bytes, live time in 20 ms increments (4 bytes), 16 bytes, 1024 4 byte integers, will be compressed to 512.

- 16 PNL ASCII data. 5 lines header, the first one is 37 characters then the livetime (i6). Then follow 1024 channels in two columns, second is read
- 17 Unused. Reserved for PNL binary
- 18 512 line header, 512 channels in 512 lines and 2 columns: second column is read
- 19 1024 channels in 1024 lines and 2 columns: second column is read
- 20 Header: 1st line tells how many in total in header. 128 channels in 3 columns: 3rd is read.
- 21 Garching: 10 line header, 1022 channels in lines of 10 channels (last line 2). Two channels with zero counts are introduced at the end by NDF.
- 22 Ortec binary data. 12 bytes, live time in 20 ms increments (4 bytes), 16 bytes, 2048 4 byte integers.
- 23 Redundant from v8.2d on. Equal to 22.
- 24 Header: 1st line tells how many in total in header. 512 channels in 3 columns: 3rd is read.
- 25 2048 lines, single column.
- 26 Axil output for use in conjunction with LibCPIXE (see section 5.4.5 below).
- 27 1024 raw XRBS output (Surrey format)
- 28 GUPIX output (format Pixconc.txt in GUPIX 96 or gupix_stdout.txt in GUPIX Win2007) for use in conjunction with LibCPIXE (see section 5.4.5 below).
- 29 1024 channels in 64 lines of 16 values each
- 30 256 channels single column
- 31 32768 channels in single column
- 32 NRP data format in two columns: beam energy, yield
- 33 SIMS (or other real depth profiling technique) data format in two columns: depth in nm, normalised yield or any other quantity measured that can be somehow translated into an amount or concentration of material. Data in this format will be treated in log scale, i.e., they are assumed to have a faint relationship to elemental concentrations due to problems such as the strong non-linear dependence of sputter yields with sample composition.
- 34 SIMS (or other real depth profiling technique) data format in two columns: depth in nm, normalised yield or any other quantity measured that can be somehow translated into an amount or concentration of material. Data in this format will be treated in lin scale, i.e., they are assumed to have a clear relationship to elemental concentrations.
- 35 "Universal" single column format. Any number of channels, preferentially a power of two.
- 36 512 channels. 3 line header, then 10 columns. Each line has 10 channels. Last line has 2 channels

5.2.1 Data compression

You can also specify in this line a given compression (see section 7.3.2) for the corresponding spectrum. Suppose you specified data type 29, which has 1024 channels. Suppose you are analysing many spectra simultaneously, and that you want to use 256 channel compression for the other spectra (because they don't have much resolution). You can use compression 2 (leading to 246 channels as default) when running NDF (see section 7.3.2), but if you want to keep the 1024 channels for this one, write simply

```
29 1024
```

This will override the default.

5.2.2 Real time and live time

Please note: for all data formats other than 1, 2, 4, 12, 14, 15, and NRP and SIMS, NDF will think that the first channel in the data is the real time in seconds, used for the pileup correction. See section 11.5 below. This is important, as a fake pileup correction will lead to bad results. For data type 4 (Lisbon), NDF thinks the first channel is the live time, and the second the real time. The real time is used for the pileup correction, while the live/real ratio is used to correct the nominal charge.

5.3 ROI: line 2

If you are doing a simulation only you don't have to bother with this line, just put two integer numbers. If you are making a fit, you MUST - the fit will only care about the channels within the ROI. Some subtle points:

You can specify multiple ROIs, up to 5, such as

```
100 200 250 300 290 500
```

In this case there are three ROIS; note that channels 201 to 249 are ignored, as well as all channels below 100 and above 500. Note also that channels 290-300 are present in both the second and third ROI - so they will be counted twice to calculate the χ^2 ! This is a strange way of giving more relevance to that part of the data.

For NRP and SIMS, the ROIs are the data point number.

Inputting 0 0 makes NDF use the entire range available for calculation.

WARNING: Even more strange and leading to strange results if you don't pay a lot of attention to this: The ROI is what NDF calculates before convolution with the energy resolution. So the yield before the first channel in the first ROI is never calculated. After convolution, the yield around this channel will look like a smoothly increasing integrated Gaussian function. This region is of no use for the fit. So some channels, from the first ROI channel, to the channel where the effect of convolution is no longer visible, will be ignored in the χ^2 calculation. That is, some channels are lost. If straggling is very strong, it can be a lot of channels. The res result file will output the real ROI used in the fit.

There is one way to force NDF to use the actual ROI that you input. It is to write instead of e.g.

```
100 500
```

```
force 100 500
```

This may lead to trouble, as explained above.

5.4 *Experimental technique: line 3*

5.4.1 RBS

If the incident and detected ions specified in the geometry file are the same, NDF will automatically assume it is an RBS experiment. The same happens in a single ion entry is input here.

You should use the correct isotope, i.e. 4He and not He - otherwise the average elemental mass will be used, leading to extra inaccuracy.

5.4.2 ERDA

The recoil ion must be specified in the geometry file, as indicated above. If the incident and detected ions are different, NDF will automatically assume it is an ERDA experiment. If they are the same, it will automatically assume it is RBS.

5.4.3 Non-resonant NRA

For NRA, you have to introduce the whole reaction together with the Q value (in keV), e.g. "d 3He 4He 1H 18352 1H" if you are using a 3He beam to profile deuterium, and you measure the proton that is generated with Q=18.352 MeV. In this case, you must also introduce the cross section values in mbarn - section 11.1 below shows how this is done.

Cross sections are calculated internally for D(3He,p)4He and D(3He,4He)p, following [1] (they can be turned off simply by inputting different files as described in section 11.1 below). Otherwise, if no cross-sections are provided, NDF will assume they are equal to 1 mb/sr (differential cross section in the laboratory system) independently of energy.

It is important to specify the isotopes since the calculations depend on the small mass differences between in and out channels.

The cross section will have to be input by the user.

5.4.4 NRP (resonant NRA, PIGE, whatever)

For NRP, the technique is actually defined by the data format in line 1. Any format, such as 32, that is defined as "NRP data format", will make NDF calculate NRP.

In this line you have to introduce the incident particle and the target particle that suffers the reaction, e.g. "1H 7Li" for protons reacting with 7Li.

In this version of NDF, it is assumed that the detected particle has an infinite range. If the detected particle is stopped in the sample, this effect will be ignored.

The cross section will have to be input by the user.

For NRP, where the system FWHM is normally input (line 5 in the GEO file), the value there will be taken to be the beam energy spread plus Doppler effect. If a second value is given in line 5, this will be taken as the resonance width and used to help determine the integration step.

Please note that most NRP codes assume constant stopping and constant straggling. NDF does not, it will make the calculations energy-dependent. This makes sense for thick targets, but in fact not much sense for high resolution surface layer analysis, where it simply leads to slow calculations without extra accuracy.

5.4.5 PIXE

The freely distributed NDF does not currently support PIXE.

NDF uses the C LibCPIXE library rewritten by Carlos Pascual-Izarra based on the DATTPIXE code [2,3] written by

Miguel A. Reis. LibCPIXE is free code under a GPL license: see <http://cpixe.sourceforge.net/>. An explicit limited permission from its authors has been obtained to link NDF with LibCPIXE. See the informative LibCPIXE Manual "LibCPIXE: the Physics behind it" on either website (LibCPIXE or NDF). Please contact the authors for further details:
 Carlos Pascual-Izarra: carlos.pascual@uam.es
 Miguel Reis: mareis@itn.pt

NDF recognises PIXE by putting in line 2 of the GEO file
 1H x

LIMITATIONS:

- LibCPIXE is only for H and He. This is because the cross-section functions are quite different for heavy ions and are not implemented.
- Secondary fluorescence is not calculated since it is not possible with the current algorithms to correctly extend the implementation for a single layer to multiple layers in the general case (other programs also only calculate fluorescence approximately for multiple layer samples in the general case, since they are limited to the case where each element in the sample can appear in only one layer).
- Only the Ka, La and Ma lines are used for fitting. This is because for these lines the production cross-sections are well known (since fluorescence has a relatively small effect). This is a general limitation on accurate PIXE due to uncertainty of these fundamental parameters.

LibCPIXE does not simulate and NDF/LibCPIXE does not fit the raw PIXE spectra, but the areas of the relevant X-ray lines, using as input the output file of AXIL [4]. The data type in the GEO file must be 26 (see section section 5.2 above). One example of a geometry file (with a two-layer foil in front of the detector) is given below.

Uniquely for PIXE programs, NDF/LibCPIXE is capable of a self-consistent analysis of multiple PIXE spectra from the same sample, together with (optionally) other IBA (RBS/EBS/ERD/NRA) spectra. NDF has one formal limitation for multiple PIXE spectra (but not for the other techniques): the foil in all PIXE spectra must have the same foil elements, although its thickness and composition can change between detectors (hint: suppose you measured 2 PIXE spectra, one with and one without foil. In the one without, put a 1.e-20 as thickness. Also, if you used different foils in different PIXE experiments of the same sample, you can always put in both foils all elements, and put as concentration 1.e-20 for those elements that are not there).

26	(line 1: data format. See section 5.2)
50 400	(line 2: ignored in PIXE)
H x	(line 3: beam in is a proton, beam out is X-rays)
1500	(line 4: beam energy in keV)
17	(line 5: ignored in PIXE, but must be here)
ibm	(line 6: geometry, either ibm or cornell. Sorry, general geometry is not implemented)
0 163	(line 7: angle of incidence - 0 means normal - and scattering angle)
7.2	(line 8: detector solid angle in msr)
calib.det error	(line 9: name of the detector calibration file, for format see below; and, facultatively, an error factor for the PIXE data for this geometry, where 1 is 100%. Default is 10%)
foil 2	(PIXE allows energy filtering foils with many layers, while RBS/ERDA/NRA one single layer!
30000 C 1 H 1	this means that the way of introducing the foil is slightly different in PIXE)
5000 Al 1	

Format of detector calibration file

The detector calibration file contains the instrumental function, that is, the detector efficiency as a function of energy.

This can be determined as usual using a spectrum from a certified standard. The efficiency is specified as a two-column file of energy (in keV) and the absolute efficiency (between 0 and 1) (that is, of a 4π sr detector) at that energy; and (optionally) by specifying the efficiency for a specified X-ray line (see below). The symbolic information, if present, will override the numerical information. Internally, LibCPIXE converts the numerical information to symbolic information, but we give it in this format to emphasise to the user that we are effectively interpolating the calibration function.

The keyword
EFFIFILEVERSION 3
 Must be present

The keywords are *DETECTORCURVE* (mandatory) and *LINEEFFICIENCIES* (optional) (in that order). (LibCPIXE is

case insensitive and accepts any blank space separator -- tab, space or end of line. The number format is free: integers, reals or scientific notations are accepted.)

The format accepted after the *LINEEFFICIENCIES* keyword is: one line with 5 items per X-ray line: zero, atomic symbol, "Ka" OR "La", efficiency, energy in keV. Using this format the efficiency for the the K line of an element (for example) can be changed without specifying a value for the L line (which will be determined from the *DETECTORCURVE* section). The energy of the line must be given as well, and it can even be used to change what is in the database.

Note that although the sum of the α_1 and α_2 lines is used, this is without any loss of generality since it is the X-ray line areas extracted by AXIL that are used instead of the raw spectrum.

Typical use of NDF/ LibCPIXE to obtain and use the detector calibration file

To find the detector efficiency function first run a simulation of the standard sample with no *LINEEFFICIENCIES* and with a *DETECTORCURVE* entry of "0 1 100 1". This sets the efficiency at 100% for all energies (between 0 and 100keV). The efficiency function is then obtained by dividing the calculated line areas by the observed ones. This can then be smoothed and interpolated as necessary (since it will be as sparse as the particular standard used), and re-entered explicitly as a *DETECTORCURVE* entry.

Alternatively, the simulation can be run with "foil layers" entries in the geometry file to model the detector electrode and dead layer. This will give a more accurate representation of the low energy efficiency behaviour, and which the user can adjust manually to fit the data better.

To avoid anomalous behaviour at the absorption edges, the user must give two entries at the energy of the absorption edge in the *DETECTORCURVE* section, where the energy is represented by the same string. Additionally the user must give the efficiency at the absorption edge energy symbolically in the *LINEEFFICIENCIES* section.

Note that at present the high energy efficiency (which falls due to the finite detector thickness) is not modelled.

Files that must be present in the WiNDF directory

The following files must be in the WiNDF directory, which is the same directory where ndf.exe and the stopping files are:

xabs.abs
SC_PRA74.fck
xener.dat

Scoef.95a
Scoef.95b

The last two mean that LibCPIXE uses ZBL95 stopping only, no matter what the user specifies.

5.4.6 SIMS (actually, any real depth profiling technique)

In real depth profiling, some signal is measured as a function of real depth, as opposed to areal density as in most IBA techniques. We call this "SIMS", but it can be any such technique, for instance when using a micro beam on the lateral side of the sample.

For SIMS, the technique is actually defined by the data format in line 1. Any format, such as 33 and 34, that is defined as "SIMS data format", will make NDF calculate SIMS.

In this line you have to introduce the incident particle and the target particle that suffers the reaction, e.g. "1H Au" for a proton beam used laterally to measure the X-ray yield from Au. In version 9.3a, the incident particle is not used in any calculation, so it can be anything.

The resolution values given in line 5 of the GEO file (see below), are for SIMS different than for the other techniques; the resolution is given in nm. A linear straggling value in nm per nm of sample crossed can also be given (normally, this can be zero) as a second value in line 5 (where the quadratic term is given in the other techniques)

In version 9.3a of NDF, the roughness models are still not used for SIMS, which is bad; in the presence of roughness the user must compensate manually for this, by giving higher depth resolution and/or straggling than real.

For other techniques, the energy calibration is normally given in line 9. Given that scales can be arbitrary, such as sputter time, we allow for a "gain" which is a direct conversion of whatever unit is in the file into nm (it can even be negative). In lateral ubeam PIXE the data normally is μm , so this would then be 0.001; and very often the position of the surface in lateral ubeam PIXE is somewhat ill-defined, so an offset (in nm) is also allowed. By default, the gain

does not change during the fit. To make this a fitable parameter, the user should use NDF.TCN (see section 9 below).

Finally, a sensitivity must be given. That is, a conversion factor from atomic fraction into whatever quantity is measured. A subtle point: the charge-solid angle product is also used to multiply the atomic fraction; so the calculated signal will be:

$$\text{calculated_signal} = \text{charge} \times \text{solid_angle} \times \text{sensitivity} \times \text{atomic_fraction}$$

This is because the sensitivity is some sort of cross section, and it can be the same for different spectra with different collected charge (e.g. in lateral ubeam PIXE)

This sensitivity is given via a file in the same way that non-Rutherford cross section files are given (see section 11.1 below). The format of the file is ASCII two columns, first column is atomic fraction (from 0 to 1), second column is sensitivity. For lateral microbeam PIXE the sensitivity should be the same for all atomic fractions! For SIMS, it can be (and almost always is) an extremely non-linear function of the concentration.

It is this extremely non-linear behaviour, which in the case of real SIMS is even worse since it changes from layer to layer, that makes SIMS a semi-quantitative technique at best. NDF does not even try to give different sensitivities according to the matrix composition, so for real SIMS NDF will only produce reasonable results for impurities in a fairly constant matrix.

So, for data format 33, NDF will take logarithms, and will fit logarithms; this means that any fit that vaguely resembles the data will be considered "good". This is more appropriate for SIMS proper.

For data format 34, the data will be taken as it is, so it is considered to be fully quantitative. This is more appropriate e.g. for lateral microbeam PIXE.

5.4.7 NDP: Neutron Depth Profiling

This is very similar to non-resonant NRA, except that the beam is made of neutrons, that for this practical purpose are not stopped in matter. The input is exactly the same as for NRA, but as beam just write down "nn", 'Nn' or "NN". Remember that for each reaction that you wish to analyse you will need one geometry file .GEO. E.g. for ${}^6\text{Li}(n,\alpha){}^3\text{H}$, where both the alpha and the tritium can be detected, you will need two geometry files, with "6Li nn 4He 3H 4782 4He" and "6Li nn 4He 3H 4782 3H".

The cross section must be input in the line where for other techniques the energy of the beam is given, that is, directly under the reaction definition. This must be in milibarn! Note that the cross sections for NDP are normally on the barn range, so e.g. for ${}^6\text{Li}(n,\alpha){}^3\text{H}$, which has a cross section of 940 b, you must input 940000. NDF will automatically consider the neutron energy to be 25 meV. **WARNING:** Any cross section file given explicitly (see section 11.1 below) will override the cross section defined here.

WARNING: The "charge" input in the SPC file is no longer a charge, it becomes a μC , i.e. proportional to the integrated neutron flux with proportionality constant given by the electron charge expressed in μC . Suppose the neutron flux was $3 \times 10^8 / \text{cm}^2$ and the sample area was 2 cm^2 , and the electron charge is $1.6 \times 10^{-13} \mu\text{C}$. Then the "particle-charge" given in the SPC file will have to be $3 \times 10^8 / \text{cm}^2 \times 2 \times 1.6 \times 10^{-13} = 9.6 \times 10^{-5}$.

In NDP there is a background with approximately $a \times 10^{k \times \text{channel}}$ shape. NDF will guess a first approximation to this background (via the a and k parameters). If you want to let them change during the fit (which you do, since the first guess is not very good), you have to use NDF.TCN (see section).

WARNING: N WARNING: NDF calculates a first guess for this background, which is normally pretty bad, but from version 9.4d (rev.8Nov2011) this can be input via the NDPBACK parameter line in the GEO file (see section 6.15). This means that simulations, as opposed to fits, normally have a bad background unless good fitted values of a & k are supplied. Note that a and k can be fitted by specifying the flags (flagndp(1) and flagndp(2)) in TCN (see section 9.2).

5.5 Beam energy: line 4

In keV.

If you want the beam energy to be a fit parameter, you introduce in the same line a second number, which is how much it change either way. For instance

2000 10

Means the beam energy will change between 1990 and 2010 keV.

Be careful not to give such a large variation that it could go negative!

This is ignored for NRP.

5.6 Resolution: line 5

5.6.1 RBS, ERDA, NRA

FWHM, not sigma. In keV.

NDF also supports FWHM that changes linearly with detected energy. To make $FWHM = 20 + 0.01 E_{\text{detected}}$, line 5 should be:

20 0.01

If you want the FWHM to be a fit parameter, you introduce in the same line a third (and fourth) number, which is how much it change either way. For instance

20 0.3 0

Means the FWHM changes between 17 and 23 keV. Note that the second number is always the linear factor, so for a constant detector FWHM, the second number must be zero.

20 0.01 3 0.001

Means there is a linear factor and it is also a fit parameter.

Be careful not to give such large variations that the FWHM could go negative!

5.6.2 PIXE

Not used, but must be in the file.

5.6.3 NRP (PIGE, RNRA)

This is the FWHM in keV of the combined beam energy spread and Doppler effect (Gaussian shaped). This is not a fitable parameter in this case.

If two values are given, the second value is the resonance width. This will be used in determining the integration step in the calculations, making it as fine as needed. If you have a very fine resonance and don't input its width here, NDF may produce inaccurate results in the very surface.

5.6.4 SIMS

This is the depth resolution FWHM in nm of the technique. It must include any effect of roughness.

If two values are given, the second value is a straggling value in nm per nm.

5.7 Geometry type: line 6

IBM, Cornell, or general. Ignored for SIMS.

5.8 Geometry angles: line 7

Ignored for SIMS.

5.8.1 Definition of angles

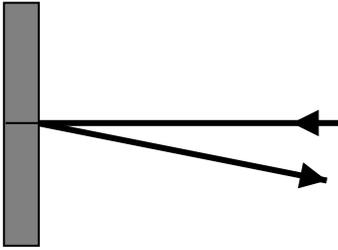
For IBM or Cornell geometry, two values are needed. Four if they are fit parameters (the last two are the variation range). For general geometry, three values are needed, six if they are fit parameters. They are:

angle of incidence, angle of scattering (and angle of exit in general geometry)

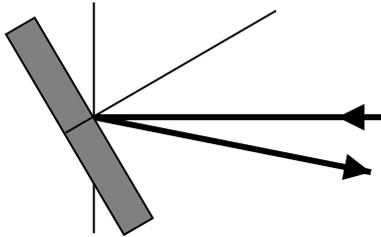
The angle of incidence is the angle between the incident beam and the normal to the surface, while the scattering angle is the angle between the direction of incidence and the direction of detection.

The sign definition of the angle of incidence in IBM geometry is: The sign definition of the angle of incidence in IBM geometry is: negative when the sample surface is turned away from the detector, i.e. sample normal towards the detector, positive when the sample surface is turned towards the detector, i.e. sample normal away from the detector. For forward scattering (or ERDA), this means that the angle of incidence and the scattering angle must have opposite signs. For instance, for (in the commonly used ERDA definition) 15° angle in (with the sample surface) and 20° out (with the sample surface), the angles must be -75° 35° (i.e. 35° scattering angle, and 75° tilt angle; +75° -35° would also work), in the IBM geometry.

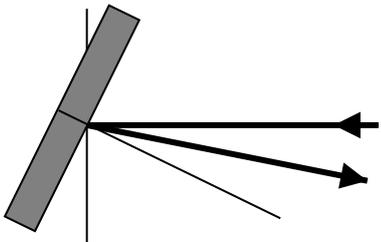
angle of incidence = 0°



angle of incidence $> 0^\circ$



angle of incidence $< 0^\circ$



In complex ERDA spectra, it is possible to have more than one atomic species hitting the detecting system. For instance, in He-ERDA, both ^1H and ^2H (d) can be detected, and their spectra might be superimposed. Also, the forward scattered incident beam ions can also be detected. NDF can handle all these situations, the way to do it is described in section 3.2 above. One geometry file per detected ion must be prepared.

5.8.2 General geometry

The angle of incidence and of scattering are defined as for IBM and Cornell geometry. The angle of exit is between the detection direction and the normal to the sample.

5.8.3 Transmission geometry

Implemented for all techniques, safe for PIXE. A transmission geometry introduced in the GEO file will be automatically recognised as such. The minimum scattering angle accepted is 2° , but the Rutherford cross section is only verified experimentally down to 15° . On-axis STIM is not implemented.

5.9 Solid angle: line 8

Given in msr.

5.10 Energy calibration: line 9

5.10.1 RBS, ERDA, NRA

Linear and quadratic energy calibrations are allowed. For linear energy calibration, input gain (in keV/channel) and offset (in keV), e.g.

2.5 50

If the gain is 2.5 keV per channel and the energy at channel 1 is 50 keV. For quadratic energy calibration, input quadratic term, gain, and offset.

5.10.2 NRP

This is ignored in NRP.

5.10.3 PIXE

This is an energy calibration file for the PIXE detector, as explained in detail in section 5.4.5 above. If the filename is e.g. pixecalib.txt, then this line should be
pixecalib.txt

Facultatively, the user can also use this line to make NDF consider a given error for the PIXE data file attached to this geometry file. The default error considered in NDF for PIXE is 10%, but sometimes it can be less, or more. If the user knows how much it is, it can be input here in fractional units (so 1 is 100%, 0.01 is 1%) . The line becomes e.g. (for 5% error)

pixecalib.txt 0.05

5.10.4 SIMS

Only linear calibrations is allowed. This transforms whatever quantity is given in the x scale into depth in nm.

5.11 Detector efficiency

Except for PIXE (and for particles stopped in the dead layer of the detector if it is specified, see section 5.13), NDF considers a 100% detector efficiency. For PIXE, the detector efficiency is an integrant part of the PIXE calibration.

For the other techniques (NRP is different, see below), it is possible to specify any given energy-dependent efficiency. The yield will be multiplied by this efficiency. The following must be written:

effi effi_file

Where effi-file is a two column file, in format (energy, efficiency). The energy column corresponds to detected energy, i.e. after any foils and after the dead layer. The efficiency column is 1 for 100%.

For NRP, the detected energy is not a parameter. In this case, a single value of the detector efficiency is given:
effi value

Where value is 1 for 100% efficiency.

Ignored for SIMS.

5.12 Foil or delta-E detector

5.12.1 RBS, ERDA, NRA, NDP

You tell the computer that you used a stopping foil (or a delta-E detector) simply by typing it at the end of the appropriate geometry file. For instance, if you have a 5×10^{19} at/cm² thick Al foil, you type

Foil 50000
Al 1

And if you have a 3×10^{19} at/cm² thick Mylar foil, you type

Foil 30000
C 10 H 8 O 4

If on top of that you also have a 3×10^{19} at/cm² thick Si delta-E detector, you must find the total thickness and average atomic composition, which in that case would be

Foil 60000
C 10 h 8 o 4 si 22

An alternative is to use for the Si the “detector dead layer” option described in section 5.13 below.

