



HAL
open science

Couplage Neutronique - Thermohydraulique à l'échelle du crayon en géométrie tridimensionnelle exacte sans recours à l'homogénéisation : application aux Réacteurs à Eau sous Pression (REP)

Francesco Filiciotto

► **To cite this version:**

Francesco Filiciotto. Couplage Neutronique - Thermohydraulique à l'échelle du crayon en géométrie tridimensionnelle exacte sans recours à l'homogénéisation : application aux Réacteurs à Eau sous Pression (REP). Modélisation et simulation. Université Paris-Saclay, 2023. Français. NNT : 2023UP-ASP142 . tel-04332220

HAL Id: tel-04332220

<https://theses.hal.science/tel-04332220>

Submitted on 8 Dec 2023

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Neutronic - Thermohydraulic coupling at the pin cell scale in exact three-dimensional geometry without homogenization: application to pressurized water reactors (PWR)

*Couplage Neutronique – Thermohydraulique à l'échelle
du crayon en géométrie tridimensionnelle exacte sans
recours à l'homogénéisation : application aux
Réacteurs à Eau sous Pression (REP)*

Thèse de doctorat de l'Université Paris-Saclay

École doctorale n°576 Particules, Hadrons, Energie, Noyau : Instrumentation,
Image, Cosmos et Simulation (PHENIICS)
Spécialité de doctorat: Physique Nucléaire
Graduate School : Physique. Référent : Faculté des sciences d'Orsay

Thèse préparée dans l'unité de recherche **Service d'étude des réacteurs et de mathématiques appliquées (Université Paris-Saclay, CEA)**, sous la direction de **Alain MAZZOLO**, HDR, PhD, l'encadrement de **Emiliano MASIELLO**, PhD, le co-encadrement de **Sandrine COCHET**, PhD, et la co-supervision de **Roland LENAIN**, PhD

Thèse soutenue à Paris-Saclay, le 16 Novembre 2023, par

Francesco FILICIOTTO

Composition du jury

Membres du jury avec voix délibérative

Patrick CIARLET Professeur, ENSTA Paris	President
Alain HÉBERT Professeur, Polytechnique Montréal	Rapporteur & Examineur
Sandra DULLA Professeure, Politecnico di Torino	Rapporteuse & Examinatrice
Jean-Francois VIDAL Directeur de recherche, CEA/DES/IRENE/SPRC, Université Grenoble-Alpes	Examineur
Ansar CALLOO PhD, CEA/DES/ISAS/DM2S/SGLS, Université Paris-Saclay	Examineur

ÉCOLE DOCTORALE

Particules, hadrons, énergie et noyau :
instrumentation, imagerie, cosmos
et simulation (PHENIICS)

Titre: Couplage Neutronique – Thermohydraulique à l'échelle du crayon en géométrie tridimensionnelle exacte sans recours à l'homogénéisation : application aux Réacteurs à Eau sous Pression (REP).

Mots clés: Transport neutronique, Géométries non structurées, Simulation des barres de contrôle sans recours à l'homogénéisation, Expansion asymptotique, Caractéristiques courtes d'ordre parabolique incomplet, Multiphysique.

Résumé: Ce travail concerne le solveur IDT (Integro-Differential Transport) d'APOLLO3® et son application aux calculs multiphysiques couplés neutronique/thermohydraulique. IDT est un solveur de transport aux ordonnées discrètes basé sur des probabilités de collision (P_{ij}) dépendantes de l'angle, avec diffusion P_N . Le domaine spatial est discrétisé en un motif cartésien modulaire, avec des nœuds hébergeant des cellules de type REP en géométrie réelle. Chaque cellule peut loger au plus un crayon, composée de cylindres hétérogènes centrés.

Plusieurs facteurs affectent l'utilisation d'IDT pour la simulation des réacteurs. Le modèle présenté restreint l'application d'IDT à un réseau de cellules conformes. De plus, l'absence de raffinement spatial le long de l'azimut limite la précision, en particulier pour des milieux très absorbants ou hautement diffusifs.

Les méthodes basées sur les P_{ij} permettent une grande flexibilité géométrique, avec la possibilité de modéliser des géométries non structurées. Ainsi, la première partie de ce travail est dédié à la mise en œuvre de nouveaux outils géométriques dans IDT. L'idée de base est de superposer des plans aux cylindres. Un nouveau traçage combinatoire est capable de combiner des bandes $X/Y/Z$ avec des anneaux concentriques. La possibilité d'associer des identifiants cartésiens et radiaux aux régions de calcul, au moyen d'un outil convivial, permet de personnaliser le maillage local et de construire des cellules non structurées/non extrudées. La nouvelle version d'IDT est également capable de modéliser des géométries non conformes, des cylindres excentrés et/ou horizontaux, qui peuvent être utiles pour concevoir des réacteurs VVER et CANDU. Le code montre également une précision accrue, grâce au raffinement spatial, à la fois dans le cas de développements polynomiaux constants et linéaires par morceaux. La possibilité de modéliser des géométries hétérogènes non conformes permet de simuler le mouvement des barres de contrôle, sans homogénéisation.

Un autre problème majeur est lié aux besoins en mémoire des matrices P_{ij} , qui empêche l'application d'IDT aux grands systèmes, avec des milliers de milieux différents. Afin d'atténuer la taille de la mémoire, IDT s'appuie sur une nouvelle technique, basée sur l'expansion de von Neumann, qui permet de réduire le nombre de coefficients de probabilité. Pour résumer, des nœuds qui partagent la même géométrie et ont des chemins optiques similaires peuvent être regroupés dans une seule cellule. L'occupation mémoire des matrices est réduite, ainsi que leur temps de calcul, au prix d'un temps accru pour les itérations internes. L'empreinte mémoire des coefficients peut être également atténuée en exploitant les propriétés de symétrie de chaque cellule. Un ensemble de transformations, basées sur la permutation matricielle et le produit de Hadamard, est maintenant disponible, pour réduire le nombre de directions stockées jusqu'à un facteur 16.

Un développement complémentaire, basé sur une expansion d'ordre parabolique incomplète, fournit un outil additionnel pour augmenter la précision numérique. Contrairement à d'ordre constant et linéaire, l'ordre parabolique nécessite peu de discrétisation spatiale et donc permet une réduction significative du nombre de régions, avec une précision de solution comparable, comme le prouvent les comparaisons avec TRIPOLI-4®.

Une fois les limites majeures d'IDT abordées, l'application aux problèmes multiphysiques est explorée. IDT est couplé avec THEDI (Thermohydraulique Diphasique) et XSTOOL (Cross Section TOOL), pour la mise à jour et l'interpolation des sections efficaces. Une nouvelle voie de simulation est proposée. Différent du calcul en deux étapes, qui est l'approche courante dans les calculs industriels, le schéma numérique présenté est basé sur l'interpolation des sections paramétriques, sans homogénéisation, et ne nécessite pas de modèles de fuite critique, qui introduirait des approximations sur la distribution du flux.

Title: Neutronic - Thermohydraulic coupling at the pin cell scale in exact three-dimensional geometry without homogenization: application to pressurized water reactors (PWR).

Keywords: Neutron Transport, Unstructured Geometries, Control-rod simulation without homogenization, Asymptotic expansion, Incomplete Parabolic Short Characteristics, Multiphysics.

Abstract: This work concerns the Integro-Differential Transport solver (IDT) of APOLLO3® and its application to coupled neutronics/thermal-hydraulics multiphysics calculations. IDT is a discrete ordinates transport solver based on angular-dependent collision probabilities (CP), with P_N -scattering. The spatial domain is discretized into a modular Cartesian pattern, with nodes hosting PWR pin-cells in real geometry. Each cell may lodge at most one pin, composed of centred heterogeneous cylinders.

Several factors prevent the use of IDT for industrial reactor core simulations. First, the presented model restricts the application of IDT to a lattice of conformal pin-cells. Moreover, the lack of spatial refinement along the azimuthal angle limits the numerical accuracy of the code, particularly within strong absorbers or highly diffusive media.

On the bright side, methods based on CP discretization allow for great geometrical flexibility, with the possibility to model unstructured geometries. Hence, the first part of this work is devoted to the implementation of enhanced geometrical capabilities in IDT. More specifically, the basic idea is to superimpose Cartesian planes to cylinders. A novel combinatorial ray tracing is able to combine orthogonal $X/Y/Z$ bands with concentric rings. The possibility to associate Cartesian and radial identifiers to computational regions, by means of a dedicated user-friendly tool, permits to customize the local mesh and construct unstructured as well as non-extruded cells. The new version of IDT is also able to model non-conformal geometries, off-centred cylinders and horizontal pins, which may come in hand to design VVER and CANDU reactors. The code also shows increased numerical accuracy, thanks to the spatial refinement, both in the case of constant and linear polynomial piece-wise developments. The possibility to model non-conformal heterogeneous geometries allows simulating control-rods insertion/withdrawal, with no homogenization and cusping effect.

Another major issue is related to CP matrices memory requirements, which prevents the application of IDT to large systems, with thousands of different media. In order to soften the memory pressure, IDT relies on a novel technique, based on von Neumann series expansion, which permits to reduce the number of angular-probability coefficients. In short, nodes that share the same geometry and have similar optical paths may be gathered into a single reference cell. The memory occupation due to CP matrices is lowered, as well as the simulation runtime invested for their computation, at the cost of an increased time for source iterations. The memory imprint of the angular probability matrices may be also mitigated by exploiting the symmetry properties of each cell. A set of transformations, based on matrix permutation and Hadamard matrix product, is now available in IDT, to reduce the number of stored directions, up to a factor 16.

A complementary development, based on incomplete parabolic series expansion (PSC), provides an additional tool to increase the numerical precision. Unlike constant and linear order, PSC requires little spatial discretization and, as a result, allows for a significant reduction of the number of computational regions, with unmodified solution accuracy, as proven by quantitative comparisons with TRIPOLI4® Monte Carlo. Once addressed the major limits of IDT, the application to multiphysics problems is explored. The solver is coupled with THEDI (Thermohydraulique Diphasique) and XSTOOL (Cross Section TOOL), for cross section update and interpolation. A new simulation pathway is proposed. Differently from standard two-step calculations, which are currently the most common approach in industrial computations, the implemented numerical scheme is based on parametric cross section interpolation, without homogenization, and does not require approximate critical leakage models, which would introduce further assumptions on the flux distribution.

Acknowledgements

Several people contributed, in a way or another, to the success of this thesis work. I therefore take the liberty of stealing a few seconds to mention some of them. Among these, particular thanks go to my supervisors, Emiliano, Sandrine and Roland, and to my thesis director, Alain, for their technical and human support during these three challenging years . I would like to underline my gratitude to Emiliano, who, with his energy, passion and technical guidance, provided me with fundamental help, without which this thesis work would not have been possible.

I also owe a lot to CEA, and in particular to SERMA, which gave me the opportunity, after the internship, to continue my technical and professional growth in reactor physics.

I would like to take this opportunity to express my gratitude to the jury members for their availability and time devoted to this thesis work, despite their busy schedules.

Another person I cannot but thank is a dear friend of mine, Deborah, who stood by me in the hardest moments, when the pandemic caused by the Coronavirus forced us to limit our contact with the outside world. Through her positivity and friendship, she allowed me to carry on even in the most trying days. May the future smile on her.

A fond memory goes to Prof. Mauro Dell'Orso. Like me, generations of students continue to remember him for his professional commitment, his genius and, not least, his extraordinary humanity.

Finally, the deepest thanks go to my family, who has constantly believed in me. I completely dedicate to Giulia, my mother, my father and to my beloved uncle Giovanni this achievement.

Francesco Filiciotto, November 16th, 2023

Contents

1	INTRODUCTION	21
2	RÉSUMÉ EN FRANÇAIS	31
I	SOLUTION OF THE BOLTZMANN EQUATION - STATE OF THE ART	41
3	THE BOLTZMANN EQUATION	43
3.1	CONTEXT	43
3.2	NEUTRONICS	44
3.2.1	Nuclear Reactions Induced by Neutrons	44
3.2.2	Differential Scattering Cross Section and Fission Neutrons Distributions	45
3.2.3	Phase-space Variables and Angular Flux	47
3.2.4	Neutron Transport Equation	48
3.2.5	Initial and Boundary Conditions	49
3.2.6	Monte Carlo vs Deterministic Methods	50
4	SOLUTION OF THE BOLTZMANN EQUATION BY DETERMINISTIC METHODS	53
4.1	TIME-DEPENDENT NEUTRON TRANSPORT EQUATION	53
4.1.1	Prompt and Delayed Neutrons	53
4.1.2	Flux Factorization and Point-Kinetics	54
4.1.3	Improved Quasi-Static Method	55
4.2	STEADY-STATE NEUTRON TRANSPORT	55
4.3	ANGULAR DEPENDENCE OF THE SCATTERING CROSS SECTION	56
4.4	MULTIGROUP APPROXIMATION	57
4.4.1	Multigroup Neutron Transport Equation	57
4.4.2	Iterations	58
4.5	ANGULAR DISCRETIZATION	61
4.5.1	Discrete Ordinates Method	61
4.5.2	PN Method	61
4.6	SPATIAL DISCRETIZATION	62
4.6.1	Integral Form of the Multigroup Transport Equation	62
4.6.2	Collision Probabilities	63
4.6.3	Method of Characteristics	65
4.6.4	Method of Short Characteristics	67
4.7	ACCELERATIONS	71
4.7.1	Linear vs Non-Linear Acceleration Schemes	71
4.7.2	Coarse Mesh Finite Differences	72

5	REACTOR CORE CALCULATION SCHEMES	75
5.1	INTRODUCTION	75
5.2	TWO-STEP APPROACH	76
5.2.1	Descriptive Elements of Lattice Calculations	77
5.2.2	Descriptive Elements of Core Calculations	79
5.3	LIMITATIONS OF THE TWO-STEP STRATEGY AND 3D DIRECT TRANSPORT	80
5.4	2D/1D FUSION METHOD	81
5.5	CONTROL ROD MOVEMENT AND CUSPING EFFECT	83
5.5.1	The Approximate Flux-Weighting Method	84
5.5.2	The Equivalence Method	84
5.5.3	The Sub-plane Method	86
6	INTRODUCTION TO THE INTEGRO-DIFFERENTIAL TRANSPORT SOLVER (IDT)	89
6.1	INTRODUCTION	89
6.2	HISTORICAL NOTE AND MAIN APPLICATIONS	90
6.3	SPATIAL DISCRETIZATION	90
6.3.1	Response Matrix Approach	90
6.3.2	Heterogeneous Cartesian Cells	91
6.3.3	Spatial Sweeping	91
6.3.4	Ray-tracing	92
6.4	DOMAIN DECOMPOSITION IN IDT	94
6.4.1	Problem Position	94
6.4.2	Power Iterations	95
6.4.3	Pivot-Grid, CMFD-Mesh and Subdomain-to-Process-Map Automatic Generation	96
6.5	DISTINCTIVE FEATURES, APPLICATION AND LIMITS OF APOLLO3® IDT	98
II	NEW MODELING CAPABILITIES IN IDT	101
7	GEOMETRICAL NOVELTIES	103
7.1	INTRODUCTION	103
7.2	NEW HCC MODULAR GEOMETRY - A GENERAL OVERVIEW	103
7.3	TUPLE-TO-REGION MAP	106
7.4	COMBINATORIAL RAY-TRACING	107
7.4.1	GEOMETRY TYPES	110
7.5	HORIZONTAL HCCs	112
7.6	NUMERICAL RESULTS - 2D BENCHMARKING	113
7.6.1	Two-dimensional C5G7 Benchmark	114
7.6.2	Two-dimensional 3x3 UOX/MOX Patterns	115
7.6.3	Hexagonal Patterns	118
7.7	NUMERICAL RESULTS - 3D 3x3 BEAVRS-LIKE BENCHMARK	121
7.7.1	Introduction	121
7.7.2	Numerical Results	122

7.8	CONCLUSIONS	125
8	NON-CONFORMING 3D MODEL FOR PWR CONTROL ROD MOVEMENTS WITHOUT HOMOGENIZATION	127
8.1	INTRODUCTION	127
8.2	CONTROL-ROD CUSPING EFFECT	128
8.3	3D C5G7 BENCHMARK: CONFORMAL VS NON-CONFORMAL MESH	129
8.4	AUTOMATIC MEDIUM RELOAD	131
8.5	CONTROL-ROD MOVEMENT IN C5G7	133
8.6	CONCLUSION	134
III	NOVELTIES ON THE METHOD OF SHORT CHARACTERISTICS	135
9	ASYMPTOTIC EXPANSION OF THE ANGULAR FLUX APPLIED TO DISCRETE-ORDINATE SOURCE ITERATIONS	137
9.1	INTRODUCTION	137
9.2	THEORY	138
9.3	APPLICATION TO IDT	139
9.3.1	Implementation in IDT	142
9.4	NUMERICAL TESTS	143
9.4.1	Application to Self-Shielded Lattice	144
9.4.2	Application to depleted lattice	145
9.5	CONCLUSION	150
10	CP COEFFICIENTS - PROPERTIES ANALYSIS AND APPLICATIONS	151
10.1	INTRODUCTION	151
10.2	COEFFICIENTS TABLES AND ASYMPTOTIC APPROXIMATION	151
10.2.1	Background	151
10.2.2	Analytic Remarks	153
10.2.3	Taylor Development and Asymptotic Behaviour	153
10.2.4	Numerical Test	156
10.3	BALANCE EQUATIONS	157
10.4	MEMORY FOOTPRINT REDUCTION, BY SYMMETRY AND RECIPROCITY RELATIONS	158
10.5	CONCLUSION	161
11	INCOMPLETE PARABOLIC SHORT CHARACTERISTICS	163
11.1	INTRODUCTION	163
11.2	THEORY	163
11.3	REMARKS	165
11.4	NUMERICAL TESTS	165
11.4.1	Numerical Accuracy of PSC - 2D 3x3 UOX/MOX patterns	166
11.4.2	Numerical Accuracy of PSC - 3D Watts Bar Benchmark	172
11.4.3	Use of Symmetry Relations	173

11.5	MIXED-ORDER SHORT CHARACTERISTICS	174
11.5.1	Numerical Accuracy of HSC - 3D C5G7 Benchmark	175
11.6	CONCLUSIONS	176

IV MULTIPHYSICS ITERATIONS IN IDT 177

12	NEUTRONICS/THERMAL-HYDRAULICS COUPLING	179
12.1	LIAISON AND OBJECTIVES	179
12.2	INTRODUCTION	179
12.3	EFFECTIVE MACROSCOPIC CROSS-SECTION GENERATION	181
12.3.1	Multigroup approximation and self-shielding	181
12.3.2	Thermal feedback	182
12.3.3	Standard two-step reactor calculations	182
12.3.4	Advanced two-step reactor calculations	183
12.3.5	A brief introduction to XSTOOL	184
12.4	PARAMETRIC EFFECTIVE MICROSCOPIC CROSS-SECTION ANALYSIS	187
12.4.1	Test Case Specifications	187
12.4.2	Dependence on burn-up	187
12.4.3	Dependence on burn-up and fuel temperature	193
12.5	Preliminary V&V and memory occupation of the parametric library of Shelf-Shielding Factors (EPL-SSF)	195
12.6	Xenon initial number density	199
12.7	XSTOOL/IDT/THEDI COUPLING	200
12.7.1	IDT input file	201
12.7.2	Automatic generation of the domain decomposed and of XSTOOL inputs	204
12.7.3	Boron concentration and temperature field initialization	204
12.7.4	TH sub-channels definition	204
12.7.5	Macroscopic Cross-Section initialization by XSTOOL	206
12.7.6	CP Matrices evaluation	207
12.7.7	Eigenpair and Fission Integral initialization	208
12.7.8	Outer Iterations	208
12.7.9	Multigroup Transport Iterations and CMFD	208
12.7.10	Error Check	209
12.7.11	Critical Boron Concentration	210
12.7.12	Neutronics/Thermal-Hydraulics coupling	210
12.7.13	Generation of Power Map input for TH simulation	211
12.7.14	IDT2THEDI. Thermal-Hydraulic feedback by THEDI	211
12.7.15	Macroscopic Cross Section Update by XSTOOL	213
12.7.16	CP coefficients recalculation	213
12.8	NUMERICAL RESULTS	214
12.8.1	Energy condensation - Verification test based on a small hexagonal pattern	214
12.8.2	Neutronics/Thermal-Hydraulics coupling - 3D 17x17 UOX/B4C lattice	217

12.8.3	Multiphysics iterations, with critical boron concentration search - 17x17 UOX/B ₄ C 3D lattice	225
12.8.4	Coupled neutronics/thermal-hydraulics simulations, with control-rod movement	228
13	Conclusions and perspectives	233
13.1	Objectives and requirements	233
13.2	Towards a new version of IDT	233
13.2.1	New geometrical capabilities	233
13.2.2	Novelties on the method of short characteristics	234
13.3	Multiphysics	235
13.4	Future work	235
13.4.1	Geometric model	235
13.4.2	PSC and hybrid-order short characteristics	235
13.4.3	Multiphysics and industrial-scale calculations	236
13.4.4	Control-rod withdrawal and insertion	236
13.4.5	IDT coupled to more detailed thermal-hydraulic codes	236
13.4.6	IDT coupled to thermal-mechanical codes	236
A	THEDI - A BRIEF OVERVIEW	239
A.1	INTRODUCTION	239
A.2	1D AXIAL THERMAL-HYDRAULICS	239
A.2.1	Notation	239
A.2.2	Balance equations	240
A.2.3	Numerical solution	241
A.3	1D RADIAL DIFFUSION	241
B	ANALYTICAL COMPUTATION OF THE FLUX MOMENTS BY TAYLOR EXPANSION	243
C	ASYMPTOTIC METHOD - SPECTRAL RADIUS ANALYSIS	245