5.12.2 Foil thickness inhomogeneity in RBS, ERDA, NRA, NDP

Any inhomogeneity in the thickness of the foil leads to extra straggling. This can be calculated by NDF, by inputting the standard deviation of the foil thickness, for instance:

Foil 30000 5000
C 10 H 8 O 4

If you have a 3×10^{19} at/cm² thick Mylar foil with a 5×10^{18} at/cm² thickness inhomogeneity (standard deviation of the thickness).

WARNING: Note that this is very different than what can be input in the input file of DEPTH, where it is FWHM and in micrometer. See section 12.1 below.

WARNING: If straggling is to be calculated with DEPTH, the value for the foil inhomogeneity that is given in the GEO file takes precedence. Whatever is in the DEPTH input file (see section 12.1 below) will be ignored, unless no foil inhomogeneity is given in the GEO file.

5.12.3 PIXE

For PIXE this works differently, since it is not NDF that makes the calculations, but the LibCPIXE library (see section 5.4.5 above), which supports foils with multiple layers. The syntax is then e.g.

```
foil 2
30000 C 10 H 8 O 4
30000 Si 1
```

the layers are given in order of proximity to the sample: so, in this example the beam leaves the sample and crosses a Mylar foil, then crosses a Si foil, then reaches the detector.

“Funny filters” are often used in PIXE. These are normally filters where a given % of the area is holes, or even a completely different material. NDF supports this from 8.2k on. The user must prepare two different geometry files, absolutely equal except for the foil composition. Note that exactly the same elements must be given in both geometry files - so if in one of them there is no foil, the user should nevertheless introduce a foil with the same elements but with thickness e.g. $1.e-5$. Then, the user must use the (1) syntax (see section 3.2 above) for “multiple detected ions”, for instance:

```
pixe_data.file 60 g1 str
pixe_data.file 40 g2 (1)
```

in this case, g1 and g2 are the two GEO files. For instance, g1 includes foil., and g2 has a \approx zero thickness (i.e., it corresponds to the holes). The area of the holes is 40%. The total charge was 100 μ C. The (1) syntax means that NDF will add the results of the second calculation to the first one. The ratio 60:40 is kept constant during the fit.

5.13 Dead layer and non-ionising energy loss in the detector

You may include the dead layer of the detector in the simulations, and also take into account that the nuclear collisions in the active layer of a Si detector do not contribute to the formation of electron-hole pairs. See e.g. Lennard et al.’s work [5]. This will mean you also have to do the energy calibration taking these effects into account. See [6]. The syntax is (include these lines at the end of the GEO file):

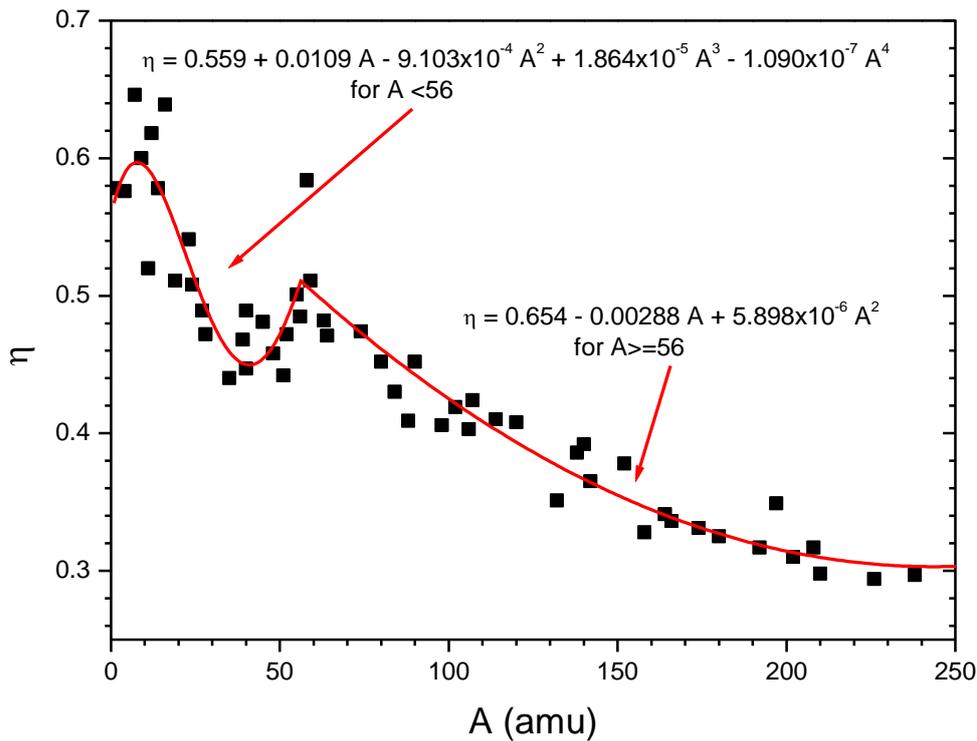
```
Dead 250
Si 1
```

for a deadlayer 250×10^{15} at./cm² thick, composed of Si. DEAD and Dead also work

NDF takes into account the non-ionising energy losses, but assumes a Si detector. This is calculated for He and H isotopes according to Lennard's algorithm [5], but accurate electronic gain calibration may not be possible with the same value for different particles striking the detector.

However, some energy, even being initially transferred to recoils in a non-ionising process, is finally converted into electron-hole pairs due to ionisations produced by the recoils themselves. This is Z-dependent but non-energy dependent. The scaling constant η may be obtained, for each detected ion, by comparison with Monte Carlo calculations (e.g. with SRIM2003) at a single reference energy and scaling appropriately for that energy.

Carlos Pascual-Izarra and Nuno Barradas calculated η for different beams. The results, used as default in NDF, are:



The scaling constant η can be changed in NDF by introducing it explicitly:

nil η

5.14 Thickness of the depleted (active) layer of the detector

For ions with such a high energy that they are not completely stopped in the active (depleted) layer of the detector, only a part of their energy is actually collected. The rest is lost. NDF can calculate this, but the user must give as input the thickness of the active layer, of a detector assumed Si:

active active_thickness

(Active and ACTIVE will also work)

active_thickness is in $1e15$ Si/cm² units.

The user can also specify a distribution of the thickness of the active layer. This may happen for instance for low bias, leading to an inhomogeneous field inside the detector. There are two ways of doing this. The first one is to specify a minimum and maximum thickness, and the number of points to be calculated. For instance

active 5e6 6e6 3

Means that calculations will be made for active layer thickness values of 5e6, 5.5e6, and 6e6.

The second way is to specify an actual distribution that does not need to be homogeneous. This is made via a file, as

active filename

The file starts with an integer which is the number of entries, and then two columns where the first is one active layer thickness and the second the respective weight. For instance, the previous case would be given as

```
3
5e6 1
5.5e6 1
6e6 1
```

To specify a non homogeneous distribution, one can have for instance

```
5
5.0e6 .3
5.4e6 .3
5.5e6 .3
5.6e6 1
5.7e6 3
```

5.15 Foil between the beam and the sample

This may be used to tune the beam energy. NDF supports one single layer in such foil. The syntax is:

```
beamfoil 30000
C 10 H 8 O 4
```

5.16 READGEO override - changing parameters in many GEO files

WARNING: This option was introduced in v9.5b. GEO files that include it will not be correctly read by previous versions of NDF.

Suppose you have a problem with many spectra with similar experimental conditions, for instance the same geometry, and only parameters such as the beam energy (and thus the ROI) need to change. This will led to many GEO files, all equal except for the parameters that change. If you want to change one of the parameters that is constant (e.g. the angle of incidence), you need to go to all the GEO files one by one and change them all. This can be cumbersome for large scale problems.

The REAGDEO command is one way of changing all the GEO files at the same time.

You need to put one extra line at the very beginning of each GEO file affected:

```
readgeo filename
```

The rest of the GEO file remains as it usually is (and the mandatory items described in 5 above need to be present as usual).

However, the file called filename can have experimental conditions, and these will override what is given in the GEO file.

Not all experimental parameters can be overridden with READGEO. In version v9.5b these ones can:

```
ROI
Beam energy
FWHM
Angles defining the geometry
Solid angle
Energy calibration
MS/channelling polynomial (see section 11.7 below)
```

The syntax is simple: first a codeword, then exactly the same as in the GEO file. No particular order for the codewords is required. One example of a READGEO file (with name filename in the example above) could be

```
energy 2020
angles 0 165
fwhm 25
solid 15
calib 3.05 142.
mspoly Si
250 0.20000E+01 -.90000E-03 0.45000E-05 -.50000E-08
mspoly Ti
600 2 2
```

Note that the MS polynomial correction can be on an element that does not have such a correction specified in the GEO

file.

6 The GEO file part 2: advanced calculation options

The geometry file can also be used to specify a large range of calculation options. These will apply only to the spectra that correspond to the geo file in question. In this way, the user can for instance use Chu straggling for a near-normal incidence spectrum and a full MDEPTH calculation for a grazing angle spectrum taken from the same sample. This gives the user complete freedom over every detail of the calculations.

This made by codewords, which are include in the geo file, at its end, that is, after line 9. No special order is required. Commands override each other, for instance if the user first specifies Bohr straggling, and then Chu straggling, the last specification prevails.

Options up to 6.11 are bona-fide calculations, several of them based on advanced physics and algorithms. From 6.12 on they are increasingly not so quite bona-fide and our advice is to use them with much care.

6.1 Stopping power

By default the stopping power is ZBL1990. ZBL95 and SRIM2003 (with any later version also supported) are also supported, but the user must download/obtain the respective executables from www.srim.org. MSTAR is integrated into NDF. See chapter 10.

The user can also scale the calculated stopping for a given element by a constant factor.

Given elemental stopping powers can be loaded in by the user, overriding the tabulated ones.

Molecular stopping powers can be input. In this case, the Bragg rule is not going to be used for that molecule.

Fitting of stopping powers is also possible. In this case, NDF will try to find the stopping curve that best fits the data. This is dangerous and can lead to huge errors in data analysis. Bayesian inference on the stopping power curve can also be done. See chapter 15.

6.2 Beam charge state

The charge state of the incident beam as a function of beam energy input by the user. This will only be used by NDF in the calculation of the stopping powers, if these are ZBL1990, ZBL95, or SRIM2003. See section 10.9.

6.3 Type of straggling

By default straggling is Chu with Tschälär effect. Tschälär effect cannot be turned off, but a range of options for straggling is available, including using in run-time the state of the art code DEPTH. See chapter 12. To turn straggling off, type
Bohr 0.
(chu 0. also works of course)

6.4 Tschälär effect

By default NDF incorporates the Tschälär effect of propagation of straggling through thick layers; the effect is always calculated, even for thin layers. The user can turn it off (for test purposes only, the effect is real and should be left there). See section 12.6 below.

6.5 Shape of the straggling

NDF currently implements two types of functions to describe straggling: Gaussian and gamma. See section 12.5.

6.6 Non-Rutherford cross sections

By default RBS and ERDA cross sections are Rutherford with Andersen screening, except for 4He ERDA of 1H and 2H, which are given in sections 11.11 and 11.12. For NRA the only cross section included is for $d(3\text{He},p)4\text{He}$ and $d(3\text{He},4\text{He})p$ [1]. The user can input any cross section (and chose the units to be ratio to Rutherford or mbarn, and in the lab or centre of mass system). See section 11.1.

Be sure to read and understand sub-section 11.1.9.

The user can also use L'Ecuyer screening, or turn off screening altogether, see section 11.3.

Fitting of cross sections is also possible. In this case, NDF will try to find the cross section that best fits the data. This is dangerous and can lead to huge errors in data analysis. Bayesian inference on the cross section curve can also be done.

See chapter 16.

6.7 Accurate calculation of resonances

NDF simulates resonances quite accurately as long as the angle of incidence is not grazing (no code can currently simulate well buried resonances at grazing angles, not even using Monte Carlo). The algorithm is used by default whenever a non-Rutherford cross section is input. The user can turn this off (for instance if there are no resonances), or turn it on for Rutherford cross sections. See section 11.2.

6.8 Electron screening

By default RBS and ERDA cross sections are Rutherford with Andersen screening. The user can also use L'Ecuyer screening, `m` or turn off screening altogether, and also use different potential screening functions see section 11.3. If a non-Rutherford cross section is input by the user, screening is not calculated as it is assumed that it is already included in the cross section input.

6.9 Double scattering

Can be calculated as user option. It is slow. See section 11.4.

6.10 Pulse pile-up

Can be calculated as user option, using different algorithms. See section 11.5.

6.11 Low energy yield correction

The yield at low energies is notoriously hard to calculate. NDF includes an algorithm to improve the calculation. See section 11.6.

6.12 Ad-hoc plural scattering/channelling corrections

Different sorts of ad-hoc corrections to the calculated yield can be made, to obtain better fits. This is dangerous and can lead to large errors in the data analysis. It can be useful if the signal affected is not important (e.g. substrate) but is superimposed to the signal that needs to be analysed. See sections 11.7 and 11.8.

6.13 Generate apparent cross sections that lead to a perfect fit to your data

That's exactly what it sounds. In fact it can be meaningfully used in at least one occasion: from a measured spectrum with a non-Rutherford cross section, simulate it with a Rutherford cross section while using this option. The result will be the real cross section (convoluted by the energy resolution and straggling, and distorted by any poorly simulated effects). See section 11.9.

6.14 Effects of multiple scattering

The effect of MS on energy spread is calculated by DEPTH as given in chapter 12. Other effects, such as spread of scattering angles and spread of pathlengths are calculated in an approximative way by NDF. This is also quite speculative as these things are not easy to calculate by analytic codes. See section 11.10.1, but do not use this.

6.15 Exponential background for NDP

Neutron Depth Profiling has a background, normally close to exponential. NDF introduces a pure exponential background, of the form $a_1 a_2^{\text{channel}}$. NDF will calculate an initial guess from the data, the initial guess is normally bad, but the fit will take care of it. If you just want to do a simulation, or want to start the fit from a given initial guess, you can give parameters `a1` and `a2` by including at the end of the GEO file the line:

```
ndpback a1 a2
```

Note that this will work depend of the compression level, that is, on the actual number of channels in the fit! `a1` and `a2` will have to be different if you do a fit on the original number of channels, or if you compress to fewer channels.

6.16 Extremely thin layers

When the energy loss in a given layer is lower than around 10^{-8} of the beam energy in the layer, the calculations become tricky. NDF tries to check for this before starting the calculation: in simulation or local search, if the PRF file has a layer thinner than 0.1×10^{15} at./cm², or if the minimum layer thickness specified in the batch file SPC is lower than that value, NDF will change the calculation mode (to automatic division of layers, i.e. the AUTOL option given in section 11.1.9 below, which will accidentally solve this thin layer problem as well). The user can force ultra-thin layer accurate calculation by imposing the option AUTOL in the GEO file. If, on the other hand, the user explicitly turned this option off with AUTOL OFF, then NDF will not attempt the accurate calculation of ultra-thin layers.

7 Running NDF

Simply type “NDF” in the appropriate directory where your data, structure, geometry and batch files reside. NDF will prompt you for the calculation and run options you want to use.

7.1 Simulate a theoretical spectrum from a given depth profile

NDF can simulate spectra based on a depth profile that you define. It will use the experimental conditions contained in the geometry files, as well as the elements specified in the structure file. It will disregard any constraints to depth and concentration range etc. present in the structure file, that is, it will accept the profile you define without corrections.

Both pure simulations (not requiring data) and simulations to be compared with real spectra are possible. There are no restrictions on the number of spectra per sample (i.e. per STR file).

To do it, you need to create a file called NDF.PRF, with the following structure:

First line: number of layers

second line: thickness of layer 1; conc. of element 1 in layer 1,... conc. of elem n in layer 1

third line: thickness of layer 2; conc. of element 1 in layer 2,... conc. of elem n in layer 2

The thickness in $1e15$ at/cm² and the concentrations are atomic (and don't need to be normalised).

As many layers as defined in the first line will be read. Extra ones will be ignored. Fewer layers will make the program crash. The number of element entries must equal (or be larger than) the number of elements n in the structure file, or the program will crash. Extra element entries will be ignored. The order of the elements is the same as in the structure file. For instance, if in the structure file you defined 3 elements: Si, Fe, O, then in NDF.PRF you can have e.g.

4
100 1 0 2
200 1 2 5
1000 1 1 0
30000 1 0 0

Meaning that the first layer is 100×10^{15} at/cm² thick with composition SiO₂, the second layer is a mixed Si-Fe oxide, the third is a SiFe film, and the third is a Si substrate. Notice that the amounts do not have to be normalised. The following file would lead to exactly the same result, since the extra fifth layer and the extra element concentration entries would be ignored:

4
100 1 0 2 1
200 1 2 5 0.1
1000 1 1 0 0
30000 1 0 0 99
111 0 0 1 1

Then you need to run NDF with speed 0 or p (see section 7.3.1 below).

With speed 0, pileup will be calculated, corresponding to the spectrum that is given in the SPC file, and to the corresponding pileup parameters (see chapter 11.5 below for details on the pileup correction).

With speed p, pileup will also be calculated, but corresponding to the theoretical spectrum calculated from NDF.PRF while the spectrum that is given in the SPC file is ignored, and to the corresponding pileup parameters (see chapter 11.5 below for details on the pileup correction). Speed p is thus for pure simulation ignoring the data.

7.2 Run a local optimisation starting from a given depth profile

To do it, you need to create a file called NDF.PRF, equal to what is used for simulations. Speed option is x or 6 (see section 7.3.1 below).

7.3 Main fit options

When the executable is run, the program will either query for option settings or echo current option settings to you (if you set them from the command line). The options are:

7.3.1 Speed

Speeds 0 and p are used to run a simulation, not fit (see section 7.1 above). Speed 6 is used to invoke the custom option for different calculation options (see section 9 below). Speed x is used to do a local search fit only (see section 7.2).

Speeds 1 to 5 are for simulated annealing, 1 is ultra-fast cooling, 3 is normal, 5 is ultra-slow. Speed 3 is advised. Ultra-slow cooling does not normally lead to better results - it was implemented in early versions of NDF for test reasons. If Normal cooling does not work, try ultra-fast cooling. If that does not work either, forget SimAnn and use NDF as a simulator in the normal iterative way.

7.3.2 Compression

{1,2,3,...,9} = {128, 256, 512,..., 32768} channels. This is the maximum number of channels allowed in the fit; spectra with more than this number will be compressed.

0 means no compression will be made, and each spectrum is analysed with the number of channels in the data format (or as defined in the geometry file, see section).

Execution is faster for shorter spectra since both the simulation and the χ^2 calculation involves a fair number of operations per channel. Where high depth resolution is not required, or where the layers of interest are known to be relatively thick, compression can be used.

This will be overridden for a given spectrum if a given number of channels is specified in the corresponding geo file. If fewer than 128 channels are required, the required number should be input directly in the first line of the geo file, after the data type number. Only powers of 2 are allowed. See section 5.2.

7.3.3 Convolute FWHM

System resolution. {0, 1, 2} The detector FWHM (specified below) can be included (1) or not (0) depending on this switch. See section 12 below on how to introduce straggling calculations.

7.3.4 Isotopes

{0, 1} Determines if the isotopic distribution (we only calculate for isotopes with at least 1% abundance) will be used (1) (i.e., if spectra for each isotope of each target element are to be calculated), or if the average mass of the target elements is to be used (0). Using the isotopes makes the calculations slower, but for 512 channel fits of targets that include light/medium elements with a lot of isotopes, it is better do use it.

7.3.5 Smooth

{0, 1} A high quality smoothing routine is available which varies the filtering function depending on whether peaks or edges have been detected in the data. *smooth*=1 if you want to use it, 0 if not. It is adaptive in that flat bits get smoothed a lot, but peaks and edges don't (try it!).

WARNING: χ^2 fitting should not be applied to smoothed data: you are likely to get spurious confidence in the results.

7.3.6 Output point by point profiles

Up to version 9.0b, the following applied: "In previous versions we used SQUEAKIE [7] as a post-processor, this option was related to it. It is no longer used."

From version 9.0c, a 1 makes NDF output the point by point profiles even for verbosity 1 (see section 8.3.1).

7.3.7 Recalculate

Any value in this flag will activate it, that is, the simple presence of a seventh flag is enough.

For instance when running speed 6 – custom cooling where no more flags are needed, NDF batch 6 0 0 0 0 0 x will activate *recalculate*.

What does it do? Imagine you just ran a batch with 4 samples, and sample 3 went wrong. You want to recalculate it after changing the sample file or whatever, but as NDF is serial, in v1.0 or v2.1 you would have had to redo all of them. Now, a file called NDF.ORD will be read if this flag is activated. NDF.ORD has as many lines as samples in NDF.SPC, each line is 0 if the corresponding sample is not to be recalculated, 1 if yes. In the example above, NDF.ORD would then be

```
0
0
1
0
```

Furthermore, suppose that you are running local search only (see sections 7.8.2, 7.1, and 7.6). Then, the same NDF.PRF file will be read for all samples. This is potentially wrong, since the samples may have nothing in common, and the

NDF.PRF file should be different for each one of them. To make this happen, the 7th element in the command line must be "p":

```
NDF batch1 1 3 1 1 0 0 p
```

Finally, if you want to have this option in the command line but you want all samples in the SPC file to be calculated, the 8th element in the command line should be "n":

```
NDF batch1 1 3 1 1 0 0 n
```

7.3.8 Normalise χ^2

By default, the partial χ^2 for each spectrum is normalised to the number of channels in that spectrum (or to the number of lines in a PIXE spectrum). The total χ^2 will be the average of the partial χ^2 . (this is true for local search and simulated annealing; during BI, a non-normalised χ^2 is used to ensure validity of the calculations).

You can make NDF use the non-normalised χ^2 also for local search and simulated annealing. The 9th element in the command line must be "0":

```
NDF batch1 1 3 1 1 0 0 n 0
```

If it is "1", this explicitly tells NDF to normalise.

Note that when the χ^2 is normalised, a good fit should have a χ^2 close to 1.

7.3.9 Form of χ^2 function - standard or robustified

Before v8.1o, the χ^2 function used in NDF was just the normal definition $\chi^2 = \sum_i \frac{(\text{fit}_i - \text{data}_i)^2}{\max(\text{data}_i, 4)}$, where $\max(\text{data}_i, 4)$ is to approximate Poisson statistics.

In order to try to implement robustification of the fitting process, from v8.1o on, NDF has two options for the χ^2 function. The definition above is one of them. The other is made to turn NDF almost impervious against problems that arise when the fit is bad in a given region of the spectrum, not because the sample structure is wrong, but because of physical effects not implemented in NDF, not calculated accurately enough, or not calculated at all due to user option (examples are straggling, plural and multiple scattering, low energy tails in peaks, channelling, etc). In this case, in a

channel i where the fit is good, the new function will be just $\chi_i^2 = \frac{(\text{fit}_i - \text{data}_i)^2}{\max(\text{data}_i, 4)}$, which is the same as before.

However, in a channel where the fit is bad and the corresponding χ_i^2 is very high, the new function tends to $\log(\chi_i^2)$, and therefore the influence in the χ^2 will be small.

This is a user option, and it should be used with care. In particular, this should work best when the physical models implemented are not good enough to make a proper fit to the data.

The cut-off between "good" and "bad" is taken as default to be 10 times the error in a given channel. That is, if the difference between fit and data is up to 10 standard deviations, the normal χ_i^2 is taken. If it is larger than that, then a smooth transition to $\log(\chi_i^2)$ is made.

To make NDF use the old χ^2 in SimAnn, the 10th element in the command line must be "0":

```
NDF batch 1 3 1 1 0 0 n 1 0
```

To make NDF use the new χ^2 in local, the 10th element in the command line must be a number larger or equal to "1":

```
NDF batch 1 3 1 1 0 0 n 1 5.
```

To change the cut-off between "good" and "bad", just input the required value as the 10th element in the command line, for instance for a cut-off of 3 standard deviations (not that any value smaller than 1 will be corrected to 1):

```
NDF batch 1 3 1 1 0 0 n 1 3.
```

7.4 Examples of possible Commands

NDF You will be prompted for all options.

NDF sic 0 A simulation will be done. You will be prompted for the other options.

NDF sic 0 2 1 0 0 1 Simulation, 256 channels, FWHM will be convoluted with fit, no isotopes, no smoothing of the data, SQUEAKIE output will be normalised to 100 at.%.
 NDF batch2 6 A custom fit will be done, with all the options defined in the file NDF.TCN (see Appendix 5).
 NDF sin 0 3 1 1 0 0 0 Simulation, 512 channels, no smoothing, SQUEAKIE output not normalised, NDF.ORD (see 6.2.7) will be read. NDF.PRF will be used
 NDF sic 0 3 1 1 0 0 x Same as above.
 NDF sic 6 0 0 0 0 0 0 Custom fit. NDF.ORD (see 6.2.7) will be read.
 NDF sin 0 3 1 1 0 0 p Simulation, 512 channels, no smoothing, SQUEAKIE output not normalised, NDF.ORD (see 6.2.7) will be read. NDF01.PRF etc will be used
 NDF fesi x 3 1 1 0 0 Local search only based on NDF.PRF as initial guess (see section 7.1 above).
 NDF sin 3 3 1 1 0 0 0 Normal SimAnn cooling, 512 channels, no smoothing, SQUEAKIE output not normalised, NDF.ORD (see 6.2.7) will be read.