List of Figures

1.1	Microscopic total cross section of plutonium 239. From https://www-nds.iaea.org/exfor/endl.htm , Evaluated Nuclear Data File (ENDF).	22
1.2	Two-step calculation scheme.	24
1.3	Two-group homogenized absorption cross section of gadolinium 157 vs burn-up. Figures from: I. Zmijarevic, "Some aspects of creating and using multi-parameter cross section library", DES/ISAS/DM2S/SERMA/LLPR/NT/2022-70741/A, Courtesy of Igor Zmijarevic.	25
1.4	$\sigma_{U-238}^{g,in}$ and $\sigma_{U-238}^{g,out}$ vs burn-up, for $g \in [42, 89]$, where <i>in/out</i> indicate the inner/outer ring of the fuel cell.	29
2.1	Section efficace microscopique totale du plutonium 239. From https://www-nds.iaea.org/exfor/endl.htm , Evaluated Nuclear Data File (ENDF).	32
2.2	Schéma de calcul en deux étapes.	34
2.3	Section efficace d'absorption du gadolinium 157 homogénéisée à 2 groupes en fonction du burnup. Figures de: I. Zmijarevic, "Some aspects of creating and using multi-parameter cross section library", DES/ISAS/DM2S/SERMA/LLPR/NT/2022-70741/A, Courtesy of Igor Zmijarevic.	35
2.4	$\sigma_{U-238}^{g,in}$ et $\sigma_{U-238}^{g,out}$ en fonction du burnup, pour $g \in [42, 89]$, où <i>in/out</i> indique la couronne interne/externe de la cellule combustible.	40
3.1	Total cross section of $^{238}_{92}U$. From https://www-nds.iaea.org/exfor/endl.htm , Evaluated Nuclear Data File (ENDF).	46
3.2	Total cross section of $^{16}_8O$. From https://www-nds.iaea.org/exfor/endl.htm , Evaluated Nuclear Data File (ENDF).	46
4.1	Fission reaction.	54
4.2	Improved Quasi-Static method.	56
4.3	MOC tracking.	66
4.4	Heterogeneous Cartesian Cell (HCC).	68
5.1	Thermal-hydraulics sub-channel mesh, represented by dashed lines.	80
5.2	VERA Core Simulator: cross section update and outer iteration scheme.	83
5.3	Rod cusping effect: reactivity vs control rod tip position.	84
5.4	Rodded node.	84
6.1	2D heterogeneous cells.	91
6.2	Examples of deformed 3D pin-cells.	91
6.3	Heterogeneous Cartesian Cell (HCC).	92
6.4	Sweeping by front.	92
6.5	Ray-tracing.	93

6.6	Serpentine path, for a 3x3x3 pivot grid.	97
6.7	Sub-domain distribution on 4 MPI processes, for a 3x3x3 pivot grid.	97
7.1	New Heterogeneous Cartesian Cell.	104
7.2	Non-conforming inner grids.	104
7.3	Quarter and half pin.	104
7.4	Chunk of structural component.	105
7.5	Hexagonal lattice.	105
7.6	Automatic mesh generator nb. 1. Each ray-tracing region is mapped onto a different computational region.	105
7.7	Automatic mesh generator nb. 2. The ray-tracing regions within the same circle are merged into a single computational region. The spatial refinement is maintained in the surrounding moderator ($r = 0$).	105
7.8	Non-automatic mesh generator. The number of computational regions has been reduced to $N_\alpha = 7$, by assigning the same α to different ray-tracing regions.	106
7.9	Non-extruded cell: on the left-hand-side, the plane-by-plane ray-tracing geometry; at the centre, the plane-by-plane computational geometry; on the right-hand-side, the transverse section.	107
7.10	Original frame of reference (before rotation).	108
7.11	XY and tZ -tracking.	108
7.12	Step 1: Discontinuity projection + uniform ray spacing (omitted) + Gauss points (omitted).	109
7.13	Step 2: Geometrical patterns.	109
7.14	Step 3: Chord-lengths and TRM, with x_{min} and x_{max} being the coordinate of the incoming and outgoing point of the trajectory. Colour legend: red $\equiv i = 1$, green $\equiv i = 2$, yellow $\equiv j = 1$, purple $\equiv j = 2$, black $\equiv r = 1$	109
7.15	Volume computation for a set of heterogeneous Cartesian patterns. On the left-hand-side, the relative error obtained by varying the ray-spacing parameters (i.e., Δ_{XY} and number of Gauss points); on the right-hand-side, the geometrical motifs (labelled by an integer number, ranging from 1 to 11).	111
7.16	HCC orientation along x	112
7.17	Non-conforming geometries and pin-cell orientation along different axes.	114
7.18	2D C5G7 benchmark. Radial reflector in blue.	114
7.19	Spatial mesh for UOX/AIC LSC simulation. UOX cells comprise 11 computational regions (of which 4 UOX rings, 1 air-gap ring, 1 Zr-clad ring and 5 moderator subdivisions), while the central cell is finely discretized into 40 regions, where all rings (8, in total) are subdivided into 4 sectors and the surrounding moderator is finely discretized into 8 slices.	116
7.20	MOX-B ₄ C benchmark: relative error on the scalar flux per energy group and material region (reference provided by TRIPOLI-4® continuous-energy Monte Carlo). More specifically, the figure refers to the innermost fuel ring, within a cell in a corner position. The curve labeled "idt" (in red) is obtained with LSC spatial approximation. The error curves follow the same trend and show localized peaks within the same energy groups.	117
7.21	Average RMS relative error per material. The largest errors are observed in AIC and Hf rods. In the fuel rods, all calculations do not exceed 1.85% average RMS relative error.	117

7.22	Relative error on the fundamental eigenvalue and CPU time.	117
7.23	Hexagonal pattern, with fuel bundle.	118
7.24	Fuel bundle construction for hexagonal patterns.	119
7.25	Fission integral distribution for a 21×14 two-dimensional hexagonal pattern, with UOX fuel and light water as moderator. Specular reflection boundary conditions are imposed on the border.	120
7.26	Absorption reaction rate distribution for a 21×14 two-dimensional hexagonal pattern, with UOX fuel and light water as moderator. Specular reflection boundary conditions are imposed on the border.	120
7.27	CR completely inserted. Image by TRIPOLI-4®.	121
7.28	CR completely inserted (on the left-hand-side), at half height (in the centre) and completely extracted (on the right-hand-side). Image by TRIPOLI-4®.	122
7.29	Qualitative comparison between IDT and TRIPOLI-4® continuous-energy Monte Carlo, for each CR position.	123
7.30	Absolute error with respect to TRIPOLI-4® continuous-energy Monte Carlo, for each CR position.	124
8.1	Control-rod cusping effect.	128
8.2	Test based on the 3D C5G7 benchmark: the control rods are gradually inserted, by 24 steps of 1.19 cm each.	129
8.3	Conformal vs non-conformal HCCs along z.	130
8.4	C5G7 benchmark: conformal vs non-conformal mesh.	131
8.5	Control-rod simulation in the C5G7 benchmark, using nodes with $\Delta h = 3.57 \text{ cm}$: k_{eff} vs control-rod position (solid lines) and relative error with respect to "Ref-IDT" (dashed lines). N.B. The control rods are gradually inserted with a step size of 1.19 cm.	133
8.6	Control-rod simulation in the C5G7 benchmark, using nodes with $\Delta h = 14.28 \text{ cm}$: k_{eff} vs control-rod position (solid lines) and relative error with respect to "Ref-IDT" (dashed lines). N.B. The control rods are gradually inserted with a step size of 1.19 cm.	134
9.1	Unstructured non-conformal 3D Heterogeneous Cartesian Cell mesh.	139
9.2	2D UOX/B ₄ C benchmark.	144
9.3	TRIPOLI4® vs Asymptotic IDT: relative error on the pin-by-pin absorption and fission distributions.	145
9.4	Standard vs Asymptotic IDT: maximum relative error per row on the pin-by-pin fission and absorption distributions.	146
9.5	Maximum total-cross-section variation (%) vs burn-up.	146
9.6	Memory for coefficients vs burn-up.	147
9.7	Time for HCC coefficients vs burn-up.	147
9.8	k_{eff} (blue and red curves) and reactivity error (green curve) vs burn-up.	148
9.9	Time for inner and outer iterations vs burn-up.	149
9.10	Time for inner and outer iterations vs burn-up (asymptotic IDT is optimized by including high-order moments only once k_{eff} has reached convergence).	149

10.1	G_n vs τ , for $n = 0, 1$ (linear short characteristics) and $\tau \in [0, 30]$	154
10.2	H_{nm} vs τ , for $n = 0, 1, m = 0, 1$ (linear short characteristics) and $\tau \in [0, 30]$	154
10.3	G_0 vs G_0^∞ , as a function of the optical path, for $\tau \gtrsim 1$	155
10.4	Relative difference between G_0 and G_0^∞ , as a function of the optical path, for $\tau > 30$	155
10.5	Symmetry relations in IDT: 2D cell with $1/8^{th}$ symmetry. N.B. Ω_{sym} is obtained by reflection of Ω_{ref} (reference direction) about the direction at 45° . Ω_d is perpendicular to Ω_{sym}	158
10.6	Symmetry relations in IDT: asymmetric HCC.	159
10.7	Reciprocity relations.	160
10.8	Symmetry relations in IDT: asymmetric 3D cell along the z-axis. Symmetry on the XY plane is still verified.	161
11.1	PSC IDT, with $N_{sub} = 1, 2, 3, 4$, vs TRIPOLI-4® continuous-energy Monte Carlo: on the left-hand side, the RMS relative error, computed over 281 energy groups, per SSH material, for the UOX-AIC benchmark; on the right-hand side, the relative error on the fundamental eigenvalue, for the ensemble of test cases. The curves pertaining to $N_{sub} = 3$ and $N_{sub} = 4$ tend to superimpose.	166
11.2	Comparison with TRIPOLI-4® continuous-energy Monte Carlo: average RMS relative error on the scalar flux, for each material type, computed over 281 energy groups.	167
11.3	Comparison with TRIPOLI-4® continuous-energy Monte Carlo: average RMS relative error on the scalar flux, for each material type, computed over 281 energy groups.	168
11.4	UOX-Hf benchmark - PSC vs TRIPOLI-4® continuous-energy Monte Carlo: relative error on the multigroup scalar flux, for the Hf-rod innermost ring.	168
11.5	RMS relative error on the multigroup scalar flux per self-shielding region, with respect to APOLLO3®-TDT Step MOC.	169
11.6	RMS relative error on the multigroup scalar flux within the three innermost rings of the control rod, for the UOX-AIC case, with respect to TRIPOLI-4® continuous-energy Monte Carlo.	170
11.7	Self-shielded total macroscopic cross section of AIC (Ag-In-Cd), within the innermost ring of the control rod, for the 2D 3x3 UOX-AIC benchmark.	170
11.8	UOX-AIC benchmark - PSC and Step MOC vs TRIPOLI-4® continuous-energy Monte Carlo: relative error on the multigroup scalar flux, for the AIC-rod innermost ring.	171
11.9	Relative error on the fundamental eigenvalue, with respect to TRIPOLI-4® continuous-energy Monte Carlo.	171
11.10	PSC vs LSC: total memory footprint of the angular probability matrices and simulation runtime on one thread, provided that only the reciprocity relations are applied. The memory imprint is reduced by 20 – 30%, from LSC to PSC.	171
11.11	3D Watts Bar benchmark.	172
11.12	LSC (on the left hand side) vs PSC (on the right hand side): fission integral distribution along the z-axis (IDT in red, TRIPOLI-4® in blue) and relative error with respect to TRIPOLI-4®, for the fuel rod located in position $x = 10, y = 9$	173