In the examples above, sic, sin, batch2, fesi were all batch files with extension .SPC.

When the program runs it will tell you the cooling schedule it is using, based on its analysis of the complexity of the spectrum. It will then echo, for each Markov chain, the value of the control parameter, the current χ^2 value, and the best χ^2 value found so far. When it switches to the local search it will echo the χ^2 for each minimisation cycle completed.

7.5 Controlling which parameters change during the fit

The user may wish to fit the charge in only some spectra, and keep it constant in the others; or let the depth profile have always the same number of layers; or be completely constant; etc. Unfortunately, this has evolved over the years, which means that the control of these things is spread in different locations as detailed in the next sub-sections.

7.5.1 Beam energy, angle of scattering, incidence and exit, FWHM

By default, these parameters do not change during a fit.

However, they can be fit parameters. This is specified in the geo files, by introducing a second value in the line where the parameter is defined. For instance, to make the energy change by plus or minus 10 keV relative to the nominal value, the line where the beam energy is defined must be e.g.

```
1600 10
```

If no second number is present, or if it is a 0, then the parameter stays constant. For the angle of incidence and scattering in Cornell or IBM geometry, the variation ranges are given after the angle values:

```
20 160 1 0
```

In this case the angle of incidence can change between 19° and 21°, and the scattering angle remains constant. For general geometry where the exit angle is also given, it could be e.g.

```
20 160 75 1 2 3
```

The angle of incidence changes between 19° and 21°, the scattering angle between 158° and 162°, and the exit angle between 72° and 78°.

WARNING: In the current version, NDF does NOT check whether changing angles lead to impossible situations; for instance, if in ERDA you specify for the angles

```
-15 30 10 10
```

This can lead to a situation where the incidence is 25° and the scattering angle 20° (that is, transmission). However, NDF will not recognise this or even check, it will simply explode or produce silly results.

7.5.2 Energy calibration and charge

By default, these parameters change during a fit.

In the tcn file (see section 9 below) the user can control whether the energy calibration (independently for the gain and offset and quadratic term) and the charge change or not via the flag4par flags. Refer to that section.

What is in the tcn file is applied by default to all geometries. If the user wants to make things different in a given geometry, these codewords can be input at the end of the geo file:

```
flagvar_charge on
flagvar_quad on
flagvar_conv on
flagvar_offset on
```

to make NDF fit the charge, quadratic term, gain and offset in the energy calibration

```
flagvar_charge off
flagvar_quad off
flagvar_conv off
flagvar_offset off
```

to make NDF not fit the parameters.

The flagvar_ codewords in the geo files override, for the geometry in question, what is specified by the defaults (all parameters change) or in the tcn file. The other geometries are not changed.

7.5.3 Multiple scattering ad-hoc polynomial

By default, these parameters change during a fit.

Whether this changes or not can be defined only in the tcn file (see section 9 below), and is the same for all geometries.

7.5.4 Number of layers, layer composition, depth profile

By default, these parameters change during a fit.

Whether these change or not can be defined in the tcn file (see section 9 below).

If Bayesian inference is done, the user can also specify that the depth profile will not be fitted in the MCMC.BI file (see section 14 below).

7.5.5 Roughness parameters

Whether these change or not is defined in the str and prf files (see section 13.4.1 below).

7.5.6 Volume fraction of inclusions

Whether these change or not is defined in the str and prf files (see section 13.5 below).

7.5.7 Fine control of the fitting process in a local search

In a local search, the initial guess is given by the PRF file, which contains the number of layers, then for each layer its thickness and the concentration of each molecule. In a local search fit, the user may want that some of these values does not change. For instance, that the thickness of a given layer remains constant, or that the concentration of a given molecule remains constant. This is a much finer control than what is allowed by section 7.5.4 above.

It can be done via one extra file, called NDFPRF.ORD. This file must have the exact same structure as the existing PRF file that contains the initial guess. The first line must be the number of layers, and it must be the same number as given in the PRF file. Then, for each layer it must have the same number of entries as the PRF file, i.e. one for the thickness and one for each molecule present. But, instead of numbers, it has a 0 if the parameter is to remain constant during the fit, and a 1 if the parameter is allowed to change.

If inclusions/voids are present, then one extra column is needed: 0/1 if the volume fraction is constant/changes.

If layer roughness (i.e. type 1 or type 4, see section 13.4 below) is present, then one extra column is needed: 0/1 if the roughness value is constant/changes.

Note that in NDF versions before 9.5a, whether the volume fraction of inclusions or the amount of layer roughness changes or not, was controlled by 0/1 flags in the PRF file, as described in sections 13.4.1 and 13.5.3. For back compatibility reasons, the PRF file still requires these 0/1 flags to be present, but whatever is in NDFPRF.ORD takes precedence and overrides it. If there is no NDFPRF.ORD file, or if it is not in the required format, then whatever is in the PRF file is used.

WARNING: If there is no existing NDFPRF.ORD and a simulation or a local search are done, NDF or WiNDF will create one. Elements with zero concentration in a layer do not change in that layer.

WARNING: There may be unexpected interactions between parameters, and also with the depth and concentration

constraints imposed in the STR file, as well as with inclusion volume density when inclusions are present. This may lead to deviations from expected behaviour.

7.6 Repeating only some of the fits in the batch file

Imagine you just ran a batch with 4 samples, and sample 3 went wrong. You want to recalculate it after changing the sample file or whatever. Section 7.3.7 above explains how to do it. Read then come back here.

To make NDF read NDF.ORD, you have to run it using command line options (see section 7.4 above), such as

```
NDF batch1 1 3 1 1 0 0 x    (all commands used)
NDF batch1 6 0 0 0 0 0 x    (all zeros are unused and could be anything else as long as they're there)
NDF batch1 0 3 1 1 0 0 x    (last two zeros are unused and could be anything else as long as they're there)
```

All that is needed is the presence of a seventh element in the command line list. So if you want to run e.g. in custom cooling mode, that requires only a "NDF batch 6", you have to put five bogus list elements and then a seventh to make NDF.ORD be read.

Suppose that you are running a simulation or local search only (see sections 7.1, 7.2, and 7.8.2). Then, the same NDF.PRF file will be read for all samples. This is potentially wrong, since the samples may have nothing in common, and the NDF.PRF file should be different for each one of them. To make this happen, the seventh element in the command line must be "p":

```
NDF batch1 1 3 1 1 0 0 p
```

Then, for sample number *s* (and batch *b*.SPC), the profile file *bs*.PRF will be read (see section 8.2.6 below).

Alternatively, NDF may be run with those options given in the command line, with command line parameter syntax as:

```
NDF [batch filename] [speed] [compression] [FWHM] [isotopes] [smooth] [not used] [recalculate]
```

NDF will then echo current option settings to you (if you set them from the command line). Missing or invalid parameters will initiate a question and answer session (invalid parameters may cause the code to give a warning message and stop). For a full description of these parameters see below. Notice that in the current version of NDF the random number sequence is always initialised at the same position (in order to make debugging easier), so if you run NDF twice with the same parameters and options, you'll get the same results. To force a different random number sequence simply change any input parameter very slightly, or change the random number sequence directly as explained in section 7.9.5 below.

7.7 Repeat simulation

A single simulation often seems slow in NDF. This is because at the beginning, before any actual calculation of yield is made, NDF pre-calculates and puts into tables all the quantities that it can, such as stopping powers, scattering cross sections, energy loss in foils (depending on beam energy), etc. This takes time, but when fitting it means that many things do not need to be recalculated.

When doing iterative manual simulation, however, it is nice to simply change the PRF or GEO file, and have NDF automatically recalculate the simulation. This is done by creating a file called TALK.W, with the single letter *r* in the first position of the first line. NDF makes the first simulation then waits until either PRF or GEO changes; when this happens, it makes a new simulation. NDF exits this repeat mode either if TALK.W is changed to have the letter *t*, or if too much time has passed (somewhat over 5 minutes).

This works only for simulation of a single sample (but with any number of spectra for that sample).

The PRF file can be completely changed, i.e. number of layers, their thickness and composition.

The GEO file changes accepted are the beam energy, energy resolution, incidence, scattering and exit angles, solid angle, and energy calibration. The calculation flags cannot be changed (for instance changing from Bohr to Chu straggling is not allowed). Furthermore, the beam energy can only be changed by a maximum of 50%.

This also works for multiple spectra: if the PRF is change, or any of the GEO files is changed, a new calculation is made.

This does not, as far as I am aware, work for PIXE.

7.8 Automated fitting: SimAnn, BI/MCMC

Not supported in ALP. NDF provides a number of possibilities for fully automated fitting, including Simulated Annealing and Bayesian inference with the Markov chain Monte Carlos algorithm

7.8.1 Run a simulated annealing search

Not supported in ALP.

This is for advanced users only, who fully understand the details of the experimental techniques. The results may need a great deal of interpretation, since no guarantee is given that the final results is the true depth profile.

You don't need an initial guess, the algorithm will find the solution for you. It is advised that you should input any knowledge about depth and concentration limits in the STR file (see section 4.3). Do not make this too rigid - if you think a given element exists only in the depth range 200 - 5000, put as limits for instance 0 - 10000.

Speed option is 1, 2, 3, 4, 5, or 6 (see section 7.3.1 above). Speed 3 (normal cooling) is advised. Ultra-slow cooling does not normally lead to better results - it was implemented in early versions of NDF for test reasons. If Normal cooling does not work, try ultra-fast cooling. If that does not work either, forget SimAnn and use NDF as a simulator in the normal iterative way.

7.8.2 Local SimAnn

Not supported in ALP.

This is for advanced users only, who fully understand the details of the experimental techniques. SimAnn normally requires that the initial state is of high entropy (i.e. molten). So, even if the initial guess is near-perfect, it is destroyed in the initial SimAnn stages. However, it may be desirable to start SimAnn from a given initial guess supplied in NDF.PRF (see section 7.1 above), and adjust the initial temperature such that the information present in the initial guess is not destroyed, or at least used. We developed a routine that adjusts the initial temperature to how good the initial guess fits the data. If the initial guess is lousy, a high initial temperature is chosen. If it is good, then SimAnn starts already at a fairly low temperature, such that only small optimisation will be done. In intermediate situations, NDF will try to find an initial temperature that always reflects how well the initial guess in NDF.PRF matches the data. To use this option you must use NDF.TCN with flagoldd = 3, as described in section 9.2 below.

7.8.3 Running Bayesian inference

Not supported in ALP.

This is for advanced users only, who fully understand the details of the experimental techniques. To do it, you need to create a file called NDF.PRF, equal to what is used for simulations. Speed option should be 6 (so you can control the entire process and the experimental errors associated to experimental parameters) but it can also be x (see section 7.3.1 above). Chapter 14 explains how to do this for depth profiles, chapters 15 and 16 explain how to do it for stopping powers and cross sections.

7.9 Fit tweaking tools

NDF provides a number of tools to tweak the fitting process for the advanced user.

7.9.1 normalise χ^2

By default, the partial χ^2 for each spectrum is normalised to the number of channels in that spectrum (or to the number of lines in a PIXE spectrum). The user can however specify that the χ^2 should be non-normalised also for local search and simulated annealing. See 7.3.8 above.

7.9.2 form of χ^2 function - standard or not

The user has two different forms of the χ^2 available. One is the normally defined χ^2 , the other, default in NDF, attempts to make the χ^2 less sensitive to misfits due to causes other than a bad layer structure (such as straggling, plural and multiple scattering, low energy tails in peaks, channelling, etc). See 7.3.9 above.

7.9.3 Errors for χ^2

NDF can calculate the χ^2 using errors. These can be the statistical Poisson error (which is the correct case, and is default); a fake error calculated from the misfit between data and fit; or an equal error to all channels. This is a user option via the MCMC.BI file, as described in section 14.4.

7.9.4 Include Poisson noise in the simulation

You may want to generate a realistic-looking calculated spectrum, by adding Poisson noise to the simulation. You can do this by specifying a simulation speed with two letters, where the second letter is a p, for instance Op or pp. Op for simulation with pileup calculated from the data, pp for a pure simulation ignoring the data.

7.9.5 Random number sequence

NDF uses a random number routine with a very long period, based on two seeds. The default starts with the seeds 12345 67890. The user can change the seed by creating a file called ISEED. in the directory where NDF is being run, with the two seeds (5 digit integers). If the user wants the random sequence to start in a random position, then the seeds in file ISEED must be zero.

7.10 Re-use fit results

This is for advanced users.

During a fit, experimental parameters given in the GEO files can change (see section 7.5 above). Also, in a STR file that includes molecules with variable stoichiometry, this may change during the fit. Suppose that, at the end of the fit, you wish to simulate (or fit) for the exact same conditions that correspond to the final previous fit. One method is to look at the RES results file, and copy by hand all the parameters that changed to the affected GEO and STR files (and SPC, since the charge also may have changed). This is a time-consuming process that is also prone to error.

NDF provides a facility for this. On each fit, NDF writes SPC, GEO, and STR files that correspond to the last fit made. These will be called (see section 8 below for an explanation of what b, s and f stand for)

b.SPX - equal to b.SPC but includes the last fitted charges and last fitted STR and GEO files

bs.STR - STR files that correspond to the last fit (unless there are molecules with variable composition, they will be equivalent to the original STR files)

bsf.GEO - GEO files that correspond to the last fit (unless there are parameters that change, they will be equivalent to the original GEO files)

The SPX file has, for each sample, whatever was done in the last fit or simulation for that sample. Suppose a SPC file with 5 samples. You first simulate all samples from NDF.PRF. No fit was done, so SPX is equal to SPC, and all created bsf.GEO and bs.STR files are equal to the original GEO and STR files.

Then you make a fit for sample 3. SPX will have, for that sample, the fitted charge, and will also include b03f.GEO and b03.STR. Then you fit sample 5. SPX will keep the changed files for sample 03, and will also include the changed files for sample 05. Then you make another simulation for all samples starting from SPC. SPX will be again equal to SPC, because the last simulation for samples 03 and 05 is simply what is in SPC.

To use SPX, the 7th element in the command line must be "f", e.g.:

```
NDF batch1 x 3 1 1 0 0 f
```

In a local search (such as in the example given above), the previously fitted depth profile bs.prf is also used.

8 Outputs

In what follows, *b* is the first three characters of the batch file name *batch.SPC*. *s* is the order number of each sample fitted in one batch. That is, if you have 15 samples to fit specified in your *batch.SPC*, *s*=01,02,...,15. Then, *f* is the order number of each file fitted simultaneously for a sample. That is, if one sample had two spectra fitted simultaneously, *f*=01,02. Then, *e* is the number of an element, given by the order it appears in the structure file. Finally, *l* is the layer number, and *i* the isotope number.

The numbering goes from 01 to 99, then a0-a9, up to z0-z9. That is, the maximum number of samples or geometries for a given sample is 359. We don't think more will ever be needed.

NDF will produce various headers with information about the fit, the sample, and the files used. These headers will be added to the output files described below.

8.1 Verbosity

The verbosity flag, which is defined in the *SPC* batch file (see section 3.6), defines what output are generated. The default, which is *verbosity=0*, makes NDF write the result file, the calculated spectra, the calculated elemental spectra, the best layer structure, and the log file. Verbosity 1 also leads to the point by point profiles. Verbosity 2 also leads to the calculated layer elemental spectra. Verbosity 3 leads to output of extended intermediate calculations.

8.2 Verbosity 0: default output

8.2.1 Result file: *bs.RES*

Comprehensible text is output to the *n* files *bs.RES*: one file for each sample analysed. These files include the experimental set-up and the partial chi-squared for each spectrum fitted simultaneously, the options used in the fit, the calculation parameters, and the layer structure found to give the best fit to the data.

8.2.2 Calculated spectra: *bFsf.DAT*

The data and the fitted calculated spectra are output in *bFsf.DAT*. For instance, for spectrum 3 of sample 8, you will have *bF0803.DAT*. You must use your own graphics package to inspect this file to verify the quality of the fit. The format is ASCII, 3 columns:

channel, data, fit

For NRP, this is different. The format is ASCII, 3 columns:

energy, data, fit

For SIMS, this is different. The format is ASCII, 3 columns:

depth, data, fit

For PIXE, this is different. The format is ASCII, 5 columns:

element, line, data, error, fit

8.2.3 Calculated elemental spectra: *bXsf.DAT*

For each sample and spectrum analysed, NDF generates files containing the calculated partial spectra of each element, and puts them all in this file. The first column is the channel number, the others are the partial spectra for the elements, in the order they appear in the structure file.

The last column has all the extra contributions to the final spectrum, including the pileup spectrum (see section 11.5 below), and any thin film ad-hoc correction (see section 11.8 below). The fit (third column in *bFsf.DAT*) is the sum of all the partial spectra plus the extra contributions in the last column.

8.2.4 Best layer structure: *bPs.DAT*

This outputs the logical elements (elements, molecules, complex molecules) as defined in the *STR* file.

The best layer structure is output as a depth profile in a format that most graphics packages can read. The format is ASCII (the number of columns equals the number of logical elements present plus 4). Please notice that the depth in nm (last column) assumes Bragg's rule for the density of each layer, calculated from the elemental densities.:

depth (10^{15} at/cm²), Element 1 (at.%),..., Element x (at.%), depth (nm), density ($1e22$ at/cm³), depth (um/cm²)

8.2.5 Best layer structure: *bPEs.DAT*

This outputs the real elements.

The best layer structure is output as a depth profile in a format that most graphics packages can read. The format is

ASCII (the number of columns equals the number of elements present plus 2). Please notice that the depth in nm (last column) assumes Bragg's rule for the density of each layer, calculated from the elemental densities.:

depth (10^{15} at/cm²), Element 1 (at.%),..., Element x (at.%), depth (nm), density ($1e22$ at/cm³), depth (um/cm²)

8.2.6 Best layer structure: bs.PRF

This is a file with the final fitted layer structure, but similar to the NDF.PRF file described in 7.1. It can be used to repeat only some of the fits in the batch file taking as initial guess the best fit already obtained (see section 7.6 above).

8.2.7 Log file: bs.LOG

This file gives different information about the fitting process, i.e. the start and finish time, as well as the chisquared at each temperature.

8.2.8 XML output

This is included in verbosity 1, but currently it is not supported by NDF yet.

8.2.9 Calculated double scattering: DSsf.DAT

The calculated double scattering is output in DSUsf.DAT. The format is ASCII, 2 columns:
channel, double scattering

8.2.10 Calculated Pileup: bUsf.DAT

The calculated pileup is output in bUsf.DAT. The format is ASCII. It first has a header with codewords that define which parameters were used in the calculation. Then 5 columns follow:

pileup ; pileup gains ; pileup losses ; data-pileup ; max(data-pileup,0)

8.2.11 Stopping powers and cross sections

Files containing stopping powers and non-Rutherford cross sections may be generated. See chapters 15 and 16.

8.3 Verbosity 1

8.3.1 Point by point profiles: bsfe.DAT, and signal areas

The best layer structure for a sample is used to calculate separate partial spectra, one for each element in the sample, and these partial spectra can be used to disentangle the different elemental signals in the original experimental spectrum. Then an analytical calculation generates depth profiles channel by channel from the original data. This works for RBS, ERDA, and for NRA if possible (see our publications 13 and 17). The output is given in bsfe.DAT (in this case, b is only the first two characters of the batch filename).

WARNING: This output relies heavily on the quality of the fit for elements whose signals were not already fully separated to start with. A bad fit will mean that the results so obtained are also bad. You must satisfy yourself that the fit is good and that you trust this result. Further, it does not rely on all signals being present, but it does rely on the stopping powers calculated during the fit.

WARNING: You must be aware of what this kind of output means in physical terms. Some people like it very much, others think its usefulness is dubious since all effects like system resolution, straggling, isotopic effects, etc., will be convoluted with the profiles obtained.

Output is in 3 cols. Please notice that the depth in nm (last column) assumes Bragg's rule for the density of each layer, calculated from the elemental densities or from the input-defined ones:

depth (10^{15} /cm²); concentration of element e (at.%); depth (nm).

This can be made to output even in verbosity 0 by using the appropriate speed option. See section 7.3.6.

NDF uses the same algorithm to calculate the integrated area of each element. It takes the data, subtracts the signals of the other elements, and integrates over what's left. To prevent integrating in far away regions of the spectrum where misfits are inevitable (e.g. at low energies), NDF only integrates where the element is expected to be (using the fit to determine what this region is).

8.3.2 Peaks and edges: bPEAKs.PE

NDF has a simple routine to recognise peaks and edges, calculate their position and their width (in channels). These files output the results of the routine.

WARNING: The files are also created if the data are smoothed.

WARNING: The routine is tweaked towards smoothing (in regions around peaks and edges, less smoothing is applied). It is not very good at recognising back edges superimposed on large backgrounds.

8.4 *Verbosity 2*

8.4.1 **Calculated layer elemental spectra: bEsfel.DAT**

This is non-DOS format, as the file names will have 12 characters plus extension. Each file has the signal of one given element in one given layer.

For instance, for a batch file called ndf.spc, the file

ndfe03020428.dat

has the signal of element 4 in layer 28, for spectrum 2 of sample 3.

8.5 *Verbosity 3*

8.5.1 **Extended intermediate calculations: bCsfei.DAT**

This is non-DOS format, as the file names will have 12 characters plus extension. Each file includes several quantities calculated by NDF. These are documented in the files. They can be for instance:

- Layer number. These are the layers as defined by the user
- Layer thickness
- Concentration of the element in the layer
- Internal layer number. These correspond to the internal sub-division of layers
- Internal layer thickness
- Amount of the isotope being considered in this layer
- Beam energy before scattering in the internal layer
- Detected energy for scattering in the current isotoped in the current internal layer
- Scattering cross section
- Total yield of the isotope in the internal layer

WARNING: If there is a gamma roughness layer distribution thickness, this output is calculated not for the average structure, but for the last point in the calculation (for the thickest layer value considered). It will NOT reflect the true average structure as given in NDF.PRF. To get it right, you need to run a simulation forcing Gaussian roughness.

9 Fine-controlling the fitting process: NDF.TCN

NDF provides several built-in cooling schedules (which determine how fast and how accurate the fit will be, as seen in section 7.3.1 above). It also has several built-in options and features, for instance, for fine-tuning the charge or the energy calibration during the fit, within certain built-in limits. You can change all this by using the file NDF.TCN.

9.1 What it is for

The NDF.TCN file allows the user to control as much as possible the fitting process. This is opposite to the “Ultra-fast” to “Ultra-slow” options, where all the settings are pre-fixed. NDF.TCN is not needed in the majority of cases; it is intended to be used only by people experienced in SimAnn/NDF, who want to understand and control the behaviour of the program better. “TCN” stands for “technical”. It is also the only way to have pile-up correction (see section 11.5 below) on data types other than “raw XRBS” or “normalised XRBS”.

9.2 Detailed description

We will go through each line of the NDF.TCN file. We first give the (internal NDF source) name of each flag or parameter in the file, and then explain what it does.

flagoldb	Integer. It is 2 for local search. If it is -1, a simulation will be made from NDF.PRF (see section 7.1 above). If it is 3, a local SimAnn search will be done (see section 7.8.2 above).
flag4par(1),chconv	The flag4par flags determine if the energy calibration and the solid angle-charge product are allowed to change during the local search. They are 1 if yes, 0 if not.
flag4par(2),choff	flag4par(1) and flag4par(2) control the gain and offset, flag4par(5) a quadratic term
flag4par(5),chquad	of the energy calibration respectively, flag4par(4) the solid angle-charge product. The ch* values determine how much each parameter is allowed to change during the fit, for instance, if flag4par(1)=1 and chconv=0.02, the energy calibration gain will change a maximum of 2% during the local search. Note that the flagvar_ flags in the geo files override the flag4par.
flag4par(4),chsolcha	
Flagms,chms	Same as for the previous four lines, but to decide whether plural scattering polynomial correction (see section 11.5) is allowed to change or not and by how much.
pacctrns	Not used in NDF. Use .9
convcool, offcool	Not used in NDF. Use 10 10
safactor, saoffset	Not used in NDF. Use .4 .4
lmarmax	Not used in NDF. Use 100
error1, error2	Real. Error2 is the final error level (% change in x2).
accmax	Not used in NDF. Use .1
factnlay	Real $\in [0,2[$. λ in Eq. (1) of NIMB136-138 (1998) 1157-1162. Values between 1. and 1.5 are strongly recommended for BI.
xadmin	Real. Minimum internal sublayer thickness (in $10^{15}/\text{cm}^2$) in the calculation of the theoretical spectra, where the stopping powers are constant. Large values lead to faster calculation at the expense of precision. In the “Ultra-fast” option, it is 400, and steps can sometimes be seen at low energies (where the stopping powers change faster).
maxthick	Real>0. Initial total thickness at the beginning of the fit, in $10^{15}/\text{cm}^2$.
nlini	Not used in NDF. Use 10.
fcompr	Integer. 0 if no data compression, 1 to 9 if the fit is to work on 128 to 32K channel spectra (compressed only, no expansion is ever made, see section 7.3.2 above). Shorter spectra lead to faster fits, but depth resolution may be lost..
fconv	Integer. 1 if the theoretical spectra are convoluted with the FWHM of the set-up (as given in NDF.PAR), 0 if not. For 512 channel fits, it is recommended to make the convolution (or the edges will not be well fitted).
fisot	Integer. 1 if the isotopic distribution (we only calculate for isotopes with at least 1% abundance) will be used (i.e., if spectra for each isotope of each target element are to be calculated), 0 if the average mass of the target elements is to be used. Using the isotopes makes the calculations slower, but for 512 channel fits of targets that include light/medium elements with a lot of isotopes, it is better do use it.
flgsmoo	Integer. 1 if the data will be smoothed (by an intelligent adaptive routine that normally makes real good smoothing [8]), 0 if not. In principle, the χ^2 statistics can not be used on smoothed data, i.e., if you are fitting, you shouldn't smooth the data.
fsquea	Integer. From version 9.0c, a 1 makes NDF output the point by point profiles even for verbosity 1 (see section 8.3.1). (from this point on, the lines need not be in NDF.TCN. However, if want to have e.g. f4_comp, you must also introduce all lines before that one)
imaxnlays	Integer, using at least 1000 is recommended. Maximum number of internal layers used in the

	calculation.
icross	Integer, using at least 1000 is recommended. Number of cross section values in the internal table used in the calculation.
innstop	Integer, using at least 1000 is recommended. Number of stopping power values used in the calculation.
f4_nlays	Integer, 1 if the number of layers may change during the fit, 0 if not.
f4_thick	Integer, 1 if the thickness of the layers may change during the fit, 0 if not.
f4_comp	Integer, 1 if the composition of the layers may change during the fit, 0 if not.
flagndp(1)	Integer. The flagndp flags determine if the a and k parameters in the $a \times 10^{k \times \text{channel}}$
flagndp(2)	background in NDP change during the fit or not. flagndp(1) is for a, flagndp(2) is for k.

WARNING: f4_nlay, f4_thick, and f4_comp are not completely independent. If the number of layers can change, then the thickness and composition also can, no matter what is in f4_thick and f4_comp. And if the thickness or composition can't change, then the number of layers also not.

10 Stopping power data

NDF can use different sets of stopping power, and the user can define their own stopping powers. As many specifications as possible for ion/target element pair can be used. Please note however that the last specification override the previous ones for the same ion/target element pair.

The user can scale the values (see section 10.5 below), or input completely new ones (see section 10.6).

Stopping in the beam foil, foil dead layer, and active layer of the detector (assumed Si) are all affected and can all be controlled by the user as explained in this section.

WARNING: Any use of a given set of stopping powers implies that the user must credit the original source, be it ZBL, SRIM, MSTAR, or anything else.

10.1 ZBL2000 stopping

By default, NDF uses ZBL2000 stopping power values. However, The input files scoef.95a and scoef.95b must be in the WiNDF directory.

The user may explicitly include the code word in the geo file:

zbl00 (or ZBL00 or Zbl00)

10.2 ZBL1990 stopping

NDF can use ZBL90 stopping power values [9]. The user must then include the code word in the geo file:

zbl90 (or ZBL90 or Zbl90)

10.3 MSTAR stopping

NDF supports MSTAR stopping power for heavy ions [10]. The way to do is to specify, in the GEO file:

MSTAR Li Si will make the stopping of Li in Si be calculated with MSTAR. All other stoppings will remain ZBL00.
MSTAR will make all stoppings be calculated with MSTAR.

As many specifications as possible can be used. Please note however that the last specification override the previous ones for the same ion/target element pair. E.g., in a sample containing C, N, and Si, measured with ³⁵Cl ERDA for the GEO file where C is detected, suppose the Geo file includes at the end:

MSTAR ³⁵Cl Si

WARNING: MSTAR does not include all ions. If a stopping is requested, that MSTAR does not support, NDF automatically reverts to ZBL00.

Any use of MSTAR stopping powers should acknowledge H. Paul, A. Schinner, Nucl. Instr. Methods Phys. Res. B 195 (2002) 166. Version 3.11 of MSTAR has been available since 8 April 2003; please do not use earlier versions.

10.4 SRIM2003 (2006, 2010, etc) stopping

These are not distributed with NDF. However, NDF can call and run automatically the SRModule.exe executable distributed with SRIM. SRModule.exe and its required files (VERSION, SCOE03.dat, SNUC03.dat) must be in the NDF directory, which must have a subdirectory \se.

The user must also include the code word in the geo file:

sr03 (or SR03 or Sr03)

Calling Srim2003 each time a stopping power is needed at the beginning of a NDF simulation is very inefficient. Since version 9.2g, each time a given SRIM03 stopping is needed, it is calculated, and stored in a file in directory PATHNDF\se (where PATHNDF is the NDF path, see section 1.2 above). The next time the same stopping is needed, if it has already been stored, the file is read. This directory must be created by the user before using SRIM03.

10.5 Scaling the stopping powers

You can scale the stopping power of any element in any element, by adding one line at the end of the geometry file. E.g. to increase the stopping power of He in Si by 10% in the whole energy range, you simply introduce the line "He Si 1.1". This will not change the energy dependence of the stopping power affected. You can also scale the stopping power of

recoils or reaction products in ERDA and NRA respectively.

NDF creates an internal look-up table with stopping power values. By default, the table has 1000 equally spaced stopping power values. The user can increase or decrease this value.

WARNING:Please note that stopping of molecules cannot be scaled. This is only for elements.

WARNING:Scaling of stopping power also affects user-input ones.

10.6 User-defined elemental and molecular stopping powers

WARNING:This changed dramatically in v9.3e! What we describe here is completely not back-compatible and it is going to work very differently from previous versions.

WARNING:This is for introducing given stopping powers; fitting stopping powers works differently, with different codewords. See section 15 below.

NDF is very flexible about stopping powers. The user can specify a given stopping power for a given ion/target element pair. this can be made for any number of elements, including for the elements present in the beamfoil, stopping foil, detector dead layer, and the active layer of the detector (assumed Si).

In ERDA or NRA, the user can specify the stopping for the ingoing and outgoing ions independently.

The user can also specify a given stopping power for a given molecule (as defined in the STR file). This only works for the molecules defined in the structure file, not for the foil, dead layer etc.

There are two ways of introducing a user-defined stopping power. The first one is by simply inputting an ASCII file with the stopping data. This is assumed to be the total stopping (i.e. including nuclear stopping)

The second method is by inputting the a1-a8 parameters of the ZBL85 parameterisation. This is assumed to be the electronic stopping only, and nuclear stopping will be calculated and added to the electronic stopping.

Both methods work by adding one line at the end of the GEO file per stopping introduced.

WARNING:The stopping introduced in one GEO geometry. only affects that geometry! Suppose you measured a sample at two different angles, with the same beam. If you want to introduce a given stopping for both angles, you must do it in each GEO file separately.

10.6.1 Elemental stopping powers

For elements, the line to add is e.g.

```
elstop 4He Si si.stp
(Elstop and ELSTOP) also work.
```

After the codeword elstop, comes the ion for which the stopping is introduced. The ion must be the same isotope as in line 3 of the GEO file. After the ion, comes the target element affected. Finally, the file with the stopping data (NDF recognises automatically if it's a file with the tabulated energy, stopping values, or a file with the a1-a8 parameters).

Note that if Si is present in more than one molecule or logical element, all of them are affected unless the stopping for that molecule is also explicitly given.

10.6.2 Molecular stopping powers

For molecules, the line to add is e.g.

```
molstop 4He Si1o26.6 sio2.stp
(Molstop and MOLSTOP) also work.
```

Why did we write " Si1o26."?. This is tricky; we are inputting the satopping for a molecule as given in the STR file. Suppose that in the structure STR file the molecule was defined as e.g.

```
Si 1 o 2 6.6
```

(this is SiO₂ with the correct bulk density). All condensed, i.e. taking the spaces out and leaving capital letters as they are, this becomes

Si1o26.6

and this is what must be used!

NDF recognises automatically if the given file has the tabulated energy, stopping values, or the a1-a8 parameters.

10.6.3 Using an input file with (energy, stopping) values

If you have for instance experimental stopping values, you can input these directly as a file. The format is two columns (energy, stopping). The units are keV for the energy, and for the stopping they are eV / (10¹⁵ atoms/cm²). Please note that this one of the very few exceptions to unit conventions in NDF - eV and not keV are used.

NDF recognises automatically the file type

10.6.4 Using ZBL90 a1-a8 parameters

The given file must have a1-a8 coefficients of the ZBL90 parameterisation that you think lead to correct stopping values, in a single line, separated by spaces.

NDF recognises automatically the file type.

You must specify ZBL90 as stopping model in the GEO file.

WARNING: NDF will think that the a1-a8 values are for electronic stopping, and will calculate nuclear stopping independently and add it to the total stopping. This will also be done for molecules (in which case averaged nuclear stopping using the Bragg rule is used)

10.7 How to fit stopping powers

See section 15 below.

10.8 How to get ZBL90 a1-a8 coefficients from an input table

If you have a stopping table for a given element or molecule, NDF will not use it directly for fitting. You first have to find a1-a8 coefficients of the ZBL90 parameterisation that fit well your tabulated data, then see section 10.6 above. To make your life easier, NDF incorporates an a1-a8 fitting routine to stopping power tabulated data.

Your data must be three ascii columns; energy (in keV), stopping (in eV/10¹⁵ at/cm²), error (in eV/10¹⁵ at/cm²)
If you do not know the error, use the same value for all entries.

Then create a batch file e.g. ndf.SPC:

```
f1 charge geo str
```

“f1” is any data file: not the file with the stopping data, but some e.g. RBS or ERDA data file; it is ignored, but it must exist in the current directory.

“charge” is any number, it is ignored.

“str” is any structure file, it is ignored but it should conform to normal syntax.

“geo” is a geometry file. It must have all the normal mandatory things in a geometry file (which are ignored but must be there). It must contain one additional line at the end, which is the only important thing in all this:

```
fitsc DATA INIT_GUESS
```

where “fitsc” is the code word to tell NDF a stopping file is to be fitted.

DATA is the data file with the stopping tabulated data

INIT_GUESS is a file with an initial guess for the a1-a8 coefficients.

Simulated annealing will work, but we advise to use local minimisation. Also, at the end check the results; if the fitted curve is not close enough to the data, start with the best fitted a1-a8 as initial guess, and fit again.

The best fitted a1-a8 coefficients can be found in the result ndf.RES file. The fitted stopping curve can be found in sg0101.dat.

10.9 Charge state

WARNING: Versions up to 9.6d did not conform to this manual. They cannot be used with the charge state option.

ZBL and SRIM stopping power calculations use a calculated equilibrium effective charge state of the analysing ion. At very small depths, this equilibrium charge state is not yet reached. The effective charge state will depend on depth, and on the initial charge state of the beam.

Again, after scattering, the beam will not be in its equilibrium charge state. It will take some time to reach it.

NDF lets the user input a given effective charge state, but only for the way in. This will be in two columns, beam energy loss (e.g. zero at the surface and a positive increasing number deeper), and effective charge state (real charge, not fractional). The file must be in increasing energy loss. You must include the following line at the end of the GEO file:

cha_eff filename in
(Cha_eff and CHA_EFF also work)

WARNING: Note that the given charge state will be used only to calculate the stopping power, and charge-exchange straggling is not calculated.

11 Corrections to single Rutherford scattering

11.1 Non-Rutherford cross sections

WARNING: See sections 11.2 and 11.1.9 below if there are sharp resonances in the cross section you need to use.

11.1.1 Angular dependent non-Rutherford cross sections from SigmaCalc

Actually, not a SigmaCalc file as such, but one prepared using SigmaCalc, including the cross section at the entire angular range possible. Where SigmaCalc is available, this is the preferred way of handling non-Rutherford cross sections.

WARNING: This is not a way to input the cross section at a single angle! This is to provide the full angular dependence, once and for all. To input a single cross section, see section 11.1.2 below.

For systems where a SigmaCalc evaluation is available, SigmaCalc can be used to calculate the non-Rutherford cross section (as ratio to Rutherford) for many different angles (from very small, say 1° to very large, say 180°). Then a file is prepared with a well defined standardised file name, as follows:

The file must be named e.g.
SigmaCalc_E4He1H.dat
for ERDA of 4He on protons

SigmaCalc_R1H12C.dat
for RBS of a proton beam and a 12C target

SigmaCalc_R1HC.dat
for RBS of a proton beam and a natural C target

(i.e., the usual NDF distinction between "C" and "12C" is used.)

These files must be in a subdirectory \sigmacalc of the NDF directory.

NDF then makes a linear interpolation for the actual scattering angle.

If such a file is available for a given reaction, double scattering will include the angular dependence. This is essential e.g. for DS of ERDA of 4He on hydrogen.

The structure of the file is as follows. First, comes a header such as:

```
<Source> SigmaCalc v1.6 (r33) </Source>
<Created> File created 09-10-2009 17:17:27 by Nuno P. Barradas </Created>
<Comment> any text </Comment>
<Reaction> 16O(a,a0)16O </Reaction>
<EnergyRange> 552 1735 </EnergyRange>
<AnglesNumber> 19 </AnglesNumber>
<EnergiesNumber> 167 </EnergiesNumber>
```

The line <Source> should describe the source of the data, normally SigmaCalc, but sometimes it can be experimental data.

The line <Created> should say who and when created the file.

The line <Comment> is reserved for anything the user may wish to say.

<Reaction> should have the reaction for which the file is.

As of 9.3f, the lines above are not used by NDF, and they need to be present, but need to have any content.

The line <EnergyRange> is meant to restrict the usage of the file. If the initial beam energy is within the two values given (in keV), then the file will be used (even at energies below the minimum energy given!). If the initial beam energy is outside the range given (either higher or lower), the file will not be used. If no values are given, the full energy range as given in the file is considered.

The line <AnglesNumber> must have the number of angles for which the cross section is given. This must be in the file and must be correct.

The line <EnergiesNumber> must have the number of energies for which the cross section is given. This must be in the file and must be correct.

After this header, comes the tag

<Data>

The next line after the tag <Data> has the angles (in °) for which the cross section is given, separated by a comma, and with an initial comma (as many values as given in <AnglesNumber>), for instance:

, 1, 10, 20, 30, 40, 50, 60, 70, 80, 90, 100, 110, 120, 130, 140, 150, 160, 170, 180,

After that come lines, one per energy value for which the cross section is given (as many lines as input with the tag <EnergiesNumber>). Each line has the energy (in keV) then the cross section values (in r2r) for each angle, separated by a comma, for instance

552.1, 1, 1, 1, 1.001, 1.001, 1.001, 1, 1, 0.999, 0.997, 0.993, 0.987, 0.98, 0.973, 0.967, 0.961, 0.957, 0.954, 0.953

In the end, this file is closed with

</Data>

WARNING: A given file specifically input by the user takes precedence over SigmaCalc!

WARNING: The file must have ratio to Rutherford values, not mbar!

WARNING: If one wants to calculate DS, for ERDA, the range of values should come from 1° or 2°, up to 90° (where a 1 value can be given for all energies). For RBS; it should be from 1° or so, and up to 180°.

11.1.2 Inputing non-Rutherford cross sections from a file

First of all, you need to have an ASCII file with your cross sections. Then, at the end of the geometry file you introduce one line with two entries, the first the target element for which you want the correction, the second the name of the file, e.g.

C c_nonrut.cor

or

c d:\ndf\sigmarepository\c_nonrut.cor

You can also tell NDF what the you store non-Rutherford cross sections:

path_sigma d:\ndf\sigmarepository\

in which case

C c_nonrut.cor

Will read not from the current working directory, but from the stated d:\ndf\sigmarepository\

You can always override the path_sigma command, for instance:

path_sigma d:\ndf\sigmarepository\

C c_nonrut.cor

Si d:\ndf\sigma\si_nonrut.cor

The cross section for carbon is read from d:\ndf\sigmarepository\ but for Si it is from the specified path.

WARNING: One fine point: For cross-section purposes, “Si”, “28Si”, “29Si”, “30Si”, etc, are all treated separately as if they were different elements. So, specifying for instance the cross section for 28Si as “28Si 28si_nr.cor” will have no influence on the cross section of an element defined as “Si” in the STR file! Similarly, specifying for instance the cross section for Si as “Si si_nr.cor” will have no influence on the cross section of an element defined as “28Si” in the STR file!

11.1.3 Internally calculated non-Rutherford cross sections

Some non-Rutherford cross sections are automatically set in NDF. These are for ⁴He ERDA detecting protons or

deuterium(see sections 11.11 and 11.12 below), and for NRA for the reactions $D(3\text{He},p)4\text{He}$ and $3\text{He}(D,p)4\text{He}$ [1]. NDF will create a file with these cross-sections for your inspection, and will warn you so in the *.RES file. You can turn these automatically generated cross sections off, by simply introducing, as explained above, your own values.

WARNING: These internally calculated cross sections are interpolations, and should not be used! Users should be aware of the cross sections they use, so they should select and input files they consider to be correct.

11.1.4 Inputting all non-Rutherford cross sections for a given angle

Most detectors are at a fixed angle. The user can group together all their favourite cross section files for this angle/detector, and NDF will then check which are to be used in a given analysis. This is made via the codeword ebsfiles in the GEO file:

ebsfiles filename

The file called filename must be in a subdirectory \sigmacalc of the NDF directory (normally something like ndf\sigmacalc\). Its content is one line per cross section. Each line must have first the incident ion, then the target atom, then the file name (normally with full path, otherwise NDF will assume the files are in the subdirectory \sigmacalc of the NDF directory), then the minimum and maximum energy range (taken as the initial beam energy):

```
1H Si sinpp0d.r33 2990 3580
1H 9Be be9pp0d.r33 140 3000
1H Be be9pp0d.r33 140 3000
1H 10B b0pp0r.r33 500 3300
1H 11B b1pp0t.r33 500 3300
1H 16O o6pp0b.r33 4100 4520
1H O o6pp0v.r33 4520 5450
1H O o6pp0b.r33 4100 4520
1H 16O o6pp0v.r33 4520 5450
4He 16O o6aa0x.r33 3980 5030
4He 16O o6aa0t.r33 5030 9000
4He O o6aa0x.r33 3980 5030
4He O o6aa0t.r33 5030 9000
```

This could be a file used for a 170° scattering angle detector. Notice several things:

- All the possible cross section files are given, for different ions; not all will be used in the same analysis.
- For protons in 16O, there is SigmaCalc available, but only up to 4100 keV. So, files that cover higher energy can be given. In this case, the angle of file o6pp0b.r33 is 167.30° , which is close enough to 170° , so we use it but restricting to energies that SigmaCalc does not reach; for file o6pp0v.r33 it is 165.67° so it is a bit further away, but in any case, for energies between 4520 and 5450 keV it is the best we have available.
- NDF does not understand that 9Be and Be are the same, so we give the same file twice, once for each of those. The same is done for 16O/O.

WARNING: An explicitly input file in GEO for a given cross section takes precedence over what is in ebsfiles.

11.1.5 Format and units of user-input non-Rutherford cross sections - R33

If you specify a file which conforms to the R33 standard, NDF will read it. Please note that as of v9.2j, NDF takes the file for granted, it does not check whether it is the appropriate reaction or not.

11.1.6 Format and units of user-input non-Rutherford cross sections - two column ascii

First of all, you need to have an ASCII file with your cross sections. Except for NRA, this must be just two columns, the first one the energy in keV, the second the correction factor relative to Rutherford values. A maximum of 1000 values are allowed. Outside the range given in the file no correction is done. If you want to give the cross section in absolute values (mbarn), then the first line of the file must be m (or M).

For NRA, it is also two columns, the first one also the energy in keV, but the second absolute values. By default, NDF considers that the cross sections are differential ones (in mbarn/sr), in the laboratory system. If you want to input a total cross section (in mbarn), then the first line of the file must be t (or T)

And if you want to input a cross section in the centre of mass system, the first line must be c (or C)

To input a total cross section in the CM system, the first line is, of course
t c

11.1.7 Scaling user-input non-Rutherford cross sections - two column ascii

You may want to scale the cross section given by a constant factor. In that case, the first line must be
scale factor

where factor is the desired value, e.g. 1.5 to make the cross section 50% larger.

In the R33 format there is a field meant for this purpose.

11.1.8 Internal handling of non-Rutherford cross sections

Until NDF 9.1g, NDF created an internal look-up table with interpolated cross section values. By default, the table has 1000 equally spaced cross section values. The user can increase or decrease this value. This table still exists for purposes of outputting, but it is no longer used in any calculation of yield.

From NDF 9.1h on, NDF explicitly integrates the scattering cross section on each energy bin being calculated. This leads to slower calculations, and very small gains in accuracy (in fact, no gain in most cases).

11.1.9 Automatic division of internal layers with non-Rutherford cross sections

When very sharp resonances exist, as for protons on Al, it is convenient to have more, very thin, internal layers around the resonance, or its shape (and size) will be miscalculated due to interpolations. The price to pay will be a much slower code. The best way to do this in NDF is to introduce in the GEO file a line
Autol (or autol or AUTOL)

This will lead to automatic inclusion of one extra internal layer for each data point in the non-Rutherford cross section file. It will also make NDF calculate the spectrum not channel by channel (which, if the resonance width is of the order of the channel energy width, can lead to wrong interpolations), but internal layer by internal layer. Straggling will also be calculated internal layer by internal layer. Also, NDF will increase the size of its internal arrays describing the internal layer structure and non-Rutherford cross sections to the maximum allowed by memory constraints. Altogether, the run time may be one order of magnitude slower than normal NDF.

WARNING: This option is the default in NDF whenever a non-Rutherford cross section is included, and it may lead to much higher execution times (factors of 30 have been reported when the cross section file has many points). In most cases the difference will be minimal or none, if the cross section has no sharp resonances.

To turn the option off even when non-Rutherford cross sections are used, write
Autol off (or Off or OFF)

11.1.10 Cross section at 0 energy

If no value for the cross section at 0 keV is stated in the file, NDF assumes for RBS and ERDA a ratio-to-Rutherford equal to 1. For energies between 0 and the minimum energy given in the cross section file, a linear interpolation is made.

For NRA and NRP, at 0 keV a 0 mb cross section is assumed. For energies between 0 and the minimum energy given in the cross section file, a linear interpolation is made. Note that this may lead to problems for NRP, since the integrated cross section can become a large and unrealistic value.

11.2 Accurate simulation of resonances

The energy spread before scattering means that, for a given depth of interaction, the beam contains particles with different energies, thus leading to a spread of cross sections for interaction at that depth. The Rutherford cross section changes slowly with energy, so this effect will be very small and can be ignored. However, if there are sharp resonances in the (non-Rutherford) cross-section, the effect can be large (particularly if the resonance occurs deep in the sample, where energy spread is large), leading to a broadening of the resonance signal.

There are three effects occurring:

1. The total cross section for interaction at a given depth becomes an average over the different beam particle energies before interaction.
2. The average energy of the beam after interaction also changes.
3. The exact shape of the beam energy distribution after interaction also changes (in general, it becomes a complex shape depending on the shape of the resonance).

The accurate and completely correct way of calculating this effect involves a double integral calculation [11], which is extremely time consuming (3 to 4 orders of magnitude slower) and difficult to implement. NDF does not implement it and there are no plans for doing so.

What NDF does is to calculate correctly points 1 and 2, that is, the correct average cross section for interaction at a given depth, and the correct average beam energy after interaction at that given depth. It will not calculate point 3, and it will take Gaussian shape for the energy distribution throughout. In this way, the increase in calculation time is very small, in fact not noticeable.

NDF will do this automatically whenever the user inputs a non-Rutherford cross section file as described in section 11.1 above, even if there are no resonances.

NDF will not do this correction automatically for Rutherford cross sections: the effect still exists, but it is very small.

To turn the correction off, write at the end of the geo file:
nores (NORES or Nores will also work).

To calculate this correction also for Rutherford cross sections, write at the end of the geo file:
allres (ALLRES or Allres will also work)

Or for only some elements:

Allres Si Au

WARNING: This correction only makes sense if the energy spread before interaction is calculated, therefore it is only applied if straggling is calculated, either by NDF (Bohr or Chu, see section 12.4 below) or with DEPTH (see section 12.3 below) or MDEPTH (see section 12.1 below).