11.13	Symmetry relations in IDT: on the left hand side, a heterogeneous cell with $1/8^{th}$ symmetry, while, in the middle and on the right hand side, a homogeneous square cell, with unitary side length, illuminated by direction Ω_{ref} oriented along the diagonal and by direction Ω_{\perp} , respectively.	173
12.1	Hierarchical structure of the External Parametric Library of Self-Shielding Factors (EPL-SSF) (in the figure, UDM is an acronym for the User Data Model of APOLLO3®).	185
12.2	Average microscopic absorption cross section vs RMS for two fuel isotopes.	188
12.3	$\sigma_{U-238}^{g,in}$ and $\sigma_{U-238}^{g,out}$ vs burn-up, for $g \in [42, 89]$	189
12.4	$\sigma_{Pu-239}^{g,in}$ and $\sigma_{Pu-239}^{g,out}$ vs burn-up, for $g \in [54, 92]$	189
12.5	$\Delta_{U-238}^{g,in}(BU)$ and $\Delta_{U-238}^{g,out}(BU)$ vs burn-up, for $g \in [54, 92]$. The black curve on the right-hand side corresponds to $g = 79$	190
12.6	$\Delta_{Gd-157}^{g,out}(BU)$ vs burn-up, for $g \in [79, 92]$	190
12.7	$\sigma_{Gd-157}^{0,out}(BU)$ and $\sigma_{Gd-157}^{1,out}(BU)$ vs burn-up, where 0 = fast group and 1 = thermal group. Figures from: I. Zmijarevic, 'Some aspects of creating and using multi-parameter cross section library', DES/ISAS/DM2S/SERMA/LLPR/NT/2022-70741/A, Courtesy of Igor Zmijarevic.	191
12.8	Legendre moments of $\sigma_{U-238,abs}^{g,in}$, up to $l = 6$	191
12.9	$\sigma_{U-238,abs}^{89,in}$ vs its approximation by Legendre series expansion of order 6 (in red and green, respectively), for group 89, as a function of burn-up. In blue, the relative error (%).	192
12.10	Linear-tent approximation. The parametric microscopic cross section is approximated by a linear spline, connecting the points $(BU_n, \sigma_j^{g,e}(BU_n))$ and $(BU_{n+2}, \sigma_j^{g,e}(BU_{n+2}))$, allowing for a major reduction of the number of burn-up steps.	192
12.11	Linear-tent approximation. $\sigma_{U-238,abs}^{89,in/out}$ (in red) vs its linear-tent approximation (in green), as a function of BU . In blue, the relative error (%).	193
12.12	Average microscopic absorption cross section vs RMS for two fuel isotopes.	194
12.13	Dependence on burn-up and fuel temperature: $\sigma_{U-238,abs}^{84,in/out}$ vs BU and T_f	195
12.14	$BU - T$ factorization. Relative difference, i.e., $\epsilon\%$ (Eq. (12.24)), observed for $\sigma_{U-238,abs}^{g,out}$ ($g = 83$ on the left-hand side, $g = 84$ on the right-hand side), as a function of BU , for four different values of temperature.	196
12.15	$BU - T$ factorization. Relative difference, i.e., $\epsilon\%$ (Eq. (12.24)), observed for $\sigma_{U-238,abs}^{g,out}$ ($g = 79$ on the left-hand side, $g = 89$ on the right-hand side), as a function of BU , for four different values of temperature.	196
12.16	State-points grid in the parametric phase space ($Bu, T_f, T_m, C_B = 0$).	197
12.17	Hexagonal pattern geometry	215
12.18	26 vs 281 energy groups: fuel power relative error (%).	216
12.19	30 vs 281 energy groups: fuel power relative error (%).	216
12.20	17x17 tridimensional UOX/ B_4C lattice: one eighth of the radial geometry.	220
12.21	k -eigenvalue error vs iteration number, for $\mathcal{N} = 1$ ('Strong') and $\mathcal{N} = 5$ ('Weak').	221
12.22	Fission source error vs iteration number, for $\mathcal{N} = 1$ ('Strong') and $\mathcal{N} = 5$ ('Weak').	221
12.23	Inner loop error vs iteration number, for $\mathcal{N} = 1$ ('Strong') and $\mathcal{N} = 5$ ('Weak').	222
12.24	'Strong' vs 'Weak': fuel power distribution for the rod in position $x = 6, y = 5$	223

12.25	'Strong' vs 'Weak': fuel power relative difference (%) for the rod in position $x = 6, y = 5$, assuming 'Strong' as the reference.	223
12.26	Fuel power relative difference, between $\mathcal{N} = 1$ and $\mathcal{N} = 5$, assuming $\mathcal{N} = 1$ as the reference, in the plane $z = 11$	224
12.27	Fuel power relative difference, between $\mathcal{N} = 1$ and $\mathcal{N} = 5$, assuming $\mathcal{N} = 1$ as the reference, in the plane $z = 3$	224
12.28	Fuel temperature difference [K], between $\mathcal{N} = 1$ and $\mathcal{N} = 5$, in the plane $z = 11$	224
12.29	Fuel temperature difference [K], between $\mathcal{N} = 1$ and $\mathcal{N} = 5$, in the plane $z = 5$	224
12.30	Moderator temperature [K]: comparison between 'Strong' and 'Weak'.	224
12.31	Moderator temperature difference [K] between 'Strong' and 'Weak', as a function of z	224
12.32	Fission source error vs number of outer iterations, for a 17×17 UOX- B_4C 3D lattice, with control rod tips at 370 cm , in tightly coupled N-TH iterations, with critical boron concentration search.	226
12.33	Inner loop error vs number of outer iterations, for a 17×17 UOX- B_4C 3D lattice, with control rod tips at 370 cm , in tightly coupled N-TH iterations, with critical boron concentration search.	226
12.34	Eigenvalue error vs number of outer iterations, for a 17×17 UOX- B_4C 3D lattice, with control rod tips at 370 cm , in tightly coupled N-TH iterations, with critical boron concentration search.	227
12.35	Total axial fuel power [MW]: comparison between 'Strong' and 'Weak'.	227
12.36	Moderator temperature [K]: comparison between 'Strong' and 'Weak'.	227
12.37	Resonant isotope microscopic cross-section interpolation, as a function of the percentage of the control rod inserted in the plane. In bold, an axial plane, housing a non-conforming mesh, which allows to reduce the spatial discretization of the fuel rods, but leaves the nodes that may host a control rod finely discretized.	228
12.38	Reference vs Target: axial fuel-power distributions.	230
12.39	Relative difference on axial fuel-power distributions.	230
12.40	Relative difference on pin-by-pin fuel-power distributions over plane 37 out of 40.	230
12.41	Absolute difference on pin-by-pin fuel temperature distributions, over plane 4 out of 40.	230
12.42	Reactivity-worth curve obtained by 'Target'.	231
12.43	'Reference' vs 'Target': observed memory saving for a single configuration, more specifically the one where the control rod tips are located at 245 cm . Note that the figure displays the number of input and computed cells (blue lines and orange line, respectively), for each subdomain, over all power iterations.	232
13.1	Neutron-transport/fuel-performance coupling: a possible line of research for future applications of IDT.	236

List of Tables

3.1	Main reaction types occurring in a fission reactor.	45
4.1	SP_N vs P_N method: moments in grey are neglected in SP_N	62
7.1	Spatial and angular discretization.	115
7.2	Numerical results and CPU time on a single thread.	115
7.3	Number of regions for each UOX/MOX pattern. Note that this number is the same for both fuel types, except for the AIC benchmark.	115
7.4	IDT: spatial and angular discretization.	122
7.5	IDT vs TRIPOLI-4®: k -eigenvalue.	124
7.6	IDT vs TRIPOLI-4®: CR worth error (%).	124
7.7	IDT: CPU time and allocated resources.	125
9.1	Lattice problem settings.	143
9.2	Standard (reference) vs Asymptotic IDT, performed by LSC and $S_{48,8}$ -Chebyshev-Legendre product quadrature.	145
11.1	Size of the CP matrices, given 1 HCC, 1 energy group and 1 discrete direction: N_r and N_s denote the number of regions and the number of surfaces for the considered cell, while C and B indicate the number of volume and surface components of the polynomial series expansion.	166
11.2	LSC vs PSC: total number of regions for each UOX/MOX pattern. For LSC, UOX-AIC and MOX-AIC do not share the same mesh. In this case, the number of regions is indicated separately as UOX/MOX.	167
11.3	LSC vs PSC: number of computational regions and numerical accuracy. The reference is provided by TRIPOLI-4® continuous-energy Monte Carlo.	173
11.4	Rodded A (= A), Rodded B (= B) and Unrodded (= U) configurations.	175
12.1	APOLLO3® vs XSTOOL+IDT, for a 2D 3.2% UOX-Gd32 assembly simulation.	198
12.2	Memory occupation of the parametric HDF file (EPL-SSF) for several lattice types.	199
12.3	Memory occupation of the EPL-SSF file for a depleted UOX-Gd32 assembly.	199
12.4	Sub-channel geometry out/input file, generated by IDT, for N/TH coupling.	206
12.5	Power-map out/input file, generated by IDT, for N/TH coupling.	211
12.6	Computed eigenvalues and critical boron concentrations, using 26/30/281-group macroscopic cross section libraries. In the last column, the relative error on the critical boron concentration, assuming the calculation with no energy condensation (i.e., 281 groups) as the reference.	215
12.7	Simulation runtime. The spatial domain is divided into 3×4 subdomains and run on 12 OpenMP threads.	216
12.8	17x17 tridimensional UOX/ B_4C lattice: axial geometry and composition.	218

12.9	Computed eigenvalues for $\mathcal{N} = 1$ and $\mathcal{N} = 5$ and relative difference between the two, calculated by taking the simulation corresponding to $\mathcal{N} = 1$ as the reference.	221
12.10	Total CPU time [min] and percentage of time spent to recompute the probability matrices and the macroscopic cross section libraries.	225
12.11	Computed eigenvalue and critical boron concentration.	225
12.12	'Reference' vs 'Target': time for microscopic cross section interpolation and coefficients and percentage with respect to the total CPU time.	232

1 - INTRODUCTION

Of all the major challenges ahead of us, climate crisis is probably the most urgent and calls for immediate solutions. A growing number of countries have already committed to achieving net-zero greenhouse gas emissions by 2050. In this framework, institutions, such as the International Energy Agency (IEA), have suggested sustainable pathways for a technically and economically feasible energy transition, [1], indicating nuclear and hydroelectric as essential for low-carbon energy production. On February 10th, 2023, the President of the French Republic announced the construction of 6 next-generation European Pressurized Reactors (EPRs), with additional 8 reactors under study, to reach 25 GW_e of new installed nuclear capacity, by mid-century. It appears clear that the development of this technology will play a key role in the coming years.

A fission reactor is a system capable of hosting a controlled self-sustained fission reaction chain. A fission reaction is a complex process, initiated by a neutral particle. More specifically, a neutron is fired at a fissile nucleus, which splits in two lighter fragments, with the release of additional neutrons and prompt γ -rays. A mass defect Δm of about 10% of the mass of the target nucleus is observed. A large fraction of Δm is converted into kinetic energy of the daughter nuclei ($\sim 170 \text{ MeV}$), whereas smaller amounts appear as kinetic energy of the secondary neutrons (few MeV each) and of the γ -photons (several MeV). Overall, the energy released by a fission event is several hundreds of thousands of times larger than the one generally released in a chemical reaction. The daughter nuclei are charged particles and lose energy by successive collisions with the surrounding atoms, determining a temperature increase in the nuclear fuel. The vast majority of nuclear reactors are based on fissions induced by thermal neutrons, i.e., neutrons in thermal equilibrium with the medium. Neutron slowing-down from high energy ($\sim \text{MeV}$) to fractions of eV may be achieved by means of a material with low atomic number, i.e., a moderator. Most nuclear reactor designs employ water for such a purpose, which also serves as coolant, to mitigate the nuclear fuel temperature. The production of steam, which drives a turbine, allows to generate an alternating current, which is fed into the electric network.