11.3 Screening

By default, NDF calculates Andersen et al. angular dependent screening [12]. It may also calculate L'Ecuyer screening [13], as well as screening calculated from different potentials: Universal, Moliere, Lenz-Jensen and Bohr. To force NDF to calculate one or the other, or to turn screening off altogether, write at the end of the geo file

For Andersen screening (no change will happen):
screen a
(Screen or SCREEN, and A, would also work)

For L'Ecuyer screening:
screen e

For Universal potential screening:
screen u

For Moliere potential screening:
screen m

For Lenz-Jensen potential screening:
screen L

For Bohr potential screening:
screen b

For no screening:
screen n

11.4 Double scattering

Double scattering corresponds to the ion undergoing two large-angle collisions before being detected. NDF currently only calculates it for RBS and ERDA (including forward scattering and transmission) [14]. It leads mostly to a low energy background, and to an increase of the signal of bulk samples at low energies. Please note however that other effects, such as inaccurate stopping power values, slit scattering, or as inaccurate energy resolution as a function of

depth, all lead to extra distortions of the low energy signal.

WARNING: The DS routine for ERDA calculates only the three following situations: ion scatters then produces recoil in sample; ion creates recoil which then scatters in sample; and ion is forward scattered, producing a recoil in the stopping foil (in case the ERDA recoil also exists in the foil). Note that these are the only contributions calculated; processes involving two recoil events, or scattering of recoil in the foil, are disregarded. Furthermore, the name of the recoil must be the same in the sample definition and in the foil definition (for instance, if it is 1H, defined as 1H in the STR file, it must be the also defined as 1H in the foil composition in the GEO file).

WARNING: During a fit, the double scattering is calculated only when the χ^2 has changed by a certain amount (a factor of 2 in the current implementation), in order to save time. It is calculated again for the final fit. Please note that small changes in the simulation have a large effect on χ^2 but only a small effect on double scattering.

WARNING: A strong issue is for non-Rutherford cross sections that have a strong angular dependence. NDF always uses the non-Rutherford correction for the nominal angle, which may lead to severe problems. The only exception is for a-ERDA of 1H, where the correct angular dependence is used, but only in the energy range up to 3 MeV. At higher energies, NDF will do something untested.

The double scattering contribution is saved as a file, called bDFsf.DAT (e.g. NDFds0102.dat for the second spectrum of the first sample in the batch file named NDF.SPC). The file is two columns (channel number, double scattering yield), with the same number of channels as used in the fit or simulation.

For verbosity 2 and above, for ERDA a second file is created, called bDFsfD.DAT, which includes the separated contributions to double scattering: ion scatters then produces recoil in sample; ion creates recoil which then scatters in sample; and ion is forward scattered, producing a recoil in the stopping foil (in case the ERDA recoil also exists in the foil). Note that these are the only contributions calculated; processes involving two recoil events, or scattering of recoil in the foil, are disregarded.

11.4.1 Calculation of double scattering - automatic control

NDF can calculate double scattering as an option. This is only implemented for RBS. This increases the computation time very much indeed, depending on the options chosen. To introduce double scattering, introduce an extra line at the end of the geo file:

```
ds full
or
ds precise
or
ds normal
or
ds fast
```

(Ds, DS, Full, FULL, etc, will also work)

The difference between “full”, “precise”, and “normal” is the following:

“Full” will make NDF calculate all the scattering events involving all possible first scatterer/second scatterer pairs. It will also calculate for all possible directions of the first scattering.

“Precise” will make NDF not calculate those scattering events that carry, all together, only 1% of the total cross-section. It also reject the directions of first scattering that, all together, account for only 1% of the total cross-section. This will reduce the calculation time without significant loss of accuracy.

“Normal” will act like “precise”, but with the threshold at 3%. In almost every situation, no significant loss of accuracy is noted, but calculation time is reduced, depending on the sample, by 1 or 2 orders of magnitude.

For instance, suppose a 500 nm Au film with a 5 nm C layer on top, over an Si substrate with a 2 nm natural oxide, measured with 2 MeV He. Both the “precise” and “normal” options will make NDF not calculate events that involve C or O. This means that only Au/Au, Au/Si, Si/Au, and Si/Si will be calculated: 4 pairs, instead of 16 with the “full” option.

“Fast” will act like “normal” on first call; but it will never be recalculated. That is, the ds will be calculated for the initial structure and initial experimental parameters (including charge), and it will not be recalculated during the fit. Furthermore, if it has been calculated in a previous run (leading, e.g. if it was sample 03, geometry 02, to file DS0302.DAT), with the same number of channels, then the previously calculated DS will be used. Scaling also works, e.g.:

ds fast 2.4

Furthermore, sometimes the shape of the spectrum at very low energies (e.g. the substrate signal at very low energies) is not relevant to the data analysis, and the DS contribution is only required for the background at energies below the high energy signals from the heavy elements. There is an option to make NDF ignore those low energy contributions, which in some cases may speed up the calculations by still one order of magnitude. For instance, on the example above (Au on Si), the Si/Si scattering will be ignored. The extra line should then be:

ds <full, precise, low> nolow

(NOLOW or Nolow will also work)

Please note that the nolow option works independently of the full/precise/norma optionl. This means that, even if you choose “full”, if you also use “nolow”, there will be a (possibly strong) restriction on the scatter pairs considered (but not on the angular directions).

11.4.2 User-input double scattering contribution

The user may input a previously calculated double scattering contribution via a file, normally a previously calculated DFsf.DAT file. The structure of this file is two columns (channel number, double scattering yield), with the same number of channels to be used in the fit or simulation. For instance, if the sample structure is fairly well known, it is not efficient to calculate double scattering all the time. Small changes in the sample lead to small changes in the double scattering contribution only. So, the user can first calculate ds as explained above, and then use the corresponding file for further fits. The syntax in the geometry file is:

ds file_name

or even

ds file_name 3.5

to scale the input file by a factor of 3.5

Now, suppose a series of samples, all using the same GEO files. Reading different DFsf.DAT files would in principle require separate GEO files, to introduce the different names. This is not necessary, all that is needed is to use the “fast” option, so the correct previously calculated files will be used for each sample and geometry (or a new one will be calculated if it does not exist already). For instance, for sample 03, geometry 02, file DS0302.DAT will be read, while for sample 05, geometry 02, it will be DS0502.DAT

11.4.3 Recommended usage

WARNING: Double scattering calculations are not as accurate as single scattering, and theory is not 100% correct, particularly at grazing angle [15]. Please read and understand [14].

The DS calculation is normally slow. The best thing to do is first to a rough simulation with no DS. Then calculate DS with “normal” or “precise” option. NDF will write out the calculated DS contribution in the files DSsf.DAT. Then, in a fit or further simulations, read in these files instead of calculating again. As long as the profile has no major changes, the DS will remain more or less the same, which is normally good enough.

As for other options (see section below):

DSMS has no effect at non-grazing angles. At grazing angles, it may make calculation slower by one order of magnitude. It will only have a measurable effect when very thin films with heavy elements are present (e.g. 3 nm Au buried in C). So it should only be used in that case.

NGR should be used for test purposes only. At grazing angle of incidence, it will lead to a wrong (much too small) DS.

CO ds_co should be used for test purposes only.

NPL should be used for test purposes only. The effect is small in most cases.

NCO should be used for test purposes only. The effect is small in most cases.

Finally, we stress that the spectral shape at low energies, as well as the low energy background, depend on other factors, not only on DS. Some of these are:

Straggling: even using MDEPTH, at very low energies possibly severe distortions may arise

Stopping power: inaccurate values will distort the spectral shape.

Slit scattering: it has been proved that scattering of the beam on beam shape defining slits will lead to additional counts at low energies; in some experimental setups, this effect will be larger than DS [16,17].

Unaccounted for effects, for which no explanation currently exists, have been claimed to exist [18].

11.4.4 Calculation of double scattering - manual control

The user may, if they so wish, have (almost) full control on what NDF does. **WARNING:** This is not recommended, it is for test purposes only. Do not use it.

The incident ions are first scattered into the whole sphere (4pi). The number of different first scattering directions is `n_scatt`, an even number, between 100 and 5000.

Only trajectories where both scattering angles (first and second) are larger than a certain cutoff value `scatter_cutoff` (another input parameter, in degrees) are considered as double scattering (except for grazing angle experiments, see the next point).

To account for trajectories at grazing angle of incidence, trajectories where the path of the ingoing ion changes by a factor `path_cutoff` larger relative to single scattering are also considered, independently of the value of `scatter_cutoff`.

Finally, to reduce calculation time at the expense of accuracy, the internal layer division for double scattering may be made coarser than the standard value for single scattering (see section 9 above), by a factor `lay_coarse` (larger than or equal to 1).

NDF will then calculate all possible double scattering events (including ones that lead to neglectable contributions, such as O/O (first/second scattering) in a sample with a 1 nm surface oxide). To specify these parameters explicitly the line in the geo file should be:

```
ds n_scatt scatter_cutoff path_cutoff lay_coarse
```

You may restrict the elements on which double scattering occurs - only for manual specification of the parameters as described above. For instance, in a Au film on Si, with a C surface contamination, double scattering Au/Au (first/second scattering), Au/Si, and Si/Au all have an important role, Si/Si may be ignored unless you are very interested in the very low energies, and all double scatterings where C is involved may be ignored. You may introduce something like:

```
ds 500 20 2. 1.5
+Au -Si 0C
```

which means that Au is allowed to participate, in principle, in all double scattering events, Si only in those events where the other scattering is from a "+" element (in this case, Au), and no double scattering event involving C is calculated. In this example, the events calculated will be Au/Au, Au/Si, Si/Au only. Si/Si is excluded due to the "-" for the Si, and Au/C, C/Au, Si/C, and C/Si are all excluded due to the "0" sign for the C. **WARNING:** this line must be located immediately below the `ds n_scatt scatter_cutoff lay_coarse` line, and ALL elements in the str file must be present. It will not work with the default option (where only "ds" was introduced in the geo file).

The options that are available (some of them only for test purposes, with in principle no interest for real analysis) are:

+Si means all events will be calculated, unless restricted by a "0", "/" or "\" code.

-Si means that events with other elements with "-" code will not be calculated.

/Si means that element/Si double scattering events will not be calculated.

\Si means that Si/element double scattering events will not be calculated.

=Si means that Si/Si will not be calculated.

For instance,

+Au /Si 0C means that only Au/Au and Si/Au are calculated: 0C excludes all events involving C; /Si excludes Si/Au

-Au /Si 0C means that only Si/Au is calculated: 0C excludes all events with C; /Si excludes Au/Si; -Au excludes Au/Au

/Au +Si 0C means that only Si/Si and Au/Si are calculated

/Au -Si 0C means that only Au/Si is calculated

-Au /Si and \Au -Si are equivalent: in both cases, only Si/Au is calculated

Advice: for a heavy element film H on a light substrate L, the only double scattering events worth calculating are H/H and L/H. Most economic usage is hence +H /L.

However, in a sandwich L/H/L, this is no longer true, L/H must also be calculated, and most economic usage is +H -L.

For complex structures with several heavy and several light elements, most economic usage is +H1 +H2 ... -L1 -L2 ...

11.4.5 Further options

To calculate double scattering efficiently in cases where it is a very thin film that is interesting, but incidence is not grazing, as for instance in thin self-supporting films (or very thin films on an uninteresting substrate), you need to put one extra keyword in geo file (after the DS keyword):

```
ds_force
```

Furthermore, NDF calculates particle loss as the beam penetrates the sample. To turn this off, write at the end of the geo file:

```
npl
```

Also, NDF does not calculate trajectories with scattering angle less than the cut-off angle when the path length has changed less than 3.11% relative to single scattering. To make NDF calculate these as well, write at the end of the geo file

```
nco
```

Between a 1.0311 change and a ds_co change in the length of the trajectory as compared to single scattering, NDF makes a linear interpolation from not calculating to calculating all. Ds_co is 1.5 by default. You can change it by adding a line at the end of the geo file (where ds_co is a number):

```
co ds_co
```

NDF may calculate a lateral spread correction due to multiple scattering, which may be important in ultra-thin films measured at grazing incidence. Tests suggest the correction is not important. To turn the correction on (calculation time will increase), add a line at the end of the geo file:

```
dsms
```

To make NDF not calculate the trajectories where one of the scattering events has angle smaller than scatter_cutoff, , add a line at the end of the geo file:

```
ngr
```

11.5 Pile-up correction

When two backscattered particles hit the detector at the same time (or within the detector's response time), the output will be a single signal, with energy somewhere between the sum of the two original pulses and the largest of them.

In two pulse pile-up, two events i and j , with amplitudes E_i and E_j , are lost from the recorded spectrum, and one event k is gained with amplitude E_k . E_k depends on the relative amplitudes and on the time separation between i and j , respecting $\max(E_i, E_j) \leq E_k \leq E_i + E_j$.

For instance, if their energies were 0.7 and 0.8 MeV, instead of two corresponding signals, a single pulse with between 0.8 and 1.5 MeV will be created (depending on the time and amplitude difference of the two pulses). That is, the spectrum will lose 2 counts at low energies and gain 1 count at high energies. This leads to a distortion of the shape of the spectrum.

If a pile-up rejection system is used, counts are still lost, but most of the counts that would be gained are discarded. Only those pulses that arrive within a time shorter than the resolving time T_{pur} of the PUR system are gained, with $E_k \cong E_i + E_j$.

The algorithm Wielopolski and Gardner algorithm assumes Gaussian shaped pulses, and ignores tail pile-up. The Molodtsov and Gurbich algorithm works for realistic pulse shapes and includes tail pile-up. The main differences will be when high count rates are used, and when a pile-up rejection system is used. The lost counts calculated with MG can be double those calculated with WG. Actually, with PUR, the lost counts depend very much on what the PUR system of the amplifier used does. WG probably calculates too little of these lost counts; the MG algorithm is correct for its assumptions, but in fact the calculation requires access to the real spectrum for perfect electronics and counting down to 0 keV, which is not in practice attainable. Given real life limitations and different electronics, one can expect some deviation between calculation and data, particularly for high count rates.

Whenever PUR is off, the MG algorithm should be used. When PUR is used, it is not clear, since different amplifiers may implement slightly different actions when two pulses come within the PUR window.

In model MG, the parameters required are:

- The realtime (collection time), to be introduced as the first yield value in the data files.
- The shaping time T_w of the amplifier.
- The resolving time T_{pur} of the PUR system, if used; otherwise, input a zero or nothing
- An integer, 1 for CR-RC filter, 2 for CR-2RC, default is 2 (if nothing is input)
- the R factor, default is 1000.
- A baseline restorer value blr, in keV; if the amplitude of the first pulse is already smaller than the blr limit when the second pulse arrives, there is no pile-up. Default is 0 (no baseline restorer circuit)

- An integer, 1 if on pile-up leading to a bi-modal signal, the 1st peak is not lost (so the effect of pile-up is to change the amplitude of the second pulse, while the 1st pulse is also counted), 2 if it is lost (so two pulses are lost and one is gained). Different amplifiers work differently! The default is 1 (first pulse not lost to pile-up). For unimodal sum pulse, this has no effect.

In models CJ, GA, and WG, the parameters required are:

- The realtime (collection time), to be introduced as the first yield value in the data files.
- The shaping time T_w of the amplifier.
- The resolving time T_{pur} of the PUR system, if used; otherwise, input a zero.
- A proportionality factor f between the time T_p required for the pulse to reach the maximum amplitude in the amplifier and the shaping time $T_p = f T_w$. By default, this is 1.9 in NDF.

Given that the models are not 100% accurate, and the real time is sometimes not very well known, some fiddling with the real time and T_w may be necessary.

11.5.1 Algorithm of Molodstov and Gurbich

We call this “MG” algorithm

Slow (if you have a very old computer) and accurate. We follow the algorithm given in [19]. We thank Alex Gurbich for all the discussions and help.

The WG algorithm is for two pulse pile-up. However, NDF will, when the pileup spectrum amounts to 1% or more of the yield, make an approximate triple pile-up calculation.

WARNING: This algorithm may be slow on old PCs. It goes with n^3 , while the Amsel et al. and CJ algorithms go with n^2 .

11.5.2 Algorithm of Wielopolski and Gardner

We call this “WG” algorithm

Slow (if you have a very old computer) and accurate. We follow the algorithm given in [20]. We thank Matej Mayer (author of SIMNRA) for pointing us out that this algorithm exists and is very precise.

The WG algorithm is for two pulse pile-up. However, NDF will, when the pileup spectrum amounts to 1% or more of the yield, make an approximate triple pile-up calculation.

WARNING: This algorithm may be slow on old PCs. It goes with n^3 , while the Amsel et al. and CJ algorithms go with n^2 .

11.5.3 Two-event pile-up: pure autocorrelation algorithm

We call this “CJ” algorithm

Very fast. Based on the algorithm previously (until 10Feb05) used by NDF, but with important differences. The shape of the gained counts is calculated from a simple autocorrelation of the spectrum. The amplitude for both gained and lost counts are calculated from a probabilistic model that takes into account T_w and T_{pur} . This does not take into account cases where $E_k < E_i + E_j$, so it leads to some distortion when PUR is off.

11.5.4 Two-event pile-up: algorithm of Amsel et al.

We call this “GA” algorithm

Very fast. Essentially the same as the previous, but with the correction given in [21,22] and in [23] (with $\gamma=0.2$). This leads to taking into account, at least partially, those events where $E_k < E_i + E_j$. It is thus better when PUR is off.

11.5.5 Automatic pile-up correction on your data

The first channel in your data MUST be the realtime in seconds (in data type 4, this is the second channel, not the first! See section 5.2). If this value is zero, no pileup correction will be made. Further, you must introduce one line at the end of the GEO file, containing the word ‘pile’, ‘PILE’ or ‘Pile’ and a number, which will be taken to be the pileup factor w in Eq. (1). WARNING: If the first channel in the data is an arbitrary number with no relation with the real time, an

arbitrary correction with no relation with reality will be done.

So, what you have to do is:

1. Make sure the first number in your data is the live time (except for data type 4 - Lisbon). The second number can be the real time, but it is not strictly necessary.
2. Introduce a line at the end of the geometry file with the keyword 'pile' followed by the algorithm codeword (CJ, GA, or WG), and the values of Tw, Tpur, f, e.g.

pile CJ 1.e-6 0 1.9

Tw and Tpur are in seconds, so if the shaping time is 1 μ s, you have to write 1.e-6.

If no PUR system is used, introduce 0 for Tpur.

The default value for f is 1.9, but it is different for different amplifiers.

The following will also work, e.g.

pile Ga 2.e-6
(implying no PUR system, and f=1.9)

pile wg 1.5e-6 0.4e-6
(implying f=1.9)

If, in the GA algorithm, you want to change the value of γ from 0.2 to some other value, introduce it at the end, e.g. for $\gamma = 0.35$:

pile GA 1.e-6 0 1.9 0.35

11.5.6 Low level discrimination and pile-up

The low level discrimination setting in the MCA means counts from channel 1 to channel ch_LLD have no counts. As the pileup calculation uses the actual data collected, this will lead to a distortion in the calculation. A not-so-good solution implemented in NDF is for the user to give the LLD channel, normally taken at the lowest channel with most counts before dropping to zero. For the pileup calculation (and only for that!), NDF will take channels 1 to ch_LLD as having the same counts as in ch_LLD. You must introduce a line at the end of the geometry file:

lld ch_LLD

(LLD or Lld instead of lld will also do)

Please note that the ch_LLD is for a 512 channel spectrum! If your raw data has 1024 channels you must divide the real channel value by two; if it has 256 channels you must multiply it by 2.

From version 9.3f on, NDF will try to determine automatically some LLD value. This is overridden either by a user input value as above, or by the new command

lld off

11.5.7 Pileup output file and recalculation from file

The calculated pileup is written to file bUsf.DAT (see section 8.2.9 above).

The pile-up is uniquely determined from the data, the real time, and the algorithm and parameters used. The depth profile in the fit is irrelevant to the calculation. So, from version v8.2d on, if pileup has been already calculated in a previous run, and if all those parameters remained the same, including the number of channels in the fit, the file bUsf.DA is read, and no calculation is made. This means that the MG algorithm in fact should always be used, since it only takes time once.

11.5.8 Number of channels in calculation

The pileup is calculated based on 256 channels (or actual number of channels after compression, if smaller than 256). This is because the calculation goes with N^3 , to save time. Normally, no visible effect arises from this. Very seldom you may notice a very slight oscillation near a peak or edge. This will not have an actual effect on fitting, but you can instruct NDF to make a full channel calculation, by creating the file PILESPEED with the number 1 as content. If it has a 0, it will recalculate for the compressed 256 channel standard. If it does not exist, the standard 256 calculation is done.

If it has a number which is a power of 2, that will be the number of channels for the pileup calculation. Note that it is the user's responsibility to make this number not larger than the actual number of channels!

11.5.9 Livetime vs realtime

Whenever the livetime is also input (only possible in Surrey and Libon data types), NDF also corrects the actual charge for the calculation. If the livetime was 90%, then the actual charge in the calculation will be 0.9 the given charge, which should then be the charge before livetime correction.

11.6 Low energy yield correction

When using straggling, the yield at low energies becomes distorted: due to the statistical nature of energy loss, a correct simulation must follow particles up to depths greater than the maximum depth at which particles with the average beam energy may be scattered and still emerge from the sample with positive energy. NDF implements an approximate solution to this problem. By default, NDF will not apply the correction, since it increases the calculation time around one to two orders of magnitude (a few seconds in a modern PC). To turn it on, write at the end of the GEO file:

```
low_e
```

It is strongly recommended that this option be used in conjunction with the option ALLRES (see section 0 above), since at very low energies it is essential to integrate the cross section over the beam energy distribution before scattering.

WARNING: This option is not compatible with the use of roughness. If roughness is on, the low_e option is disabled. Or vice-versa.

WARNING: Only Bohr/Chu straggling are taken into account in the calculation, DEPTH is not completely supported as yet. It is partially supported, but since the results are very sensitive to the tails of the energy distribution, several problems arise, particularly at grazing angles: non symmetry of scattering along trajectory, non Gaussian distributions, and others. You can try, but at angles between beam (in or out) and sample surface smaller than around 30°, the results are expected to be pretty bad. Occasionally they are pretty good, so do try. If nevertheless you want to use DEPTH, see the dperror option in section 12.

WARNING: Only RBS and ERDA in reflection geometry (not transmission) are supported.

11.7 Ad-hoc substrate plural scattering / channelling correction.

WARNING: The double scattering calculation given in section 11.4 above has mostly superseded this section.

11.7.1 Basics

As NDF is fitting theoretical spectra to the experimental data, it is important to get the generated functions right. That is, in principle we should have to introduce full plural and multiple scattering calculations, as well as channelling/dechannelling yield calculations. This is normally done by extremely slow Monte Carlo methods, hence impracticable for our purposes.

Nevertheless, we want to get the shape of our substrate signal right. NDF allows you to introduce a multiplicative correction to the shape of the signal of any specified target element. This correction is just a third degree polynomial. You introduce in NDF the coefficients, together with a channel number above which the correction no longer applies, by adding two lines in the geometry file, e.g.

```
Si
```

```
220 1.72 -0.0126 7.591e-5 -1.494e-7
```

This polynomial actually has value 1 at channel 220, and it is a plural scattering ad-hoc correction. It will increase the Si yield at lower channels.

We now give more information:

11.7.2 The problem

Plural scattering (as well as other effects such as deviations from the Rutherford cross sections at low energies, and inaccurate values of the stopping powers) leads to deviations of calculated spectra from the measured yield, even if the structure of the samples is perfectly known. In general, the measured yield at low energies is larger than the calculated theoretical yield.

11.7.3 A (relatively general) ad-hoc solution

It is possible to account for this effect in an ad-hoc way, provided a reference sample can be obtained. For instance, in a series of SiC samples implanted with some element, the RBS spectrum of pure SiC can be used as a reference. The heavier Si will in general be responsible for the deviation, so we want to find a correction for the Si only.

Let us define the *experimental* yield as $Y_{\text{exp}}(\text{channel})$, and say it is the sum of the *experimental* partial yields due to all elements X, $Y_{\text{exp}}(\text{X,ch})$, as

$$Y_{\text{exp}}(\text{ch}) = \sum_X Y_{\text{exp}}(X, \text{ch}), \quad (1)$$

and that the *theoretical* partial spectrum corresponding to Si is $Y_{\text{theory}}(\text{Si}, \text{ch})$. Then the plural scattering correction factor $P(\text{ch})$ will be

$$P(\text{ch}) = Y_{\text{exp}}(\text{Si}, \text{ch}) / Y_{\text{theory}}(\text{Si}, \text{ch}). \quad (2)$$

If we can assume that the implanted extra element will not change significantly this correction factor, we can use it in the fits of the real interesting data. We do this regularly at Surrey, so far with very good results.

11.7.4 What you have to do

1. You have to have a reference sample from which you trust you can obtain a valid plural scattering correction that will work in your particular problem. You get an RBS spectrum taken in the same conditions as your real interesting data.
2. You fit it with NDF. You give as fitting region of interest only channels where you believe plural scattering plays no role (that is, in the high energy part of the spectrum). You HAVE to convince yourself that the fit is good. Look at the partial spectrum of the element you are interested in: do you believe it? For instance, if you have an homogeneous multi-element reference sample, you should use as minimum layer thickness in the structure file .STR a very large value, say 10000 (think for yourself why; either understand it or don't proceed!).
3. We provide you a little utility program called MS.EXE to calculate Eq. (2) for you. Just run it and follow the instructions. You will be prompted for an output file name that will contain the result of Eq. (2), i.e. $P(\text{ch})$.
4. This is NOT all! Plot $P(\text{ch})$ using whatever graphics routine you use. It is likely that it will have kinks and peaks near edges, where the fit wasn't perfect. Look carefully at it, and decide which points you have to throw away. Normally, in the low channels it will be above 1, and then it will stabilise around 1 (with large statistical fluctuations!), and then will fluctuate wildly (above the highest-channel edge, normally): throw away all points above that!
5. Fit it, using whatever routine you have, with a third degree polynomial as

$$P_{\text{fitted}}(\text{ch}) = a_0 + a_1 \cdot \text{ch} + a_2 \cdot \text{ch}^2 + a_3 \cdot \text{ch}^3. \quad (3)$$

You also need to define a cut-off channel $\text{ch}_{\text{cutoff}}$ (integer) above which the correction is not going to be used: normally $P_{\text{fitted}}(\text{ch}_{\text{cutoff}}) = 1$. $\text{ch}_{\text{cutoff}}$, a_0 , a_1 , a_2 , and a_3 define the plural scattering correction you want: they are the numbers, in this order, that you put in the proper place in the geometry file *.geo (see section 11.7.5).

11.7.5 Dirtier trick

Actually, sometimes you don't have a reference sample, and nevertheless you want to correct for the shape of the background. Then you can put just any polynomial that will fit.

Or, you can just give a first guess of what the correction should be at channel zero (a_0), and what it should be at $\text{ch}_{\text{cutoff}}$ ($P_{\text{fitted}}(\text{ch}_{\text{cutoff}})$). Then NDF will adjust this during the fit. You have to introduce in the geometry file *.geo the following lines (e.g. for Si)

```
Si
ch_cutoff, a0, P_fitted(ch_cutoff)
```

If $P_{\text{fitted}}(\text{ch}_{\text{cutoff}}) = 1$, the correction at $\text{ch}_{\text{cutoff}}$ will be kept constant at $P_{\text{fitted}}(\text{ch}_{\text{cutoff}})=1$ during the fit.

If $P_{\text{fitted}}(\text{ch}_{\text{cutoff}}) \neq 1$, the correction at $\text{ch}_{\text{cutoff}}$ will also change during the fit: it is then a completely ad-hoc correction that will change the shape of the signal (it can be used as a channelling correction).

11.7.6 Wim trick

NDF can try to determine the correction by itself, based on a ROI given by the user. It works like this:

- As in the other uses described above, the user gives the cut-off channel $\text{ch}_{\text{cutoff}}$ (integer), as well as the value $P_{\text{fitted}}(\text{ch}_{\text{cutoff}})$ that the polynomial should have at the cut-off channel.
- The user gives three ROIs, where NDF will try its best to make the fit close to the data by using the polynomial correction for a given element. (be creative: if you only need one ROI, introduce it three times.)
- In a simulation, NDF will calculate the polynomial that leads to the best fit in the ROIs.
- In a local search fit, NDF will recalculate the polynomial at the beginning of some iterations.
- In SimAnn, NDF will recalculate the polynomial a few times (don't ask), and then in local search at the beginning some iterations. Given that SimAnn changes things randomly, the algorithm may produce funny results.

Usage is (e.g. for Si)

```
Si
ch_cutoff P_fitted(ch_cutoff) ROI11 ROI12 ROI21 ROI22 ROI31 ROI32
```

WARNING: This is best used for initial guesses that are quite good, including charge. This may produce terrible results. Tested for RBS only, but should work for other techniques.

WARNING: Pay attention to the ROI! If the lowest channel is below what NDF calculates (in a fit, it is the ROI for the fit plus a number of channels that depends on the FWHM and straggling), the results may become catastrophic. Also, if the ROI falls in a region where the given element has no counts, unpredictable results may happen.

The cut-off channel does not need to be above the last ROI.

You can also have NDF calculate the value at the cut-off channel. In that case, it will not be 1, and it will always be a fit parameter as well. Simply input:

```
Si
chcutoff ROI11 ROI12 ROI21 ROI22 ROI31 ROI32
```

11.8 Ad-hoc thin film plural scattering correction

WARNING: The double scattering calculation given in section 11.4 above has mostly superseded this section.

A thin film of a heavy element will lead to some plural scattering at energies lower than it (if it is thick enough, extra counts due to plural scattering will start to appear already in the film). This leads to a background that can be in some cases disturbing, for instance if you are trying to fit a relatively small peak at energies lower than the heavy element film, or in extreme cases like Au on a C substrate, the background due to the Au plural scattering can be important when compared with the C signal height. Also, if you are looking at diffusion from the thin film, it would be nice to get rid of those extra counts.

The background is relatively constant for some energy range, below which it increases. It has a film thickness t dependence approximately as t^{2-3} [24].

NDF allows you to do an extremely ad-hoc correction for this background. This is extremely ad-hoc, not general, and indefensible from a physical point of view. That is, it is almost certainly wrong. It might be helpful on occasion, we used it exactly once (unpublished results although it was nice work!), so it's a NDF feature and here it goes. Suppose you have thin films of the same material with different thicknesses. You measure the background level below the peak. Normalise to 1 μC charge. Plot against the width Δx (in channels!) of the film. Fit it with a $c \cdot \Delta x^2$ equation, as shown in Figure 1 for Au thin films.

You can give to NDF the fit parameter c (in this case $0.00557=0.00258/0.463$), the low and high channels where the film is, and the element to which the correction corresponds (at the end of the geometry file you include a line "Au 0.00557 xmin xmax"). Below the low channel, NDF will subtract a constant background as given by the figure (corrected for the charge of the spectrum in question). In the region of the film it will subtract a background that depends on Δx as in the figure. Please note that NDF will subtract the correction from the data, and fit the corrected data. This will work only on an ad-hoc basis. **WARNING:** Do not try this at home.

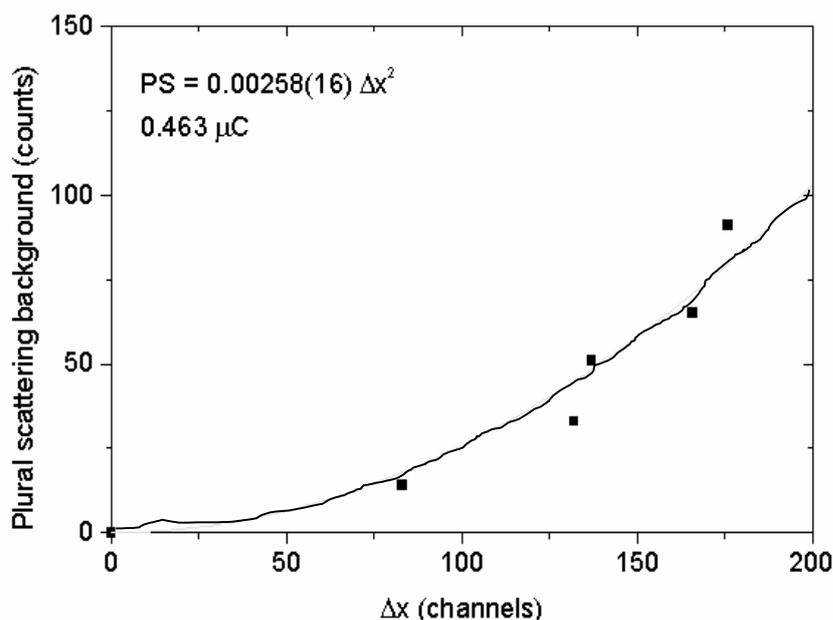


Figure 1. Plural scattering background correction as function of the channel width of a Au film.

11.9 Generate apparent cross sections that lead to a perfect fit to your data

Suppose you are analysing Elastic Backscattering data, with non-Rutherford cross sections. Suppose also that you do not have reliable cross section values for your experimental conditions, but that you have some reference sample. For instance, you could be interested in analysing samples with carbon using a proton beam, and you have available a bulk carbon sample.

NDF has a function that allows you to calculate apparent cross sections that lead to a perfect fit to the data from the reference sample. All you have to do is to run a simulation of the reference sample for the correct experimental conditions, using any cross section values you may have, no matter how wrong they are, including Rutherford values. The simulation should be on single-sample; multiple spectra can be made, as long as they are all similar (same technique), for instance with different beam energy.

You must create a file called NON_RUTH.COR, with two lines: the first one is the number of the element (order in the structure file) for which you want to determine the cross section, the second the name of the output file with the cross section values. This should be 6 characters or less. NDF will add the geometry number 01 etc, and .dat extension. For instance:

```
3
pp
```

Means the apparent cross sections will be generated for the 3rd element, with file names e.g. pp01.dat, pp02.dat, etc, one per geometry.

If you want the output to exclude the initial and final calculated values (which can be bad due for instance to multiple scattering changing the yield at low energies, and edge disturbances at high energies, then include a third line with the number of channels to exclude at the beginning and end. For instance to exclude the 30 low energy channels and the 5 high energy channels, :

```
3
pp
30 5
```

You can use the output file as non-Rutherford correction as given in section 11.1 above. You should probably, however, first smooth the file up a bit.

11.10 Effects of multiple scattering on energy, angular and pathlength spread

11.10.1 Effect on energy spread

This can be included using DEPTH, as explained in chapter 12.

11.10.2 Effect on scattering angle spread and pathlength spread

These two corrections are deeply related to each other and are always applied together. The multiplicative fudge factor on both is also always applied together. There are some severe problems with the model:

1. The integration extends only to 3 sigma, to avoid problems with the long MS tails.
2. To calculate the scattering angle spread, the angular spread on the way in and on the way out are added.
3. The underlying assumption of symmetry of MS in respect to the trajectory is only strictly valid for normal incidence and 180° detection.
4. Rutherford scattering is assumed to calculate the cross section. The effect will not be well calculated for resonances that change rapidly and non-linearly with angle.

The model calculates the effect of scattering angle spread on the effective cross section and kinematic factor at each depth.

The model also calculates the average path length on the way in, correcting the energy loss and the amount of scattering centres crossed, and the average path length on the way out, correcting the energy loss.

We expect that only an order of magnitude is calculated in this way. Thus we allow the user to introduce a multiplicative fudge factor.

NDF will not do this correction automatically. To turn the correction on, write at the end of the geo file:
f_as (F_AS or F_as will also work).
or, to introduce a multiplicative fudge factor:
f_as factor

WARNING: This correction is only applied if straggling is calculated with MDEPTH or WDEPTH (see section 12.1 below).

11.11 Non-Rutherford cross sections: ^4He on H

SigmaCalc 1.6 includes this cross section in the range 1-10 MeV, for any angle. We have used it to calculate it at different angles, mostly with 5° steps, and put this in a file called SigmaCalc_E4He1H.dat. If this file is in the WINDF\sigmacalc directory, it will be read and interpolations on angle will be used.

The user should use SigmaCalc to calculate for their exact angle, thus avoiding any interpolations, and then introducing the file as explained in Section 11.1.2.

If the file SigmaCalc_E4He1H.dat is not present, and if the user does not specify a given cross sections, then the default up until version 9.21 is used. This is an ad-hoc interpolation based in Figures 5.13-5.16 of the IBA Handbook and in ref. [25] (and references therein).

WARNING: Please note that this can NOT be used to extrapolate the cross section, only interpolate, and that these cross sections have been superceded by SigmaCalc.

11.12 Non-Rutherford cross sections: ^4He on D

The He on D ERDA cross-sections are non-Rutherford. They are given for 5 different values of the ϕ angle ("scattering angle" in section 5.1 above) in Figures 5.17-5.21 of the IBA Handbook. This was in NDF up to 9.2.m, and was wrong since version unknown. From 9.2.o it is correct again. Anyway, the file sigmacalc_E4He2H.dat was prepared by NP Barradas based on interpolated experimental cross sections (from Kellock and Quillet), that will work between 10° and 40° and up to 3 MeV.

12 Straggling calculations

Since version 8.1g, the default in NDF is Chu straggling, as detailed in section 12.4 below, using the Gaussian or gamma function for the straggling shape as detailed in section 12.5 below. There are several other options available to calculate the energy spread:

12.1 WDEPTH

This is the recommended option. Use it whenever you want to calculate energy spread.

WDEPTH is a code developed by Edit Szilágyi and co-workers [26], completely independent of NDF. It does state of the art depth resolution calculations, and for all we know there is nothing comparable available. NDF can call WDEPTH automatically during runtime, and use its output. During a fit, NDF will call WDEPTH several times, each time that it considers that the depth profile has changed enough to warrant recalculation of the energy resolution as a function of depth. You can obtain WDEPTH from Edit Szilágyi at szilagyi@rmki.kfki.hu (be sure to mention you need it to run with NDF), or Nuno Barradas (nunoni@itn.pt). Look also in the IBIS website (<http://www.kfki.hu/~ionhp/>)

WARNING: Whenever you publish results where WDEPTH was used, please cite [26].

You should be familiar with WDEPTH in order to understand what it does. Please read the literature and Edit Szilágyi's Manual.

WARNING: WDEPTH is provided "AS IS", WITHOUT WARRANTY.

To make NDF use WDEPTH (implemented for RBS, ERDA, NRA), add one line in the geometry file:

```
WDEPTH (or Wdepth or wdepth) wdepth_filename
```

To make NDF use WDEPTH only for some elements (and thus save calculation time), the line in the geometry file can be:

```
WDEPTH (or Wdepth or wdepth) wdepth_filename el1 el2 el3 ...
```

The file `wdepth_filename` contains data that WDEPTH needs that are otherwise not used by NDF. These are:

path where the WDEPTH directory is	e.g. c:\codes if it is in C:\codes\WDEPTH
Temperature in K	e.g. 300
foil thickness inhomogeneity (FWHM)	in micrometer
Q for reaction in MeV	0 for RBS and ERDA
energy level for reaction	0 for RBS and ERDA
shape of the beam spot (if previous line is C):	C if circular, R if rectangular
diameter of beam spot (if previous line is R):	in mm
width of beam spot	in mm
length (height) of beam spot	in mm
shape of detector slit (if previous line is C):	C if circular, R if rectangular
diameter of detector slit (if previous line is R):	in mm
width of detector slit	in mm
length (height) of detector slit	in mm
sample-detector distance	in mm
beam energy spread	in keV
angular divergence of beam	in degrees
type of straggling	B for Bohr, C for Bohr with Chu correction

One example is (for the ITN-Lisbon standard detector of the universal chamber with beam slits closed to 0.2 mm for grazing angle incidence):

```
d:\codes                                wdepth.exe is in d:\codes\WDEPTH
300
0                                         RBS, no foil
0                                         RBS, no reaction
0                                         RBS, no reaction
R                                         0.2×0.6 mm2 beam
0.2
0.6
```

```

R                               1.5×5.0 mm2 slit aperture
1.5
5.
70.                               7 cm between sample and detector
1.                               1 keV beam enery spread
0.02                             0.02° beam angular divergence
C                               Chu stragglng

```

You can also scale the influence of stragglng by a factor:

```
WDEPTHwdepth_filename el1 el2 el3 ... scaling_factor
```

Note that this is not a multiplicative scaling to the resolution values in the file `ndf_1.dat` which is output of WDEPTH (columns 2 & 3); it is a scaling in the increase in resolution values due to stragglng; for instance, the resolution at the surface will not be changed. Now, suppose you introduced `scaling_factor=0.5`, and at a given depth the FWHM in the original file is 30 keV, for a surface value of 15 keV. The contribution of stragglng so far has been $30^2-15^2=675$ keV², or 25.98 keV. This will now be $0.5 \times 25.98 = 12.99$ keV. The total resolution will then be $12.99^2+15^2=393.75$ keV², or 19.84 keV.

On top of that you can add a constant (positive or negative) to the values given in the file `ndf_1.dat`:

```
WDEPTH depth_filename scaling_factor add_constant
```

In the example above, with `scaling_factor=0.5`, if `add_constant = -1.5`, then the final resolution at that depth would be $19.84-1.5=18.34$ keV.

NDF writes intermediate input data for WDEPTH in the WDEPTH directory: these are `ndf_layer.lay`, `ndf_foil.lay`, `ndf.par` & `ndf.inp`. The output files of WDEPTH are: `ndf_1.dat` and `ndf_2.dat`. These are described in detail in the WDEPTH Manual of Edit Szilágyi, but the NDF user does not need to look at these: the results are read automatically by NDF.

WARNING: Note that the foil thickness inhomogeneity is FWHM and in micrometer. This is completely different to what can be input directly in the GEO file, which is the standard deviation in 10^{15} /cm² units. See section 5.12.2 above.

WARNING: If stragglng is to be calculated with DEPTH, the value for the foil inhomogeneity that is given in the GEO file takes precedence. Whatever is in the DEPTH input file (see section 5.12.2 above) will be ignored, unless no foil inhomogeneity is given in the GEO file.

WARNING: From 25Oct07 on NDF uses not the entire WDEPTH calculation, but discards results when the WDEPTH calculation error as given in the WDEPTH result file are 3% or larger (this is the default). NDF then makes a linear extrapolation from the previous two values. This is to try to avoid problems at grazing angle, where the MS scattering dominates and the MS theory is close to its limits of validity. If you want to change the default, the codeword in the GEO file is e.g.

```
dperror 1.
```

(this will set the error level at which the DEPTH calculation is disregarded to 1%)

```
dperror 1.e10
```

(this will effectively make NDF use the entire DEPTH calculation as this error is never reached!)

Note that when using the low energy yield option (see section 11.6 above) together with DEPTH, `dperror` becomes critical. Trial and error is the only advice.

12.2 MDEPTH

MDEPTH is the DOS code predecessor to WDEPTH. NDF can call MDEPTH automatically during runtime, and use its output. During a fit, NDF will call MDEPTH several times, each time that it considers that the depth profile has changed enough to warrant recalculation of the energy resolution as a function of depth. NDF uses a version of MDEPTH that accepts up to 50 layers, while the standard downloadable version accepts only 25. To use NDF with MDEPTH, you need the 50 layer version. You can obtain it from Edit Szilágyi at szilagyi@rmki.kfki.hu (be sure to mention you need it to run with NDF), or Nuno Barradas (nunoni@itn.pt). Look also in the IBIS website (<http://www.kfki.hu/~ionhp/>) First, you need to install MDEPTH in your computer, using the default directory structure, and it must be a top level directory, e.g. `c:\mdepth` or `d:\mdepth`. Otherwise, MDEPTH will of course work, but NDF will not be able to call it automatically.

WARNING: Whenever you publish results where DEPTH, MDEPTH or WDEPTH was used, please cite [26].

You should be familiar with MDEPTH in order to understand what it does. Please read the literature and the manual.

WARNING: MDEPTH is provided "AS IS", WITHOUT WARRANTY.

To make NDF use MDEPTH (implemented for RBS, ERDA, NRA), add one line in the geometry file:

MDEPTH (or Mdepth or mdepth) mdepth_filename

everything else is exactly the same as for WDEPTH

WARNING: From NDF 8.0j on, the first line of the MDEPTH input file can no longer be just the disk (e.g. C, or D); it now has to be e.g. C: or D:, or something such as C:\codes. That is, the path specified can have up to about 78 characters (but less is recommended), and NDF will append to it the rest of the path (MDEPTH) (which includes the initial back slash)

WARNING: From 25Oct07 on NDF uses not the entire WDEPTH calculation, but discards results when the WDEPTH calculation error as given in the WDEPTH result file are 3% or larger (this is the default). NDF then makes a linear interpolation from the previous two values. This is to try to avoid problems at grazing angle, where the MS scattering dominates and the MS theory is close to its limits of validity. If you want to change the default, the codword in the GEO file is e.g.

dperror 1.e10

(this will effectively make NDF use the entire DEPTH calculation as this error is never reached!)

12.3 DEPTH

It is also possible, to input a file containing energy resolution values.

WARNING: Whenever you publish results where DEPTH, MDEPTH or WDEPTH was used, please cite [26].

The format of this file is (a maximum of 4000 values are allowed):

six dummy lines, then four columns with
dummy dummy energy-resolution (keV) beam-energy-before-interaction (keV)

This is the output format of Edit Szilágyi's DEPTH code.

The energy resolution is different for each element in the sample (straggling and plural scattering will be different after scattering because of the different energy after scattering off different elements).

So, all you have to do is to calculate the energy resolution for the elements you have (using e.g. the nominal sample composition), by any method you like (using DEPTH is the best way), save the result as files, and then, in the geometry file (see section 5.1), add one line per element:

DEPTH (or depth or Depth) element depth_filename

You can also scale the influence of straggling by a factor:

DEPTH (or depth or Depth) element depth_filename scaling_factor

Note that this is not a multiplicative scaling to the resolution values in the file depth_filename; it is a scaling in the increase in resolution values due to straggling; for instance, the resolution at the surface will not be changed. Now, suppose you introduced scaling_factor=0.5, and at a given depth the FWHM in the original file is 30 keV, for a surface value of 15 keV. The contribution of straggling so far has been $30^2 - 15^2 = 675$ keV², or 25.98 keV. This will now be $0.5 \times 25.98 = 12.99$ keV. The total resolution will then be $12.99^2 + 15^2 = 393.75$ keV², or 19.84 keV.

On top of that you can add a constant (positive or negative) to the values given in the file depth_filename:

MDEPTH (or Mdepth or mdepth) mdepth_filename e11 e12 e13 ... scaling_factor add_constant

In the example above, with scaling_factor=0.5, if add_constant = -1.5, then the final resolution at that depth would be $19.84 - 1.5 = 18.34$ keV.

All elements for which no such line exists will be considered as having straggling given by the Chu or Bohr flag above, or no straggling in their absence.

WARNING: This will slow NDF down by 1-2 orders of magnitude. Expect long calculation times even if you have a good PC. Do not use it unless you need it.

12.4 Chu and Bohr straggling

The default in NDF, if the user does not say anything, is Chu straggling with effective charge scaling and Tschalär scaling in the propagation of straggling in thick films [27].

The user can tell NDF explicitly to calculate Chu straggling with effective charge scaling, or Bohr straggling. NDF v8 does not calculate the influence of other effects (geometrical dispersion, plural and multiple scattering, etc). This means

that any interface width will be overestimated by NDF. You should use MDEPTH for best results.

In all cases, Tschalär scaling is used. This affects the propagation of FWHM from layer to layer due to a changed stopping power: suppose you have `sigma_so_far`, and now are calculating the sigma after a new layer `i` where there is a `sig_i`. The total will *not* be $\sqrt{\text{sig_so_far}^2 + \text{sig_i}^2}$, but something like $\sqrt{(f \times \text{sig_so_far})^2 + \text{sig_i}^2}$, where $f = \text{stopping_end_of_layer_i} / \text{stopping_beginning_of_layer_i}$ - there will be an expansion or contraction of the previous sigma, because it is propagated through the stopping power. This has quite an effect in thick layers where you go from above to below the stopping maximum!

NDF also lets the user scale the value of (Chu or Bohr) straggling calculated, by introducing a multiplicative scaling factor. The user must add one line at the end of the geometry file (see section 5.1), which for Chu straggling is simply:

CHU (or Chu or chu) `scaling_factor` (1. for no scaling)

and for Bohr straggling is

BOHR (or Bohr or bohr) `scaling_factor` (1. for no scaling)

To turn straggling off, set the `scaling_factor` to 0.

WARNING: This will slow NDF down.

WARNING: In the current version, the straggling in any existing stopping foil is not calculated. Any way, Edit Szilágyi has shown multiple scattering to be dominant for grazing angle of incidence as in ERDA, so MDEPTH should always be used.

WARNING: When using Chu or Bohr straggling, the straggling is calculated at the beginning of each sub-Markov chain, i.e., only once for each temperature, because it is a very time-consuming procedure. This has the consequence that the χ^2 may increase occasionally, particularly at high temperatures (although it can also happen during local minimisation).

12.5 Shape of straggling function

Normally, the straggling function in NDF is taken to be a Gaussian. Close to the surface this leads to some particles actually gaining energy. The reason is that the Gaussian function has endless tails, non-zero for energies larger than the surface energy. To prevent this, NDF can use a Gamma function for the straggling. The Gamma function is close to the Gaussian for small standard deviation relative to the average value (it actually is a Gaussian for the standard deviation tending to zero), but it does not have a tail at values smaller than zero. The FWHM system resolution is still taken to be a Gaussian.

This is valid for RBS and ERDA, whenever the FWHM is smaller than three times the energy calibration gain. Otherwise, or for NRA and NDP, the default is still the Gaussian function.