Safe operation and performance are key points to be met in the design of a fission reactor. Several factors are involved, one of the most decisive being the prediction of the neutron distribution. The fundamental equation describing the neutron density, as a function of position, angle, energy and time, namely $n(\mathbf{r}, \Omega, E, t)$, is the neutron transport equation. This may be derived from the more popular Boltzmann equation, by neglecting neutron-to-neutron collisions, as rarely occurring in real multiplying systems. Neutrons may undergo reactions of different kind, including fission, sterile capture, scattering, $(n, 2n)$ -reactions, to name a few, depending on the target nucleus and the incident neutron energy. Neutron interactions are intrinsically stochastic and may only be described in terms of microscopic cross sections, σ , which may be pictured as the cross-sectional area exhibited by the target nucleus when a neutron is about to collide. Within a certain energy range, which is comprised between few eV to hundreds of keV for heavy nuclides, microscopic cross sections show sharp variations as a function of the neutron speed, known as resonances. This behaviour strongly affects the trend of the neutron flux, defined as the product between neutron density and speed, namely $\varphi = n \cdot v$, which, in turn, will exhibit abrupt variations. The probability of neutron interaction per unit path de-

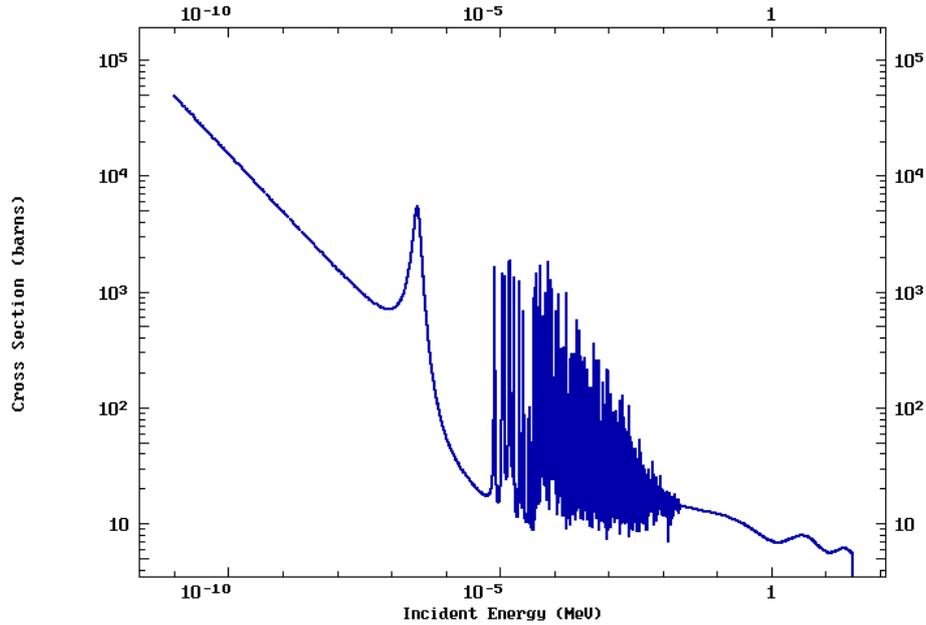


Figure 1.1: Microscopic total cross section of plutonium 239.
 From <https://www-nds.iaea.org/exfor/endl.htm>, Evaluated Nuclear Data File (ENDF).

depends linearly on the microscopic cross section and on the nuclear density, N . The product of the two defines the macroscopic cross section, $\Sigma = N \cdot \sigma$.

As mentioned, nuclear reactions release a considerable amount of energy, thus affecting the temperature distribution of all components, and, in particular, of the nuclear fuel and the moderator. In return, the temperature raise has two consequences. The first is related to thermal agitation and goes under the name of Doppler effect. In practice, due to the relative motion between the projectile and the target, the cross-section resonances broaden, increasing the probability of a neutron to disappear due to an absorption event. The second effect mostly concerns the mass density of the moderator. Due to the increase in temperature, and assuming that the reactor is under-moderated, the number of light targets per unit volume reduces and, as a result, moderation drops. As most fission reactions are caused by thermal neutrons, the neutron population lowers. This simple analysis permits to comprehend that a nuclear reactor is a complex multiphysics system, where several processes, in particular, neutronics, thermal-hydraulics and fuel performance, show strong interdependencies. Generally speaking, a multi-physics problem may be approached in two different manners, i.e., by a monolithic process or, alternatively, a segregated strategy, [2]. The former solves the whole set of balance equations concurrently at each step, whereas the latter analyses each physical process by itself, assuming the solution of all the other physical problems as fixed. The second approach requires multiple iterations between the two 'mono-physics', where each process provides a set of data to feed into the others, to simulate the corresponding feedback. Reactor core calculations generally implement the segregated strategy, solving, on the one hand, the neutron transport equation, and, on the other, thermal-hydraulics and fuel thermal-mechanics, by dedicated modules or coupled codes, running successive iterations between the 'mono-physics', to converge the multiphysics solution.

The neutron transport equation may be solved by either deterministic or Monte Carlo methods, [3]. The latter are better suited for problems with high geometric complexity, to evaluate

integral quantities, such as the absorbed dose, in radiation protection calculations, or the reactor period, in criticality safety assessments. Conversely, the evaluation of differential quantities, such as the neutron flux, is generally better addressed by deterministic methods, as more computationally efficient and less time consuming. Although being more accurate in terms of energy representation, Monte Carlo simulations are, at best, impractical, when considering the whole reactor cycle.

In recent years, the outstanding growth in computational power has rendered the application of *3D direct transport* techniques more attractive for reactor core calculations. An example is provided by the discrete-ordinate OpenMOC transport solver [4], applying the method of characteristics in real 3D geometries and implementing scalable domain decomposition for massive parallelization. In general, 3D full core neutron transport shows great convergence stability and high numerical accuracy. This approach is particularly suited for applications to reactor cores with strong axial heterogeneities, and, in general, exhibits better agreement with reference continuous-energy Monte Carlo solutions.

Despite progress in computer hardware, high-fidelity full core transport solutions are still unfeasible for industrial reactor cycle simulations. Hence, the vast majority of commercial analyses rely on the so-called *two-step* strategy, based on dimensionality reduction, by spatial homogenization and energy condensation, in order to limit the computational effort as well as the required resources, to run the simulation.

In the first step, the neutron transport equation is solved with refined energy and spatial discretization, for a set of repetitive patterns, e.g., the fuel assemblies, with simplified boundary conditions. As for a large set of industrial reactors most of the heterogeneity is concentrated within the radial plane, fuel assemblies are assumed to be two-dimensional. At first, mutual neutron current exchanges between assemblies are neglected, imposing reflection boundary conditions. The spatial solution of the neutron transport equation is typically derived by using either 2D long characteristics or (current-coupled) collision probabilities. Neutron leakage at a lattice boundary is included artificially. Assuming that the reactor is *critical*, i.e., it is able to support a self-sustaining fission reaction chain, an approximate model is employed, where the *effective multiplication factor* (namely the ratio between neutron production and disappearance, k_{eff}) is enforced to 1. In practice, the neutron flux is represented as the product of a "macroscopic" distribution, depending on the sole position, and the fine spectrum, depending on the whole set of independent variables, obtained by solving the transport equation with reflection boundary conditions. The "macroscopic" flux is assumed to solve a Laplace equation, where a critical parameter (buckling, B^2) is searched by successive iterations. As the buckling represents, up to the sign, the second derivative of the 'macroscopic' distribution, this procedure may be pictured as an iterative adjustment of the flux curvature, in order to converge k_{eff} to 1, [5]. The lattice simulation allows to compute a few-group cross section library, with homogenized media. This calculation is repeated for a variety of operating conditions or *parameters*, which may include a set of temperatures (e.g., the local value T , the temperature of the fuel and the temperature of the moderator), the moderator density, the type of assembly, the fuel burn-up, the boron concentration, etc.

In the second step, the neutron flux and reaction rates are computed over the whole reactor core, using the constructed database, containing the full set of nuclear data generated by the lattice calculation. The reactor core is modelled as a tridimensional array of homogenized assemblies or pin-cells and is solved by simplified transport, with condensed energy groups. Neutron leakage is accounted for by imposing exact boundary conditions at the reactor bor-

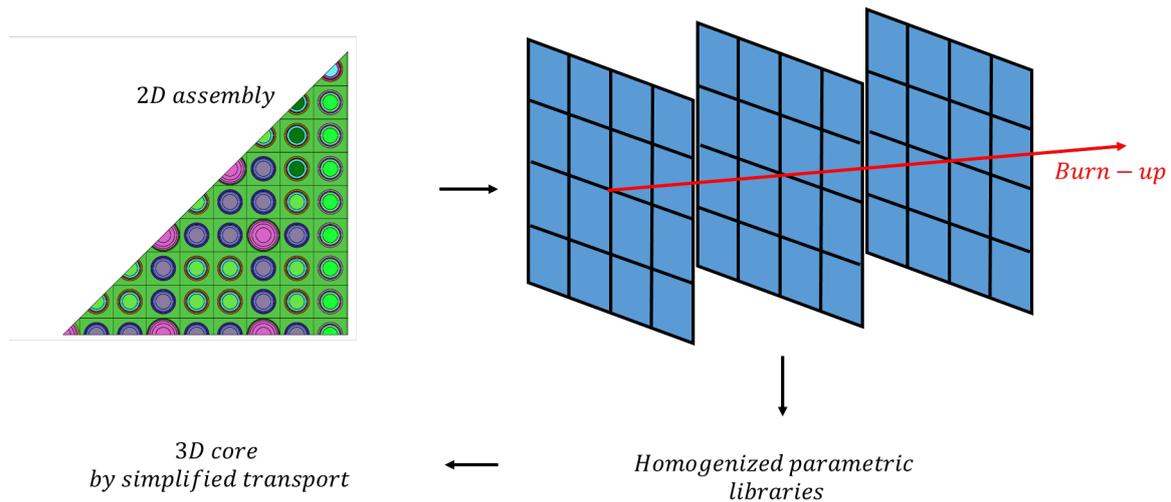


Figure 1.2: Two-step calculation scheme.

der. The solution may be searched for reactor transients or under stationary conditions. A simple neutron balance, equating neutron disappearance and production, is solved over each coarse node and condensed energy group, where leakage over the node boundary may be represented by the *Fick law*, relating the neutron current to the negative of the gradient of the flux, [6].