If the user explicitly wishes to use a Gaussian for the straggling, a codeword at the end of the geo file must be included:
strag gauss

If the user explicitly wants to use a Gamma, or to use a Gamma function for NRA/NDP then it should be
strag gamma

12.6 Tschalär effect

By default NDF incorporates the Tschalär effect of propagation of straggling through thick layers; the effect is always calculated, even for thin layers. The user can turn it off (for test purposes only, the effect is real and should be left there), by writing in the GEO file:

tschalar off

Tschalar, TSCHALAR, Off and OFF also work.

13 Roughness, porosity, inclusions, density variation

13.1 Roughness: fast vs. slow calculations

There are many types of roughness. From small to moderate levels of roughness, its effect can be approximated as a broadening of any edges or peaks, that is, it can be treated as an extra contribution to energy spread. This leads to approximative models that are very fast, described in section 13.4 below. In this case, the effect of roughness is assumed to be Gaussian. For moderate to large levels of roughness, these models break down.

In general, there is not one single algorithm that can deal correctly with all types of roughness. NDF implements a) fast approximate models for low to moderate roughness; b) inhomogeneous layer thickness with a gamma distribution; c) user-input layer thickness distribution; d) and also implements Gurbich's model for re-entrant beams.

13.2 Gamma roughness - layer thickness inhomogeneity

In the case that the layer thickness changes from point to point, i.e. for inhomogeneous layer thickness, one can simply calculate one sub-spectrum for each possible value of the layer thickness, and add them all together, weighted by their probability. NDF, following Matej Mayer's SIMNRA, can use a gamma distribution for the roughness in this case, as described in Section 13.2. This can be very slow, particularly if several layers have roughness.

However, inhomogeneous layer thickness is the approximate fast model 1). So, what NDF does is decide which calculation model to use, the slow gamma distribution summation of many partial spectra, or the fast approximate Gaussian calculation. By default, for a roughness value up to 20% of the layer thickness, the fast model is used. Above that level of roughness, the slow summation method is used. The user can force one method or the other to be used, as described below.

In this model it is supposed that the contribution to the depth resolution caused by the roughness is independent from the other contributions, and can be added to them in quadrature.

The roughness is caused by the *inhomogeneity in the layers thickness*. See figure a) below. In this case, we calculate many sub-spectra corresponding to each possible layer thickness, that we assume follows a gamma distribution.

To make NDF calculate roughness, you need to introduce at the end of the str file:

```
rough
1 0 0
0 0
0 0
0 0
```

By default, this model is used for a roughness value above 20% of the layer thickness. Below that level of roughness, the fast approximate method described in section 13.4 below is used. To make this model be used even for lower levels of roughness, you need to introduce at the end of the str file:

```
rough gamma
1 0 0
0 0
0 0
0 0
```

On the contrary, to force the fast model, you can do:

```
rough gauss
1 0 0
0 0
0 0
0 0
```

Gamma roughness can be very slow to calculate, because NDF may decide to use very many points - hundreds, in the worst case scenario. You can limit the number of points by writing e.g.

```
rough gamma 10
```

in which case NDF will use up to about double the number given (10 in the example above).

13.3 User input layer thickness distribution

There are many circumstances when the thickness of a layer is known to vary widely; if there are pinholes in a layer or (conversely) if there are particles on a sample; or if a layer is known to have a certain structure. This can be calculated using an existing roughness model in NDF, with a gamma function implementation of the resulting straggling (see section 13.2). Note that these "gross roughness" models will work only approximately, tending to fail at larger incidence or exit angles.

Alternatively, users can input their own layer thickness distribution, independently for each layer. This is in conjunction with model roughness 1, so it will only work for simulation and for local search fitting, not for simulated annealing.

It is complicated, and what the user must do reflects the complications:

- The user can input directly the desired distribution, as thickness value/probability. The total probability does not need to be normalised, NDF takes care of that.
- The user can input probability segments defined by two thickness values with their respective unnormalised probabilities. Between the two points given, a linear variation of the thickness probability is assumed. For instance, if the segment is defined by the two points (1000,1) and (2000,2), at thickness 1500 the probability (before normalisation) is 1.5. NDF normalises and generates the necessary number of points for the calculation.
- The user can input delta functions. This is a thickness value with a probability value. In this case, the probability must be normalised by the user. If only delta functions are given, the total probability must add to 1.
- Segments and delta functions can be combined in the same input. The total probability for the segments is then whatever is left from the delta functions (and this is the reason why NDF does not normalise the delta probabilities). Distributions cannot be combined with segments or delta functions.
- There are three modes for the thickness values given. In the first, called "t" for "thickness", the thickness values in the user input file are directly what NDF will take. Thus, whatever is given in the PRF file for that layer is disregarded.
- In the second mode, called "r" for "raw", the values given in the user input file are directly added to the thickness given in PRF. Suppose this is 1000, and that the user input file has two delta functions at 100 and 500 thickness. Then, the calculation will be for final thickness 1000+100 and 1000+500.
- The third mode, called "n" for "normalised", works like "raw", but the user input values are normalised to an average of zero. The averaging is made by weighting by the probability of each thickness, so this mode does not change the average thickness given in PRF. In the previous example, if the probabilities of 100 and 500 were equal, the average was 300 and the normalised values become 100-300=-200 and 500-300=200. Then, the calculation will be for final thickness 1000-200 and 1000+200. But if the probability of 100 was 0.2 and of 500 0.8, the weighted average is 420, the normalised values become 100-420=-320 and 500-420=80. Then, the calculation will be for final thickness 1000-320 and 1000+80.
- NDF will translate the input into a series of thickness/probability points. **WARNING:** Note that a segment 0-100, for which 3 points are created, the probabilities will be (0,0.25) (50,0.5) (100,0.25), because the extremities (0 and 100) represent each a quarter of the segment. NDF will output the actually used points in a file bROUGHDISTsfl.DAT, where "l" stands for layer number, but since they are not equally spaced and each one does not represent the same thickness width, connecting them with a straight line is NOT a thickness distribution.
- Note that it is possible that modes r and n lead to negative thickness values (because in mode r, negative values of the values input are allowed). NDF forces such layers to have a very small thickness of 1e-3 TFU.

The user must write in the PRF file, instead of the roughness value, the filename. E.g. for a two layer structure with one logical element on a substrate, it could be, for thickness distribution fl.txt:

```
2
500 1 0 fl.txt 0
30000 0 1 0 0
```

The format of the file is

- first line defines several things:

the mode (t, r, or n) with a line such as

mode t

the units, which can be nm or 1e15at/cm2. Default is 1e15at/cm2. The line becomes e.g.

mode n units nm

xdiv and nmin, for fine control of the calculations for segments. The calculation interval will be divided by xdiv. So xdiv=1. has no effect on the calculation; xdiv<1 means wide segments will be divided in fewer points; xdiv>1. means more points will be calculated. However, each segment will have at least nmin points in the calculation. Suppose you have a distribution centered at a fairly shallow depth, with a long tail. The details in the shallow depth are important, the long tail not very much. You can set xdiv=0.3. and nmin=5, so the shallow segments will have at least 5 points, with the long tail not getting too many points that would slow down the calculation very much. The line becomes e.g.
mode r units 1e15at/cm2 xdiv 2.5 nmin 2

The order is fixed, but only mode is mandatory. Note that to have nmin you need all the previous ones as well, and to have xdiv you need units and nmin as well. To have units you only need mode and units.

- If the user wants to input a distribution, this is followed by two columns, thickness/unnormalised probability, e.g.:
distribution

100 .1

120 .2

...

200 .01

- If the user wants to input deltas, segments, or a combination thereof, for each delta a single thickness/probability is given, for a segment two thickness/probability pairs are given, e.g.:

delta

2000 .3

segment

1000 1

1500 0.1

In local search, NDF can fit the probability in the (thickness,probability) points derived from a delta/segment description, or directly given as a distribution. This is done in the usual way, with the proper flag 1 in the PRF file (or in NDFPRF.ORD, see section 7.5.7 above). For instance, using the PRF file, it would be e.g.

2

500 1 0 f1.txt 1

30000 0 1 0 0

WARNING: In version 9.5h, a maximum of 100 points in the distribution are supported.

13.4 Approximate fast roughness models

NDF has an algorithm to calculate the influence of roughness on the spectra. **WARNING:** It will only do so when the stragglng or DEPTH options (see section 12 above) are used. For instance, if you use DEPTH for one element but not for the others, and no stragglng option, the influence of roughness will be only calculated for that element.

NDF models several different kinds of roughness, described below and given in the figure. What it does is to try to describe the influence of the roughness solely as an increase in the energy resolution as a function of depth. This can be more or less successful as noted below. The roughness models are those given in ref. [28], and the description below follows that paper. A more complete description of the models is given in [29], and a discussion of their limits of validity is given in [30]. Please read these papers if you want to use the roughness option in NDF.

In all the models it was supposed that the contribution to the depth resolution caused by the surface roughness follows Gaussian distribution, is independent from the other contributions, and can be added to them in quadrature. The models are as follows:

1) The roughness is caused by the *inhomogeneity in the layers thickness*. See figure a) below. In this case, assuming that the previous layer does not affect the thickness of the following one (like filling up valleys), the contributions due to each layer, δX_i , are independent from each other and we can derive that the total contribution, ΔX^{inh} , to the depth resolution at the interface between the n^{th} layer and the following one is

$$\Delta X^{\text{inh}} = \sqrt{\sum_{i=1}^n \delta X_i^2}, \quad (14.1)$$

where δX_i is the inhomogeneity of the i^{th} layer (counting from the surface). NDF implements Eq. (14.1) in the following way: The δX_i for each layer, taken as the standard deviation of the layer thickness (not the FWHM) is given by the user. The corresponding energy resolution (standard deviation, not FWHM) is added to the energy resolution at the interface between layer i and the next one. In layer i itself, the energy resolution will be a changing function, starting at the value

at the bottom layer $i-1$. **WARNING:** Please note that there is no way to correctly implement this model 1 simply by a changing energy resolution. The implementation described is thus reasonable, but not perfect. In particular, it will fail when the roughness value is of the same order of magnitude of the layer thickness.

2) The variation in the layer thicknesses is insignificant, and instead the contribution is connected to the fact that *the substrate surface is slightly deformed* and consequently causes a variation $\delta\theta$ (standard deviation) in the angle of incidence, that leads to a spread δp in the length p of the path the particles make to probe a certain depth (see fig. b). That spread will be perceived as coming from a spread in the thickness of the layers. The resulting contribution at depth X , ΔX^{def} , can be calculated by differentiating $X=p\cos(\theta)$, and is

$$\Delta X^{\text{def}}(X,\theta) = X \operatorname{tg}(\theta) \delta\theta \quad (14.2)$$

NDF implements equation (14.2) by converting ΔX^{def} into an energy resolution value (standard deviation, not FWHM). This means it can have discontinuities from layer to layer due to different stopping values.

3) The contribution is still connected to *the surface roughness of the substrate*, but in a more sophisticated way. All the interfaces closely follow the substrate surface (see fig. c). Consequently, when the ions enter the surface in point A and are backscattered at a thickness X in a different point B, the contribution will appear only if there is a lateral distance l between A and B. The contribution, ΔX^{subst} , will increase with l until reaching a saturation value δX characterizing the substrate surface roughness. δX is actually the standard deviation (not FWHM) of the height distribution of the substrate surface points. The exact shape of this saturation behaviour depends on the shape characterizing the surface roughness (i.e., smooth waves, steps, sawtooth shaped elevations, etc.). Without detailed knowledge of it, we estimated the behavior to be error function shaped, resulting in

$$\Delta X^{\text{subst}}(X,\theta) = \delta X \operatorname{erf}(X \operatorname{tg}(\theta)/L), \quad (14.3)$$

where L is the correlation length, being of the order of magnitude of the lateral size of the surface structures. NDF implements equation (14.3) by converting ΔX^{subst} into an energy resolution value (standard deviation, not FWHM). This means it can have discontinuities from layer to layer due to different stopping values. Note also that for small L values, the resolution will change very rapidly, which will lead to strange spectral shapes. In particular, our model will fail when the roughness value is of the same order of magnitude of (larger than) the layer thickness. See also model (5).

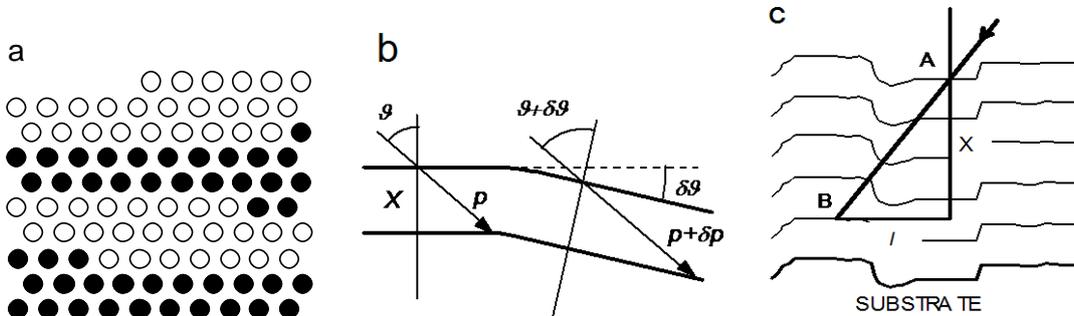


Fig. Diagrammatic explanation of the roughness models considered. a) Inhomogeneity in the layers thicknesses. A $(\text{Re } 0.5 \text{ nm}/\text{Co } 0.5 \text{ nm}) \times 3 \text{ SL}$ is schematically represented. Re and Co atoms are depicted by black and white circles respectively. The discrete nature of the layers leads to a variation in their thickness from point to point. b) Deformed substrate surface. The $\delta\theta$ incident angle variation leads to different paths inside each layer, that will be perceived as different layer thicknesses. c) Substrate surface roughness. The lateral distance between the impact point A of an ion to the surface and the point B where the backscattering takes place is $l = X \operatorname{tg}\theta$. The deviation in the observed layer thickness increases with increasing l .

(4) Roughness/intermixing in one (or more) interface. The contribution δX_i to the standard deviation will be introduced only in the interface(s) in question, and does not propagate to the deeper layers. This will be a localised broadening of the features associated with the interface(s) considered. The δX_i for each interface, taken as the standard deviation of the interface (not the FWHM) is given by the user. The corresponding energy resolution (standard deviation, not FWHM) is added to the energy resolution at the interface between layer i and the next one. **WARNING:** Please note that there is no way to correctly implement this model 4 simply by a changing energy resolution. Our implementation is reasonable, but not perfect. In particular, it will fail when the roughness value is of the same order of magnitude of the layer thickness.

(5) Intermixing in a multilayer. We assume a multilayer with equal layer intermixing in all interfaces. The intermixing is supposed to be equivalent to convoluting the real box profile with a Gaussian of width δX (in nm) (this results in a

error function shaped mixing). The contribution to the apparent depth profile is similar to that of model (3):

$$\Delta X^{\text{subst}}(X, \theta) = \delta X \operatorname{erf}(X/L), \quad (14.4)$$

but without the angular dependence, as saturation is reached at the first bilayer thickness. In this case, L should be input as the bilayer thickness, in nm.

13.4.1 Include roughness in a simulation or fit

To include roughness in a simulation or fit, you must do the following:

1) Introduce 5 lines at the end of the str structure file. They are:

Rough (or rough or ROUGH)

F1 F2 F3: these are flags to define if models 1, 2 or 3 are used. 0: not, 1: yes. More than one model can be used simultaneously. To use model (4), put F1=4. Models (1) and (4) cannot be used simultaneously. To use model (5), put F3=5. Models (3) and (5) cannot be used simultaneously

$\delta\theta$, flag_var: initial value for standard deviation $\delta\theta$ in Eq. (14.2) (in degrees), varies during fit (1) or not (0)

δX , flag_var: initial value for standard deviation δX in Eq. (14.3) (in nm), varies during fit (1) or not (0)

L, flag_var: initial value for L in Eq. (14.3) (in nm), varies during fit (1) or not (0)

2) If you want to use model (1), you must introduce two new columns in file NDF.PRF, after the element concentrations (see the description of NDF.PRF in section 7.1 above). The first one contains the initial roughness standard deviation value δX_i (from eq. 14.1) of that layer, in $1e15$ at/cm², the second is a flag: 1 if the roughness of that layer varies during the fit, 0 if not. If you want to use model (4), you must introduce similar columns in file NDF.PRF, but the first one contains the mixing value δX_i between layers i and $i+1$, in $1e15$ at/cm².

WARNING: from NDF version 9.5a, NDFPRF.ORD takes precedence over the fit control process and overrides the 0/1 flags in the PRF file. See section 7.5.7 above. If you want the PRF file to retain precedence for the control or roughness or void fraction, after the depth profile it must contain one line with the word "rough".

WARNING: If there is no existing NDFPRF.ORD and a simulation or a local search are done, NDF or WiNDF will create one.

For model 1, by default, for a roughness value up to 20% of the layer thickness, this fast model is used. Above that level of roughness, the slow summation method is used. The user can force the gaussian fast model to be always used:

Rough gauss

1 F2 F3

$\delta\theta$, flag_var

δX , flag_var)

L, flag_var:

And the user can also force the slow summation model:

rough gamma

1 F2 F3

$\delta\theta$, flag_var

δX , flag_var

WARNING: The current implementation for models (1) and (4) allows, in each layer, a roughness up to a maximum of 50% of the layer thickness, and it is recommended to use it only up to 20%. Larger values lead to increasing inaccuracy in the results. For model (1), NDF will automatically revert to explicit sum of structures with gamma distribution function when roughness is above 20%.

WARNING: The current implementation is prepared to use models (1) and (4) with a local search only. SimAnn (even local SimAnn - see section 7.8.2 above) will produce undetermined results: the calculations will be right, but in which layers will have roughness been introduced is pretty much random. In principle, there should be no restrictions for models (2) and (3), but I am not quite certain about that: never tested. As a general rule, if you are including roughness in the fit then you should be certain of what the layer structure is. That is, using roughness ONLY with local search is a good thing to do. Sorry for the inconvenience. **WARNING:** Use of roughness is incompatible with determination of stopping powers (see section 15 below) or non-Rutherford cross sections (see section 16 below).

13.5 Voids and inclusions - approximate fast model

WARNING: The limits of validity of these models aren't tested yet. We have tested them on different systems (Ge quantum dots in Si; GaN quantum dots in AlN; etc) and the results were good. However, just like in the roughness models, artefacts may appear in some cases: in large quantum dots with stopping very different from the matrix, and located in just a very thin layer (as opposed to distributed in the whole matrix), the calculated signal of the matrix at the depth of the ultrathin layer may present a sudden unphysical oscillation; this is due to a sudden large increase in the apparent resolution value.

13.5.1 Spherical

NDF implements the algorithm given in [31] to calculate the influence of spherical voids and inclusions in the spectra. **WARNING:** Read and understand the paper before you use this. **WARNING:** Can be used only in simulations or local searches.

You must specify the void radius R (in nm) and the order parameter K (read the paper: K=1 for randomly distributed voids, K=1/8 for a fully ordered/regular distribution of the voids). Only one type of voids is allowed.

In the STR file, you must increase the number of elements by 1, and add the voids as the last element, in the following way:

```
sphere (or SPHERE or Sphere)
0          (nsphere)
R K       (radius in nm of the voids and K factor)
0 1000    (depth constraint)
0.1 0.4   (concentration constraint)
```

where the last two lines are the usual constraints (0 0 for no limitation).

If nsphere=0, then this is empty voids (pores).

Otherwise, nsphere is the molecule number (by the order given in the STR file) that fills the inclusions. Please note that you can have one single molecule in the inclusion.

Then, the PRF file needs two extra columns, with the volume density of the voids (1 meaning 100% voids), and a control flag 0/1 (1 if the void volume fraction changes during a local search, 0 if not). In case there is roughness types 1 or 4 present, the corresponding columns appear after the volume fraction columns.

13.5.2 Cylindrical

We developed the theory for cylindrical voids and inclusions as well. It works exactly like for spherical ones, but besides the radius of the cylinders (base is assumed parallel with the surface), their height is also required.

```
cylinder (or CYLINDER or Cylinder)
1          (nsphere)
R H K     (radius and height in nm of the voids and K factor)
0 1000    (depth constraint)
0.1 0.4   (concentration constraint)
```

13.5.3 The PRF file

As said above, two extra columns are needed, which is the volume fraction of the voids or inclusions, from 0 to 1, and whether they change or not (0 if not for that layer, 1 if yes).

WARNING: from NDF version 9.5a, NDFPRF.ORF takes precedence over the fit control process and overrides the 0/1 flags in the PRF file. See section 7.5.7 above. If you want the PRF file to retain precedence for the control or roughness or void fraction, after the depth profile it must contain one line with the word "rough".

Some subtle points need to be considered, in case it is inclusions, regarding the inclusion molecule (given as nsphere):

1. If the volume fraction is larger than zero, whatever is given in PRF as concentration for the inclusion molecule is completely ignored. NDF will calculate the correct concentration taking into account the given volume fraction and density of the molecule. In this case, the exact concentration will depend on the density of the matrix and of the inclusions. For the same volume fraction, the inclusion concentration will be different for a different matrix density.

2. On simulating a structure given in NDF.PRF where the nsphere (or ncylinder) element concentration is given as "0" but the volume fraction of nsphere is non-zero, NDF will not output the identical NDF.PRF (normalised): instead it will output the actual concentration of nsphere (calculated according to [1] above), with the other concentrations maintaining their relative ratios (normalised as usual). This change is somewhat disconcerting at first sight, but it is the

only consistent way to proceed. Please note that the calculated values depend on the values for density (g/cc) used by NDF. This (and in some roughness models) is the only place where NDF does not use natural units of atoms/cm².

3. If the volume fraction is zero, then the inclusion molecule will be treated as a normal molecule, and whatever is given in PRF as concentration for the inclusion molecule is used by NDF.

4. We emphasise that if there are inclusions (with the volume fraction >0) then NDF treats the nsphere (or ncylinder) logical element as being entirely in the inclusion. To treat the case where a distribution of a logical element only partially condenses into inclusions you have to use two different logical elements, one entirely in the inclusions and the other in the matrix.

If there is also roughness present (models 1 or 4), then the volume fraction must come in the column before the roughness value of the same layer.

13.5.4 Examples

For instance, suppose you have a thin layer of porous graphite (with 10 nm radius voids randomly occupying 20% of the volume) on Si. The STR file can be:

```
100
3
C
0 0
0 0
Si
0 0
0 0
sphere
0
10 1
0 0
0 0
```

And the PRF file can be:

```
2
1000 1 0 0.2
30000 0 1 0.
```

Now suppose you have a Si sample with one buried layer made of GaN with Er 3 nm radius spherical inclusions randomly distributed and occupying 0.1% of the vlume. The STR file can be:

```
100
4
GaN
0 0
0 0
Er
0 0
0 0
Si
0 0
0 0
sphere
2
3 1
0 0
0 0
```

And the PRF file can be (a constant volume fraction will be taken):

```
3
2500 0 0 1 0 0
1000 1 0 0 0.001 0
30000 0 0 1 0 0
```

13.5.5 Energy spreading and self-organisation

N.P. Barradas disagrees with J.-P. Stoquer about the effect of self-organisation, as in Appendix B of ref. [31]. Except for specific angles of incidence, related to the kind of order, on average after crossing enough layers there will be an averaging effect. In this case, $K \approx 1$. In any case, some ions will go through the inclusions and some will not, leading always to energy spread due to the number of inclusions traversed, and not only the pathlength inside them. Depending on angle, the effect can even be that some ions will be exclusively in the matrix, and others most of the time in the inclusions, in which case a code that implements only energy spread (such as NDF) will fail. N.P. Barradas advises to use $K=1$ in almost all cases, except for extremely high density and degree of order in angle of incidence/exit aligned with the structures. The users should however make their own minds, and not use this capability of NDF unless they read and understand Stoquer's paper.

13.6 Local variations of density

The density of a material can change from point to point, leading to a distribution of density values around its macroscopic average d , with standard deviation σ_d . This leads to an extra contribution to the energy spread of the beam, as calculated by Mistuo Tosaki [32]. We have now included this model in NDF. To introduce it, you must make $F1=6$ (see section 13.4.1 above), and then introduce the normalised standard deviation, that is σ_d/d for each layer in the layer structure as a new column in NDF.PRF (and a second column with 0 or 1), in a similar way as for models 1 and 4 (see section 13.4.1 above).

This effect can not be introduced together with roughness models 1 or 4. It can be introduced together with roughness models 2 and 3 (or 5).

13.7 Extra pathlength due to secondary crossings

Gurbich et al. presented an algorithm to deal with reentrant beams [33]. This leads to an extra pathlength, and hence to extra energy loss, leading to changes in the spectrum that can be very severe. All algorithms presented above in this section ignore this effect. Valentina Paneta et al. implemented the algorithm in a stand alone code, that needs as input the actual spectrum and other information (such as stopping powers). This is open source code. We now linked this implementation in NDF.