Despite their popularity and undisputed role in nuclear industry, two-step calculations are affected by a number of limitations. At the lattice level, the net exchanges between fuel assemblies are first neglected, providing a multiplication factor that can differ significantly from the one of the whole system. This is particularly true for fast reactors, where neutron mean free paths tend to be longer, resulting in stronger coupling between neighbouring assemblies, [7]. Assuming the system is critical, a B^2 -critical leakage model is needed, in order to converge k_{eff} to 1. As mentioned above, one of the main approximation of this method consists in assuming that the neutron flux can be represented as the product of a position-dependent macroscopic function and of the detailed 2D-infinite transport solution.

A crucial point of the two-step calculation lies in the production of the few-group parametric library, with homogenized pin-cells, for a large set of *state points*. Each state point is defined by a collection of operating parameters, assuming discrete values over a multidimensional grid. The generated library is used by a simplified neutron transport solver, to infer the solution over the whole reactor. More specifically, in the second step, the interdependencies of neutronics, thermal-hydraulics and isotope evolution are considered. Each depletion simulation is treated as a steady-state calculation, which takes into account thermal-hydraulics feedback on neutron transport, along with finding the boron concentration and the position of the control rods that make the system critical. In order to deduce the cross sections for the current reactor state, i.e., the set of operating conditions that define the system at present, based on the data available in the homogenized few-group cross-section library, a multi-parameter interpolation is necessary. On the other hand, homogenized parametric sections exhibit very strong variations, requiring high order interpolating polynomials for their representation. An example is provided in Fig. 1.3, showing the homogenized cross section of gadolinium 157, as a function of burn-up, which underwent energy condensation over $H = 2$ groups. A strong variation is observed over the whole fuel cycle, which can be justified by noting that the homogenized cross section does not depend only on the aforementioned set of parameters, \mathbf{p} , but, due to

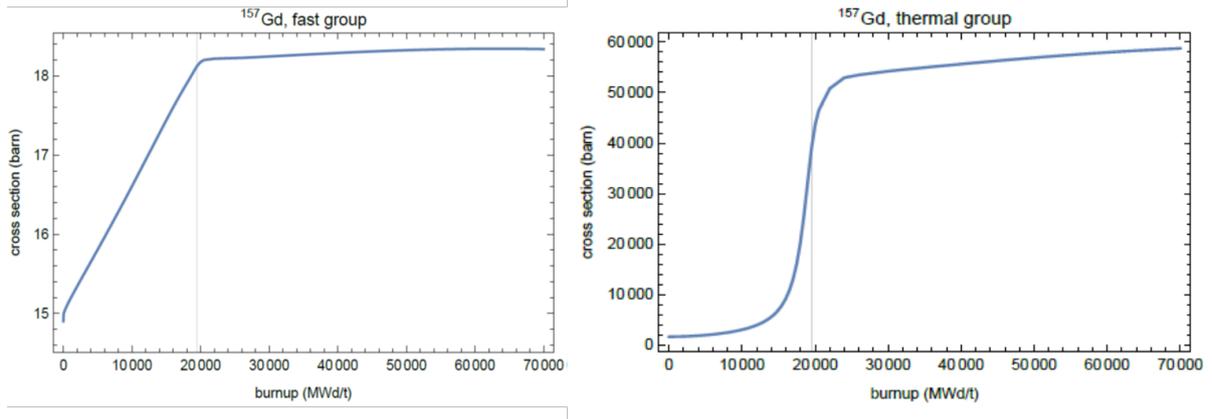


Figure 1.3: Two-group homogenized absorption cross section of gadolinium 157 vs burn-up. Figures from: I. Zmijarevic, "Some aspects of creating and using multi-parameter cross section library", DES/ISAS/DM2S/SERMA/LLPR/NT/2022-70741/A, Courtesy of Igor Zmijarevic.

flux-weighting, on the entire reactor operation. In other terms,

$$\sigma_{hom}^{c,h} = \frac{\langle \sigma(\mathbf{p}), \varphi \rangle_{c,h}}{\langle 1, \varphi \rangle_{c,h}} \quad (1.1)$$

is the homogenized few-group cross section, obtained by integration over a coarse spatial/energy mesh, with $\langle \cdot, \cdot \rangle_{c,h} = \int_{D_c} d^3r \int_{\Delta E_h} dE$. Its elementary variation about a point (\mathbf{p}_0, ϕ_0) is defined by

$$\delta\sigma_{hom} = (\nabla_{\mathbf{p}}\sigma_{hom})|_{(\mathbf{p}_0, \phi_0)} \cdot d\mathbf{p} + \left(\frac{\partial\sigma_{hom}}{\partial\varphi} \right)|_{(\mathbf{p}_0, \phi_0)} d\varphi, \quad (1.2)$$

where the first term reads as

$$(\nabla_{\mathbf{p}}\sigma_{hom})|_{(\mathbf{p}_0, \phi_0)} \cdot d\mathbf{p} = \sum_m \left(\frac{\partial\sigma_{hom}}{\partial p_m} \right)|_{(\mathbf{p}_0, \phi_0)} dp_m. \quad (1.3)$$

Several interpolation nodes are required, in order to reconstruct the dependence of the homogenized cross sections on the full set of operating parameters. The estimated cross sections are affected by numerous sources of error, of difficult estimation, including the thermal-hydraulics feedback and the assembly environment.

The present thesis work proposes a novel simulation pathway for coupled neutronics/thermal-hydraulics calculations, in exact three-dimensional geometry, without cross section homogenization. In particular, focus is given to the solution of the neutron transport equation by means of the Integro-Differential Transport code (IDT) of APOLLO3®. IDT solves the multigroup linear Boltzmann equation by discrete ordinates combined with angular-dependent probability matrices, with P_N scattering expansion. More precisely, IDT applies *short characteristics* to either 2D or 3D Cartesian nodes, lodging heterogeneous cylinders, in a modular XYZ frame. Neighbouring *Heterogeneous Cartesian Cells* (HCCs) are coupled by piece-wise linear currents, which are transmitted from one cell to another, by spatial sweeping. Similarly to the method of current-coupled collision probabilities (CCCP), short characteristics remain finite, when the optical approaches zero, thus allowing for accurate representation of empty media. Unlike other codes based on short characteristics, IDT does not approximate the angular flux on the inner surfaces

of the HCC, but just on the interface grid.

The current version of APOLLO3® IDT suffers from several limitations, which may prevent or restrain its application to reactor core calculations. In particular,

- Prior studies on the code have shown poor numerical accuracy for multiplying systems with large gradients, induced by strong absorbers or high-diffusive media;
- APOLLO3® IDT is not able to model unstructured /non-conformal/non-extruded meshes and its practical use is limited to pressurized water reactor (PWR) geometries;
- The spatial domain is discretized by a conformal Cartesian mesh. The control rods insertion/withdrawal would require fine axial planes, in order to represent the interface between the control rod tip and the water below. Such a choice is impractical and may lead to large simulation time. The alternative is to perform cross section homogenization, by volume or (preferably) flux-weighting techniques, which is a common choice in traditional nodal solvers. If such a model is put in practice, large discontinuities are observed in the first derivative of the control-rod reactivity worth, causing large discrepancies between the computed multiplication factor and its actual value;
- The memory pressure and computational cost increase rapidly, as the number of regions becomes larger, limiting the application of CCCP-based methods to large systems, with several thousands of media. This is justified by observing that the memory occupied by the *collision probability matrices* scales linearly with the number of HCCs, energy groups and S_N directions and quadratically with the number of regions per cell and with the number of components of the spatial basis.

The aim of the thesis is to implement a new version of IDT, capable of solving or drastically reducing the listed problems, in order to enable its practical use for coupled neutronics/thermal-hydraulics calculations on a variety of industrial systems. These may comprise reactors of different types and geometry.

The present manuscript includes four parts and is organized as follows.

In Part I, state-of-the-art numerical methods for the solution of the neutron transport equation are recalled.

In Chap. 3, a rapid overview on neutron cross sections is provided, together with the main hypotheses leading to the time-dependent integro-differential neutron transport problem. Two solution pathways, i.e., Monte Carlo and deterministic methods, are compared, underlining their advantages and drawbacks.

In Chap. 4, a closer focus on deterministic methods is proposed. Each phase-space variable, i.e., \mathbf{r} , Ω , E , is discretized. The multigroup formalism is introduced for the energy variable. The nested cycle, including power, thermal and inner iterations, is illustrated. The angle may be discretized by discrete ordinates or P_N expansion. Simplified P_N is also mentioned, as an approximate low-memory alternative to classic P_N . The spatial solution may be provided by a variety of numerical techniques. A subset of methods based on the integral form of the Boltzmann equation are discussed, in particular, the method of collision probabilities, the method

of (long) characteristics and the method of short characteristics. The final section of Chap. 4 provides an introduction to acceleration techniques. In this frame, a dedicated section presents the coarse mesh finite difference method.

Chapter 5 describes the major challenges of reactor core calculations. Three approaches are compared, namely the two-step strategy, 3D direct transport and 2D/1D fusion. Control-rod insertion/withdrawal simulations are discussed, focusing on the control-rod cusping effect. Four numerical workarounds are mentioned to reduce reactivity-worth wiggles, including flux-weighting techniques, the equivalence approach and two subplane-based methods.

In Chap. 6, an introduction to IDT is proposed. Provided that the code is equipped with several spatial discretization schemes, emphasis is placed on the application of short characteristics to heterogeneous Cartesian cells. A modular ray-tracing allows for the integration of the angular probability matrices. Finally, IDT relies on the method of domain decomposition, to lower the CPU time, by massive parallelism. A dedicated section provides few keynotes on its particular implementation in IDT.

Parts II-IV retrace the fundamental steps of this thesis work.

In Part II, details are provided on the new geometrical module of IDT.

Chap. 7 presents the new geometric capabilities of IDT, including the possibility to superimpose non-conformal Cartesian meshes onto the set of concentric cylinders, model off-centered rings, design non-Cartesian components (e.g., hexagonal patterns) and simulate horizontal cylinders. The new version of IDT can describe unstructured/non-conformal/non-extruded geometries, based on the Tuple-to-Region Map (TRM) and a combinatorial ray-tracing module. The former allows to customize the spatial discretization, by establishing an association between a tuple of 4 integers (i, j, k , corresponding to discrete steps along the $x/y/z$ axis, and r , to identify each cylinder) and a computational region, which is represented by a single integer number. On the other side, the new ray-tracing module is able to combine the trajectory intersection with a defined set of geometric patterns, comprising Cartesian bands and concentric rings. The numerical accuracy is tested by 2D/3D case studies, comparing IDT with TRIPOLI4® continuous energy Monte Carlo and APOLLO3® TDT MOC.

In Chap. 8, the application of IDT to control-rod movement is discussed. By taking advantage of the new geometric capabilities, which introduce the possibility to refine the local HCC discretization and generate non-conforming XYZ meshes, control-rod movements are simulated without resorting to cross-section homogenization. The HCCs hosting a control rod are finely discretized, to allow for the insertion of absorbing material by discrete steps, while coarser axial discretizations are used for the surrounding cells. This way the heterogeneous nature of each node is preserved, with limited increase in the CPU time.

Part III is dedicated to the advancements on the method of short characteristics (MOSC).

In Chap. 9, a novel asymptotic method permits to lower the memory imprint of the CP matrices, by reducing the number of computed cells. All HCCs sharing the same geometry and having 'similar' cross sections, for each energy group and computational region, are seen as a 'perturbation' of a 'reference' HCC. This entails that the CP coefficients are calculated only

for a subset of ‘unperturbed’ physical cell configurations, with a consequent reduction of the CPU time and of the memory occupation of the CP matrices. The solution of the transport equation can be reconstructed by a Neumann series development, if the perturbation is ‘small enough’ with respect to the reference configuration. The series can be computed by consecutive spatial sweeps, using the ‘unperturbed’ CP matrices, applied to successive updates of the neutron source. The overall CPU time is generally larger than standard MOSC, due to the extra cost of Neumann iterations, within the inner loop. According to our preliminary studies, this technique is particularly suited to self-shielded lattices or evolution calculations, to relieve the memory pressure of CP-based solvers.

In Chap. 10, an analysis is proposed for CP coefficients properties. First, the CPU time is reduced by piece-wise Taylor series expansion of exponential-like functions, depending on the optical path and appearing in the definition of the CP matrices. The coefficients of the polynomials are precomputed parameters, which are stored in memory at the beginning of each simulation. If the optical path exceeds a certain (predefined) quantity, the exponentials approach zero and the CP coefficients reduce to simpler expressions. Second, a set of conservation relations, relating all incoming and outgoing degrees of freedom, allows to check the CP matrices up to double precision. Finally, the application of symmetry and reciprocity relations is discussed, in order to lower the memory imprint and runtime of the CP coefficients, by reducing the number of stored directions, for linear short characteristics.

In Chap. 11, the neutron source and the interface angular flux are approximated by piece-wise parabolic polynomials. Previous studies have proven that the bilinear terms do not allow to significantly coarsen the spatial discretization, thus they have been neglected. The presented analysis shows that incomplete parabolic short characteristics (PSC) permit a major reduction of the number of regions, while preserving good numerical accuracy. Nevertheless, the memory occupation is still larger than linear short characteristics, as the second-order terms do not verify the aforementioned symmetry relations. Hence, a hybrid numerical approach, using piece-wise linear developments on the radial plane and a piece-wise parabolic expansion along the z-axis has been tested in the frame of PSC, in order to recover the prior radial symmetry transformations and, in the meantime, reduce the number of axial steps.

Once addressed the major limits of the past version of the code, IDT is coupled to THEDI (THErmohydraulique Diphaseique), App. A, and XSTOOL (Cross-Section TOOL), for multiphysics calculations, in Part IV.

In Chap. 12, a novel numerical pathway is proposed for coupled neutronics/thermal-hydraulics (N/TH) simulations, in exact tridimensional geometry, without cross-section homogenization. The thermal feedback is included within the power iteration loop. A user-defined parameter allows to weaken the N/TH coupling, by executing THEDI every \mathcal{N} outer iterations. For the time being, thermal-hydraulics is solved under simplified hypotheses, i.e., stationary monophasic flow, with independent sub-channels. A python script allows to transfer data from IDT to THEDI and, based on the new temperature map, updates the input file for XSTOOL. If the temperature distribution has undergone significant change with respect to the prior multiphysics iteration, i.e., $\delta T^{(e)}(\mathbf{r}) > \epsilon$, with $\delta T^{(e)}(\mathbf{r}) = T^{(e)}(\mathbf{r}) - T^{(e-\mathcal{N})}(\mathbf{r})$, e being the outer iteration index and ϵ a given tolerance, XSTOOL is executed for cross-section interpolation. Depending on the variation of the macroscopic cross sections, CP matrices update may not be necessary. If this is the

case, the 'perturbation' is simply updated and Neumann iterations permit to reconstruct the angular flux. Note that, in this framework, critical-leakage iterations are not required. Moreover, effective microscopic cross sections show milder dependence on the operational parameters than homogenized nuclear data (Eq. 1.1-1.3), and, as such, open the way to a reduction of the number interpolation nodes (Fig. 1.4).

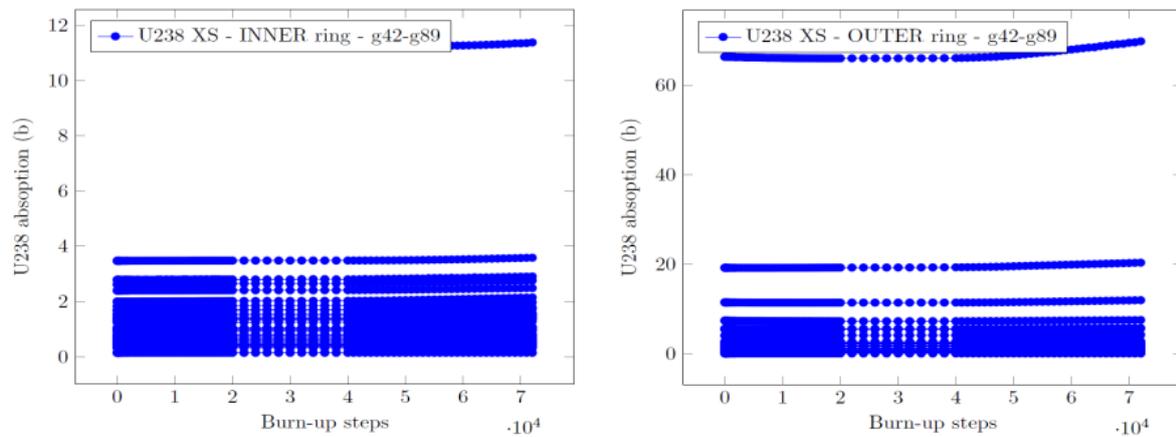


Figure 1.4: $\sigma_{U-238}^{g,in}$ and $\sigma_{U-238}^{g,out}$ vs burn-up, for $g \in [42, 89]$, where *in/out* indicate the inner/outer ring of the fuel cell.

LIST OF PUBLICATIONS

F. Filiciotto, E. Masiello, S. Lapuerta-Cochet, R. Lenain, "New modelling capabilities in IDT", *PHYSOR 2022: International conference on physics of reactors*, United States, 2022.

F. Filiciotto, E. Masiello, S. Lapuerta-Cochet, R. Lenain, "Nonconforming Three-Dimensional Model for PWR Control Rod Movements Without Homogenization and Cusping Effect", *Nuclear Science and Engineering*, 2023.

E. Masiello, F. Filiciotto, S. Lapuerta-Cochet, R. Lenain, "Asymptotic Expansion of the Angular Flux Applied to Discrete-Ordinates Source Iterations in Lattice Depletion Calculations", *Nuclear Science and Engineering*, 2023.

F. Filiciotto, E. Masiello, S. Lapuerta-Cochet, R. Lenain, "Incomplete Parabolic Short Characteristics in IDT", *International Conference on Mathematics and Computational Methods Applied to Nuclear Science and Engineering, M&C 2023*, Canada, 2023.

2 - RÉSUMÉ EN FRANÇAIS

De tous les grands défis qui nous attendent, la crise climatique est probablement la plus urgente et nécessite des solutions immédiates. Un nombre croissant de pays se sont déjà engagés à réduire à zéro leurs émissions de gaz à effet de serre d'ici à 2050. Dans ce cadre, des institutions telles que l'Agence internationale de l'énergie (AIE) ont proposé des voies durables pour une transition énergétique techniquement et économiquement réalisable, [1], indiquant que le nucléaire et l'hydroélectricité sont essentiels pour la production d'énergie à faible teneur en carbone. Le 10 février 2023, le président de la République française a annoncé la construction de 6 réacteurs pressurisés européens (EPR) de nouvelle génération, ainsi que de 8 réacteurs supplémentaires à l'étude, afin d'atteindre 25 GW_e de nouvelle capacité nucléaire installée, d'ici le milieu du siècle. Il semble évident que le développement de cette technologie jouera un rôle clé dans les années à venir.

Un réacteur à fission est un système capable d'accueillir une chaîne de réaction de fission contrôlée et auto-entretenu. Une réaction de fission est un processus complexe, initié par une particule neutre. Plus précisément, un neutron est envoyé sur un noyau fissile, qui se divise en deux fragments plus légers, avec la libération de neutrons supplémentaires et de rayons γ prompts. Un défaut de masse Δm d'environ 10% de la masse du noyau cible est observé. Une grande partie du Δm est convertie en énergie cinétique des noyaux fils ($\sim 170 \text{ MeV}$), tandis que de plus petites quantités apparaissent sous forme d'énergie cinétique des neutrons secondaires (quelques MeV chacun) et des photons γ (plusieurs MeV). Globalement, l'énergie libérée par un événement de fission est de plusieurs centaines de milliers de fois supérieure à celle généralement libérée lors d'une réaction chimique. Les noyaux fils sont des particules chargées et perdent de l'énergie par collisions successives avec les atomes environnants, entraînant une augmentation de la température du combustible nucléaire. La grande majorité des réacteurs nucléaires est basée sur des fissions induites par des neutrons thermiques, c'est-à-dire des neutrons en équilibre thermique avec le milieu. Le ralentissement des neutrons d'une énergie élevée ($\sim \text{MeV}$) à des fractions de eV peut être réalisé au moyen d'un matériau à faible numéro atomique, c'est-à-dire un modérateur. La plupart des réacteurs nucléaires utilisent de l'eau à cette fin, qui sert également de liquide de refroidissement pour réduire la température du combustible nucléaire. L'échange de chaleur se fait au niveau des générateurs de vapeur. La production de vapeur, qui entraîne une turbine, permet de générer un courant alternatif qui est injecté dans le réseau électrique.

La sûreté de fonctionnement et les performances sont des points clés à respecter dans la conception d'un réacteur à fission. Plusieurs facteurs entrent en jeu, l'un des plus décisifs étant la prédiction de la distribution des neutrons. L'équation fondamentale décrivant la densité neutronique en fonction de la position, de l'angle, de l'énergie et du temps, c'est-à-dire $n(\mathbf{r}, \boldsymbol{\Omega}, E, t)$, est l'équation de transport des neutrons. Elle peut être dérivée de l'équation de Boltzmann, en négligeant les collisions entre neutrons, qui se produisent rarement dans les systèmes multiplicatifs réels. Les neutrons peuvent subir des réactions de différents types, notamment la fission, la capture stérile, la diffusion, les réactions $(n, 2n)$, pour n'en citer que quelques-unes, en fonction du noyau cible et de l'énergie incidente du neutron. Les interactions neutroniques sont intrinsèquement stochastiques et ne peuvent être décrites qu'en termes de sections transver-

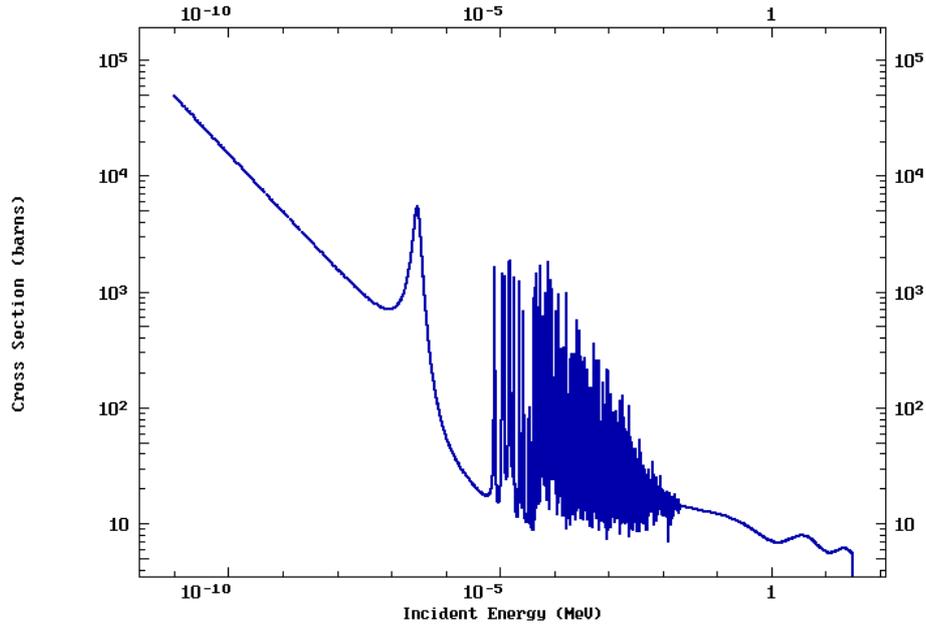


Figure 2.1: Section efficace microscopique totale du plutonium 239.
 From <https://www-nds.iaea.org/exfor/endl.htm>, Evaluated Nuclear Data File (ENDF).