NDF needs the parameters σ and d from ref. [33] (in NDF, they are given in nm). These are respectively the standard deviation of the height of the surface (assumed Gaussian distributed) and the separation d between features. NDF needs further parameters, namely, to know which layers in the layered description are included in the surface asperities. This is to calculate the stopping power in the asperities. NDF will use the average composition in the layers included, weighted by the respective thicknesses.

WARNING: NDF makes several extra assumptions not present in the original model. In particular, the original model was developed for monolayered samples, measured in normal incidence. The current implementation is general, but several other approximations were made.

This surface roughness effect can be used in conjunction with the other roughness models described above.

To include this in NDF, at the end of the STR file, after the codeword roughness, the user must include (sigma and d are given in nm):

```
surface
sigma sigma_var
d d_var
layer1 layer2
```

where sigma_var and d_var are 1 if sigma and d can change during a fit, 0 otherwise. layer1 and layer2 are integers that correspond to the first and last layer to consider as part of the surface asperities. layer1 should always be set to 1, the surfacemost layer! sigma and d are given in nm.

13.7.1 Correlations between models

In principle, there should be correlations between parameters in different models.:

- The standard deviation of the height of the surface in GR model (from "Gurbich Roughness") should be the same as the surface roughness that is given in models (1) and (3) above (see section 13.4).
- The separation between features in GR and correlation length in model 3 should be related.
- The layers used for stopping in GR should be coincident with GR asperity height taking into account individual layer thickness values.

However, all these models or their implementations involve approximations or simplifications, and reality is more complex; furthermore, to translate from some parameters to the others, material density must be accurately known, which is not always the case. So parameters from one model are independent from parameters in other models, and treated completely separately. It is up to the user to make sure what they are doing makes any sense.

14 Calculation of error bars

14.1 Calculate error bars on your depth profiles using Bayesian inference

This Bayesian inference algorithm allows you to calculate confidence limits on the depth profiles you obtain. It is a Monte Carlo calculation, which means it takes long to calculate.

Basically, MCMC runs after SimAnn/LocalSearch (although the calculations are actually completely separate). It will run if and only if you have a file called MCMC.BI in the current directory. This file must consist solely of four integers. The first one will be the length of the Markov Chain calculated. The number of elements in the Markov required to obtain a statistically significant result depends from problem to problem, you must satisfy yourself. No diagnostic tools are provided with this version of NDF to tell you which length is OK.

14.2 Output files

You can get two kinds of output. The first one is the average depth profile plus the confidence limits, in a 4-column file: depth, <profile>, <profile>-sigma, <profile>+sigma

The second kind of output is the average thickness and composition of each layer, with the respective errors (sigma): thickness, sigma_thickness, element1(at%), element1_sigma, element2(at%), element2_sigma, etc.

To choose the first kind of output (default), the MCMC.BI file should be e.g. 1000 1 0 1, to choose the second it should be 1000 2 0 1 (see section 14.4 below for an explanation of the last two numbers). The second number, 1 or 2, is the relevant one. A 0 there will mean no MCMC on sample profile/ composition is made.

The files BURNCs.DAT, BURNEs.DAT, and MAINCs.DAT, MAINEs.DAT, are also created.

The *Cs.dat files are:

number of accepted state; chisquared

The *Es.dat files are:

number of accepted state; (calibration_factor, calibration_offset, charge×solid_angle_product)×number_of_geoms

14.3 Error bars on roughness parameters

If you are including roughness in the calculations (see section 13 above), then NDF will automatically do Bayesian inference on them, that is, you will get in the res files the average and standard deviation of the relevant roughness parameters (note that the average is in general not equal to the best fitted value). You will also get the normal output files as described above. **WARNING:** However, note that when using roughness, the number of layers remains constant, that is, the transition function never splits a layer into two or merges two layers into one. Hence it is slightly less flexible than the transition function used when roughness is not included. This should not be a serious problem, particularly taking into account that you should only use roughness when you are very certain of what the layer structure of your system is. **WARNING:** Use of roughness is incompatible with determination of stopping powers (see section 15 below) or non-Rutherford cross sections (see section 16 below).

14.4 Model limitations

WARNING: MCMC relies heavily on the chisquared (the one that is the distance between data and fit weighted by the error bar). If the chisquared is too large, e.g. because the model you are using is bad (e.g. you forgot your silicon wafer had Si and you're trying to fit it with O and Ge only), or because you need a substrate plural scattering correction, or whatever reason, MCMC will give the confidence limits GIVEN THE MODEL YOU USE! This might be different from the real errors given a correct model. Satisfy yourself you're doing the right thing.

There are some ways of getting around this. The first one is terribly fake, and it is to consider that the best fit obtained with SimAnn actually represents the real structure of the sample, and that all misfits are due to other problems (e.g. multiple scattering etc). Then we can take the fit, put Poisson noise on it, and analyse that as if it were the data. You do that using the third number in the MCMC.BI file, which should be a 1 to turn this option on, e.g. 1000 1 1 0 or 1000 2 1 0. The other way is to consider that the error for the yield in a given channel is not necessarily from statistics (in which case it would be Poisson), but that or the misfit; whichever is greater. To take just Poisson noise (which is the default in NDF) the fourth number should be a 1: 1000 1 0 1, 1000 2 0 1, 1000 1 1 1, 1000 2 1 1.

Finally, the user can assign equal errors to all channels. In this case the error will be 2; and this is done by putting a 2 in the third number of MCMC.BI, e.g. 1000 1 0 2.

The recommended usage is to do things properly, i.e. use the real data, and Poisson statistics: 1000 0 0 1, 1000 1 0 1 or 1000 2 0 1. Nothing else is justifiable by a purely statistics reasoning.

14.5 Use the NDF.TCN file

BI/MCMC will include in the error structure the error with which you determine the energy calibration and the charge-solid angle product. If you don't use custom cooling in the initial SimAnn run, these errors will be internally determined – they are actually different for the different pre-set cooling speed options, and I'm not telling which values are pre-set. This means you will get errors that will be partially unrelated to your real experiment. The only way you can control this properly is by using custom cooling, that is, the NDF.TCN file (see section 9 above). **MAKE SURE YOU USE IT.** This is on purpose: MCMC is trickier than SimAnn. Either you know what you're doing (and hence you won't have a problem working with NDF.TCN) or don't use MCMC at all. The errors MCMC will use for the energy calibration, sensitivity, and plural scattering correction are the variation limits given in NDF.TCN.

WARNING: The MCMC algorithm we implemented is not terribly stable. We did not use the maximum entropy prior (for very good reasons), which would have made the solutions more stable. Great care is required when using our algorithm, satisfy yourself!

14.6 Output files

MCMCse.DAT

This is the output from the "depth profile search" BI/MCMC calculation. It's one file per element per sample. It has 4 columns:

$$\text{depth (10}^{15}\text{/cm}^2\text{), } \langle e \rangle, \langle e \rangle - \sigma_e, \langle e \rangle + \sigma_e$$

where $\langle e \rangle$ is the mean value found for the concentration of e and σ_e is the standard deviation. Please notice that the depth in nm is not given.

MCMCs.DAT

This is the output from the "thickness and composition search" BI/MCMC calculation. It's one file per sample. It is columns:

$$\langle t \rangle, \sigma_t, \langle e \rangle, \sigma_e \text{ (1}^{\text{st}} \text{ element), ...}$$

where $\langle t \rangle$ is the mean value found for the thickness of one layer and σ_t is the corresponding standard deviation, $\langle e \rangle$ is the mean value found for the concentration of element e and σ_e is the standard deviation. Please notice that the depth in nm is not given.

14.7 Other parameters

Other parameters also change during the BI procedure. Concentration formula parameters (see section 4.4 above) that are allowed to change during the fit will change. The average and standard deviation are reported in bs.RES.

The energy calibration and charge can also change during the fit (within the limits specified by the user in NDF.TCN, see section 9). The average and standard deviation are not report in version 9.6a. the file MAINEs00.DAT contains the individual energy calibration parameters and charge for all geometries, for each accepted state. The user can calculate statistics based on this file. The structure is:

i (gain offset charge) (gain offset charge) ... (gain offset charge)

where the gain, offset and charge are given successively for each geometry. If the energy calibration is quadratic, then it is

i (quad gain offset charge) (quad gain offset charge) ... (quad gain offset charge)

WARNING: MAINEs00.DAT contains the first 50000 accepted states. For BI runs with more, files MAINEs01.DAT etc will each contain 50000 more.

The beam energy and the angles can also change during BI within their user specified range (see sections 5.5, 5.8). In NDF 9.6a this is not reported anywhere.

15 Determination of stopping power values

WARNING: Use of roughness (see section 13) is (possibly) incompatible with determination of stopping powers.

15.1 How it works

First, read section 10.6 on how to introduce user-input stopping powers. This is going to work in a similar way, but with some important differences. First, the codeword is molfit or elfit:

```
elfit 4He Si si.stp
```

or

```
molfit 4He Si1O2 si.stp
```

Second, the file must have the a1-a8 ZBL85 parameters. It is these parameters that are actually fitted. The values given in the file are taken as initial guess.

Third, this only works for RBS.

Finally, all the geometries with the same incident ion are affected. That is, if you fit the stopping power of 4He in Si, and you measured 5 spectra at different beam energies, you only need to include the codeword in one of the geometries. The same stopping power curve, based on the a1-a8 given and fitted, will be used for all the geometries.

15.2 Simple fitting

With the codewords elfit or molfit, the stopping power is fitted. the final parameters are given in the RES file and the final stopping power (which includes the nuclear stopping) is given in an output file (see below for the file name).

15.3 Bayesian inference

NDF has implemented a simple method for the determination of stopping powers. It involves measuring the energy spectra of backscattered particles from a known target (i.e., RBS spectra), which is experimentally trivial. We then simulate a theoretical spectrum for the known experimental parameters, and fit the stopping powers to obtain a match between the simulated spectrum and the data. This is done with a Bayesian inference algorithm, in order to get error bars on the results obtained.

You must set up the MCMC.BI file as in section 14 above, and introduce in the GEO file e.g.

```
elfit 4He Si si.stp
```

or

```
molfit 4He Si1O2 si.stp
```

Note: you can also do e.g.

```
molfit 4He Si si.stp
```

but in this case Si will be treated as a molecule, not as an element. This means that, if you have for instance defined a molecule including Si, such as FeSi₂, the stopping power of Si in this molecule will not be changed. Only the stopping power of the logical element "Si" as so given in the STR file is affected. For instance, if you also define one logical element as "si" (not capitalised!) it is not affected: for molecules, NDF is extremely literal. For elements, on the contrary, e.g. si, Si, SiO₂, sio₂, all are affected.

There are many details that you must take care of. In particular, this algorithm has only been tested by us with the recommended usage below. **WARNING:** Use of roughness is (probably) incompatible with determination of stopping powers (I think).

15.4 Restrictions and usage

Single element/molecule

The stopping power of one single target element or molecule can be studied (you can not run the routine on two elements at the same time).

Single ion

We have only tested the algorithm for a single incident ion. In principle, when using multiple spectra, it might be

advantageous (for accuracy) to use several different beams (e.g. H, He, and heavy ions) in the fit. The code does not handle this.

RBS only

The algorithm only works for RBS.

Charge state

The algorithm will ignore any user input charge state, it will use the calculated charge state in ZBL.

ROI and multiple/plural scattering

Multiple and plural scattering change the shape of the spectra. As the stopping power determination depends on the spectra shape, you cannot use data where multiple and plural scattering are not reliably calculated. You also cannot use the ad-hoc PS/MS corrections given in sections 11.7 and 11.8 above. This means you must use a ROI corresponding to a region where there is no PS/MS, or where you are confident that you calculate it correctly. This is typically the surface channels. The usable ROI will be larger for higher beam energies because MS/PS are higher at low energies. Satisfy yourself that the ROI you are using is adequate.

Multiple spectra

For better precision you should use multiple spectra, measured with different beam energies. This circumvents the problem of not being able to use the whole data but just surface ROIs, as it means a larger energy range will be covered.

Parameterisation

In this work we use the pre-1995 ZBL parameterisation for the electronic stopping, which has a functional form based on 8 parameters to be fitted. It is these parameters that we fit.

Pre-BI/MCMC fitting

The BI/MCMC process starts after an initial fit. We recommend that you first have a good fit to the data, including correct charge, calibration and other experimental parameters, then do a local search. Don't run SA beforehand.

Use the TCN file

You should use the TCN file to control the variation range of the experimental parameters, as mentioned in section 14.4. The model limitations mentioned in that section also apply.

15.5 Output files

An output file with the stopping curve is created. Its name will be indicated in the RES file. The structure is energy; original; [S], [S]- σ , [S]+ σ .

Where "original" is the curve normally used in NDF, [S] is the average stopping determined, and σ is the standard deviation determined. The meaning of s and f is given in section 8 above.

The best local search fitted stopping power curve will also be given as output:

STOPsG.DAT, containing
energy; [S]_best_fitted_local_search

The files BURNCs.DAT, BURNEs.DAT, burnttest.dat, BURNHs.DAT and MAINCs.DAT, MAINEs.DAT, maintest.dat, MAINHs.DAT are also created.

The *Cs.dat files are:

number of accepted state; 8_stopping_coefficients; chisquared

The *Es.dat files are:

number of accepted state; (calibration_factor, calibration_offset, charge \times solid_angle_product) \times number_of_geoms

The *test.dat files are:

number of tested state; 8_stopping_coefficients; chisquared

The *Hs.dat files are:

number of accepted state; surface height for all spectras fitted (counts per μ C per msr per keV)

16 Determination of non-Rutherford cross section values

WARNING: Use of roughness (see section 13) may be incompatible with determination of cross sections - we never tried.

16.1 How it works

NDF has implemented a simple method for the determination of non-Rutherford cross section values. It involves measuring the energy spectra of backscattered particles from a known target (i.e., RBS spectra), which is experimentally trivial. We then simulate a theoretical spectrum for the known experimental parameters, and fit the non-Rutherford cross section values to obtain a match between the simulated spectrum and the data. This is done with a Bayesian inference algorithm, in order to get error bars on the results obtained.

The BI algorithm is an advanced one that uses an algorithm to dramatically improve the transition function used. Without this algorithm (in versions before 8.1f) the code was unable to find even reasonable estimates of cross-sections with many resonances (like $^{28}\text{Si}(\alpha,\alpha)^{28}\text{Si}$).

What you must do: at the end of the first GEO file introduce one line with the flag "CROSS" (or "cross" or "Cross" or "cros") and then the element name for which you want to determine the stopping powers, e.g.

```
cros Al
```

to determine the non-Rutherford cross section of Al. It is necessary to enter the CROS keyword in only one GEO file.

You may want to limit the energy range that will be analysed/fitted. In this case, you should write:

```
cros Al emin emax
```

where emin and emax are the minimum and maximum energy to be analysed, in keV.

There are many details that you must take care of. In particular, this algorithm has only been tested by us with the recommended usage below, and is not particularly stable. **WARNING:** Use of roughness may be incompatible with determination of non-Rutherford cross sections.

16.2 Restrictions and usage

Non-Rutherford cross section file

You must input a non-Rutherford cross section file as described in section 11.1 above. The fit will be done on the data given in that original file. That is, the energy spacing between points will be that in the original file.

WARNING: if the spacing is tight, with sharp resonances, you should use the AUTOLAYER option described in section 11.1.9 above.

Single element

The non-Rutherford cross section of one single target element can be studied (you can not run the routine on two elements at the same time). In principle, you can at the same time run a stopping power search (even on a different element), but this is untested. The target may have more than one element.

Single ion

The algorithm is written for a single incident ion. When you use multiple geometries, in all of them the same non-Rutherford cross section file must be given. What happens if you do otherwise is undetermined.

Straggling

The shape of the data will depend on the shape of the resonances and on the shape distortion due to straggling. You should have a good straggling description, for instance by using DEPTH calculations as described in section 12.3 above. In some cases just Bohr or Chu straggling can suffice. This could be the case if the cross section changes only slowly.

Plural and multiple scattering

The considerations made in section 0 above also apply.

Multiple spectra

To avoid over-sensitivity to statistical fluctuations in the data, you should fit more than one spectrum at the same time. This should probably also be at different beam energies.

Use the TCN file

You should use the TCN file to control the variation range of the experimental parameters, as mentioned in section 14.4.

The model limitations mentioned in that section also apply.

16.3 Output files

The following output files are created:

CROSSf.DAT

Has the cross section curve. The structure is
energy; original; c, $c-\sigma$, $c+\sigma$.

Where "original" is the original file values, c is the correction factor determined, and σ is the standard deviation determined. The meaning of s and f is given in section 8 above.

The files BURNCs.DAT, BURNEs.DAT, burnttest.dat, and MAINCs.DAT, MAINEs.DAT, maintest.dat, are also created.

The *Cs.dat files are:

number of accepted state; chisquared

The *Es.dat files are:

number of accepted state; (calibration_factor, calibration_offset, charge×solid_angle_product)×number_of_geoms

The*test.dat files are:

number of tested state; chisquared

17 Trading-off accuracy and speed

The following options lead to higher accuracy at the cost of calculation speed:

- Straggling on or off (see section 12 above)
- Gaussian or gamma shape for the straggling (see section 12 above)
- Roughness and inclusions (see section 13 above)
- Double scattering (see section 11.4)
- Automatic division of layers when non-Rutherford cross sections are used (see section 11.1.9)
- Effect of energy spread before interaction (see section 0)
- Improved low energy yield calculation (see section 0 above)
- Wielopolski and Gardner algorithm for pile-up correction (see section 11.5.1 above)

NDF will assume several of these as default, even in cases where they don't lead to appreciable improvements. This is the case, for instance, when automatic division of layers is imposed as default when non-Rutherford cross sections are used, which don't have thin resonances. NDF can then become 1 order of magnitude slower without major gains. The user should then turn off the corresponding option in the geo file.

The following defaults apply, and should be changed by the user if necessary:

- Straggling is Chu with factor 1 unless otherwise specified
- Straggling function shape is gamma if FWHM is comparable to the energy width of a channel. Gamma is slower.
- Introduction of non-Rutherford cross sections leads to the calculation of automatic division of layers (autol option becomes on) and of the effect of energy spread before interaction (allres option becomes on for the element in question).

18 The IAEA intercomparison of IBA software: NDF

The International Atomic Energy Agency sponsored first a review of existing codes for IBA data analysis, and then an intercomparison exercise of the different codes [34,35]. All the results for all the codes are given in the dedicated web site <http://www.mfa.kfki.hu/sigmabase/ibasoft/>

We prepared the files required to run the NDF calculations. Please note that the results obtained with current NDF versions, even for the same exact calculation options (and note that the defaults have changed so the same calculation option may require different user options!) will not be numerically equal to what is in the web site - but they will be equivalent, with either no change, minor changes due to improved algorithms, or major changes due to improved physics.

All the files can be found in the NDF web page:
http://www.itn.pt/facilities/lfi/ndf/uk_lfi_ndf.htm

They should be run with the command line

```
ndf iaea p 0 1 1 0 0 p
```

which means “pure simulation” (it only affects the pile-up calculation), no compression.

Note that you must have MDEPTH, WDEPTH, and SRIM03 installed in your computer as detailed in this manual.

Calculations 11 and 28 were not made by NDF in the intercomparison exercise. Calculation 28 is made, but only the random spectrum is calculated.

Calculations 17 and 22 make no sense, since they include geometrical straggling but no multiple scattering. They require (in NDF) manually manipulating the WDEPTH output files to remove the contribution of multiple scattering, then inputting the resulting depth resolution files with the DEPTH option - this is not recommended usage, and therefore these calculations are not included here.

WARNING: Since June-July 2008, the handling of cross sections changed completely in NDF. It is now as accurate as feasible. This led to small changes in the results for the sharp resonance cases.

References

- [1] W. Möller and F. Besenbacher, NIM 168 (1980) 111.
- [2] M.A. Reis, L.C. Alves, NIMB68 (1992) 300.
- [3] M.A. Reis, L.C. Alves, A.P. Jesus, NIMB109/110 (1996) 134
- [4] B. Vekemans, K. Jensens, L. Vincze, F. Adams, P. Van Espen, X-Ray Spectrom. 23 (1994) 278. AXIL can be downloaded from the IAEA: <http://www.iaea.org/OurWork/ST/NA/NAAL/pci/ins/xrf/pciXRFdown.php>
- [5] W.N. Lennard et al. NIMA248 (1986) 454
- [6] Improved ion beam analysis facilities at the University of Surrey; C. Jeynes, N. P. Barradas, M. J. Blewett, and R. P. Webb, Nucl. Instrum. Methods Phys. Res. B136-138 (1998) 1229
- [7] P. Børgesen, R. Behrisch, and B.M.U. Scherzer, Appl. Phys. A27 (1982) 183.
- [8] QA/QC-“Intelligent” smooth and differentiation routine for RBS/IBA, N.P. Barradas, Internal Report 9/12/1997, Dept. Elec. Eng., University of Surrey, unpublished.
- [9] J. F. Ziegler, J. P. Biersack, U. Littmark/Stopping and Ranges of Ions in Solids (Pergamon, New York, 1985).
- [10] H. Paul, A. Schinner, NIMB179 (2001) 299.
- [11] M. Tosaki, S. Ito, N. Maeda, NIMB168 (2000) 543.
- [12] Large-angle scattering of light ions in the weakly screened Rutherford region, H.H. Andersen, F. Resenbacher, W. Möller, Phys Rev A (1980) 1891
- [13] J. L’Ecuyer, J.A. Davies, N. Matsunami, NIM 160 (1979) 337.
- [14] Double scattering in grazing angle Rutherford backscattering spectra; N.P. Barradas, Nucl. Instrum. Methods Phys. Res. B 225 (2004) 318.
- [15] Double scattering in RBS analysis of PtSi thin films on Si, N. P. Barradas, C. Pascual-Izarra, Nucl. Instrum. Methods Phys. Res. B (Proc. COSIRES 2004).
- [16] L. Varga, Nucl. Instrum. Methods B17 (1986) 260.
- [17] S. I. Bazhukov, A. V. Kibardin, T. M. Pyatkova, Nucl. Instrum. Methods B58 (1991) 242.
- [18] A. F. Gurbich, Nucl. Instrum. Methods A364 (1995) 496.
- [19] L. Wielopolski, R.P. Gardner , NIM133 (1976) 303.
- [20] L. Wielopolski, R.P. Gardner , NIM133 (1976) 303.
- [21] Radiation detection and measurement, 2nd edition, Glenn F. Knoll, Wiley New York 1989.
- [22] Accurate RBS measurements of the indium content of InGaAs thin films, C. Jeynes, Z.H. Jafri, R.P. Webb, A.C. Kimber, and M.J. Ashwin, Surface and Interface Analysis 25 (1997) 254.
- [23] G. Amsel, E. Girard, G. Vizkelethy, G. Battistig, Y. Girard, E. Szilágyi, NIMB64 (1992) 811.
- [24] A. Weber, H. Mommsen, NIM 204 (1983) 559.
- [25] I. Bogdanovic-Radovic, O. Benka, NIMB (2001).
- [26] E. Szilágyi, Nucl. Instrum. Methods B 161-163 (2000) 37.
- [27] C. Tschalär NIM 61 (1968) 141, NIM 64 (1968) 237, Phys.Rev.B1 (1970) 2863
- [28] Study of multilayer substrate surface roughness using RBS with improved depth resolution, N.P. Barradas, J.C. Soares, M.F. da Silva, F. Pászti and E. Szilágyi, Nucl. Instr. and Methods B94 (1994) 266.
- [29] Rutherford backscattering analysis of thin films and superlattices with roughness; N. P. Barradas, J. Phys. D: Appl. Phys. 34 (2001) 2109
- [30] Roughness in GaN/InGaN films and multilayers determined with Rutherford backscattering; N. P. Barradas, E. Alves, S. Pereira, V. V. Shvartsman, A. L. Kholkin, E. Pereira, K. P. O’Donnell, C. Liu, C. J. Deatcher I. M. Watson, M. Mayer, Nucl. Instrum. Methods Phys. Res. B 217 (2004) 479
- [31] J.P. Stoquert, T. Szörenyi, Phys. Rev. B 66 (2002) 144108
- [32] M. Tosaki, Energy-loss caused by the inhomogeneity of target material, J. Appl. Phys. 99 (2006) 34905.
- [33] Accurate ion beam analysis in the presence of surface roughness, S.L. Molodtsov, A.F. Gurbich, C. Jeynes, J. Phys. D: Appl. Phys. 41 (2008) 205303
- [34] E. Rauhala, N. P. Barradas, S. Fazinic, M. Mayer, E. Szilágyi, M. Thompson, Nucl. Instrum. Methods B244 (2006) 436
- [35] N.P. Barradas, K. Arstila, G. Battistig, M. Bianconi, N. Dytlewski, C. Jeynes, E. Kótai, G. Lulli, M. Mayer, E. Rauhala, E. Szilágyi, M. Thompson, Nucl. Instrum. Methods B262 (2007) 282