sales microscopiques, σ , qui peuvent être représentées comme la section transversale présentée par le noyau cible lorsqu'un neutron est sur le point d'entrer en collision. Dans un interval d'énergie, compris entre quelques eV et des centaines de keV pour les nucléides lourds, les sections efficaces microscopiques présentent de fortes variations en fonction de la vitesse du neutron, connues sous le nom de résonances. Ce comportement affecte fortement l'évolution du flux neutronique, défini comme le produit entre la densité et la vitesse des neutrons, c'est-à-dire $\varphi = n \cdot v$, qui, à son tour, présentera des variations abruptes. La probabilité d'interaction des neutrons par unité de longueur dépend linéairement de la section efficace microscopique et de la densité nucléaire, N . Le produit des deux définit la section transversale macroscopique, $\Sigma = N \cdot \sigma$.

Comme indiqué, les réactions nucléaires libèrent une quantité considérable d'énergie, qui affecte la distribution de la température de tous les composants et, en particulier, du combustible nucléaire et du modérateur. En retour, l'augmentation de la température a deux conséquences. La première est liée à l'agitation thermique et porte le nom d'effet Doppler. En pratique, en raison du mouvement relatif entre le projectile et la cible, les résonances de la section transversale s'élargissent, en augmentant la probabilité qu'un neutron disparaisse à la suite d'un événement d'absorption. Le second effet concerne principalement la densité du modérateur. En raison de l'augmentation de la température, le nombre de cibles légères par unité de volume diminue et, par conséquent, la modération baisse. Comme la plupart des réactions de fission sont provoquées par des neutrons thermiques, la population de neutrons diminue. Cette simple analyse permet de comprendre qu'un réacteur nucléaire est un système multiphysique complexe, où plusieurs processus, en particulier la neutronique, la thermohydraulique et les performances du combustible, présentent de fortes interdépendances. D'une manière générale, un problème multiphysique peut être abordé de deux manières différentes, c'est-à-dire par un processus monolithique ou, au contraire, par une stratégie séparée, [2]. La première résout l'ensemble des équations d'équilibre simultanément à chaque étape, tandis que la seconde analyse chaque processus physique en soi, en supposant que la solution de tous

les autres problèmes physiques est fixée. La seconde approche nécessite plusieurs itérations entre les deux "monophysiques", où chaque processus fournit un ensemble de données pour alimenter les autres, afin de simuler la rétroaction correspondante. Les calculs de cœur du réacteur mettent généralement en œuvre la stratégie séparée, en résolvant, d'une part, l'équation de transport des neutrons et, d'autre part, la thermohydraulique et la thermomécanique du combustible, par des modules dédiés ou des codes couplés, en effectuant des itérations successives entre les "monophysiques", afin de faire converger la solution multiphysique.

L'équation de transport des neutrons peut être résolue soit par des méthodes déterministes, soit par des méthodes Monte Carlo, [3]. Ces dernières sont mieux adaptées aux problèmes de grande complexité géométrique, pour évaluer des quantités intégrales, telles que la dose absorbée, dans les calculs de radioprotection, ou la période du réacteur, dans les évaluations de sûreté-criticité. Inversement, l'évaluation des quantités différentielles, telles que le flux neutronique, est généralement mieux traitée par les méthodes déterministes, car elles sont plus efficaces en termes de calcul et prennent moins de temps. Bien qu'elles soient plus précises en termes de représentation de l'énergie, les simulations Monte Carlo sont peu pratiques si l'on considère l'ensemble du cycle du réacteur.

Ces dernières années, la croissance exceptionnelle de la puissance de calcul a rendu l'application des techniques de transport direct 3D plus attrayante pour les calculs de cœur du réacteur. Un exemple est fourni par le solveur de transport OpenMOC à coordonnées discrètes [4], qui applique la méthode des caractéristiques dans des géométries 3D réelles et met en œuvre une décomposition de domaine échelonnable pour une parallélisation massive. En général, le transport de neutrons en 3D dans le cœur entier montre une grande stabilité de convergence et une grande précision numérique. Cette approche est particulièrement adaptée aux applications aux cœurs de réacteurs avec de fortes hétérogénéités axiales, et, en général, présente un meilleur accord avec les solutions de référence Monte Carlo à énergie continue. Malgré les progrès réalisés en termes d'hardware, les solutions de haute fidélité pour le transport "full core" sont encore irréalisables pour les simulations industrielles du cycle du réacteur. La grande majorité des analyses commerciales s'appuie sur la stratégie dite *en deux étapes*, basée sur la réduction de la dimensionnalité, par homogénéisation spatiale et condensation en énergie, afin de limiter l'effort de calcul ainsi que les ressources nécessaires pour exécuter la simulation.

Dans un premier temps, l'équation de transport des neutrons est résolue avec une énergie et une discrétisation spatiale affinées, pour un ensemble de motifs répétitifs, par exemple les assemblages de combustible, avec des conditions limites simplifiées. Étant donné que pour un grand nombre de réacteurs industriels, la majeure partie de l'hétérogénéité est concentrée dans le plan radial, les assemblages de combustible sont supposés être bidimensionnels. Dans un premier temps, les échanges mutuels de courant neutronique entre les assemblages sont négligés, en imposant des conditions limites de réflexion. La solution spatiale de l'équation de transport des neutrons est généralement dérivée en utilisant soit les caractéristiques longues 2D, soit les probabilités de collision (avec des courants d'interface). La fuite neutronique au bord du réseau est incluse artificiellement. En supposant que le réacteur est *critique*, c'est-à-dire qu'il est capable de supporter une chaîne de réaction de fission auto-entretenu, un modèle approximatif est utilisé, où le *facteur de multiplication effectif* (c'est-à-dire le rapport entre la production et la disparition des neutrons, k_{eff}) est forcé à 1. En pratique, le flux neutronique est représenté comme le produit d'une distribution "macroscopique", dépendant de

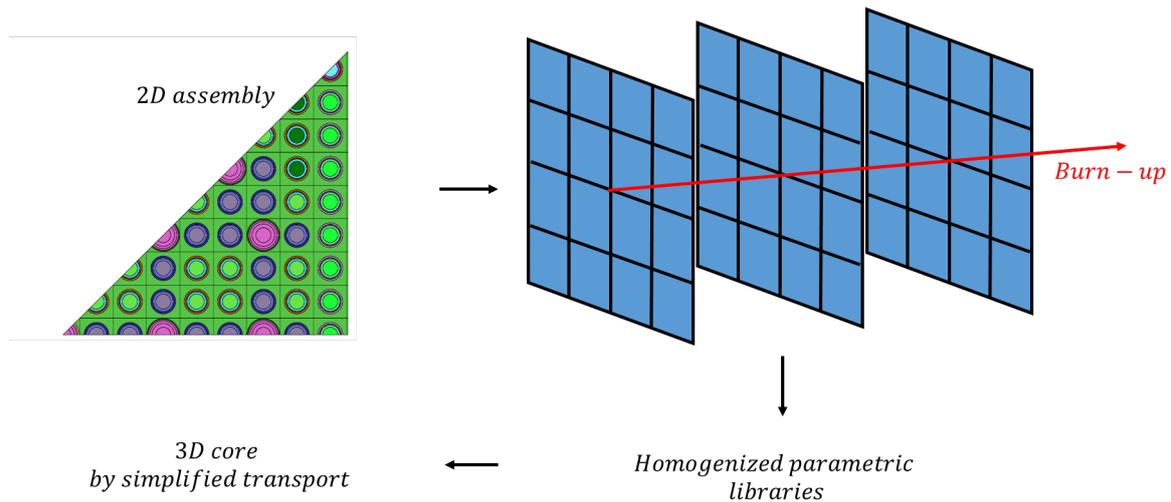


Figure 2.2: Schéma de calcul en deux étapes.

la seule position, et du spectre fin, dépendant de l'ensemble des variables indépendantes, obtenu en résolvant l'équation de transport avec des conditions aux limites de réflexion. Le flux "macroscopique" est supposé résoudre une équation de Laplace, où un paramètre critique (le "buckling", B^2) est recherché par itérations successives. Comme le buckling représente, jusqu'au signe, la dérivée seconde de la distribution "macroscopique", cette procédure peut être représentée comme un ajustement itératif de la courbure du flux, afin de faire converger k_{eff} vers 1, [5]. La simulation sur réseau permet de produire une bibliothèque de sections efficaces à quelques groupes, avec des milieux homogénéisés. Ce calcul est répété pour diverses conditions de fonctionnement ou *paramètres*, qui peuvent inclure un ensemble de températures (par exemple, la valeur locale T , la température du combustible et la température du modérateur), la densité du modérateur, le type d'assemblage, le "burn-up" du combustible, la concentration en bore, etc.

Dans la deuxième étape, le flux de neutrons et les taux de réaction sont calculés sur l'ensemble du cœur du réacteur, en utilisant la base de données construite, qui contient les sections efficaces générées par le calcul de réseau. Le cœur du réacteur est modélisé comme un réseau tridimensionnel d'assemblages ou de cellules homogénéisés et est résolu par un transport simplifié, avec des groupes d'énergie condensés. Les fuites neutroniques sont prises en compte en imposant des conditions aux limites exactes à la frontière du réacteur. La solution peut être recherchée pour les transitoires du réacteur ou dans des conditions stationnaires. Un simple bilan neutronique, mettant en équation la disparition et la production de neutrons, est résolu pour chaque nœud grossier et groupe d'énergie condensée, où la fuite sur la frontière du nœud peut être représentée par la *loi de Fick*, reliant le courant neutronique à la valeur négative du gradient du flux, [6].

Malgré leur popularité et leur rôle incontesté dans l'industrie nucléaire, les calculs en deux étapes sont affectés par un certain nombre de limitations. Au niveau du réseau, les échanges nets entre les assemblages de combustible sont d'abord négligés, ce qui donne un facteur de multiplication qui peut être très différent de celui de l'ensemble du système. Cela est particulièrement vrai pour les réacteurs rapides, où les libres parcours moyens des neutrons ont tendance à être plus longs, entraînant un couplage plus fort entre les assemblages voisins, [7]. En supposant que le système est critique, un modèle de fuite critique au B^2 est nécessaire pour faire converger k_{eff} vers 1. Comme mentionné ci-dessus, l'une des principales approxi-

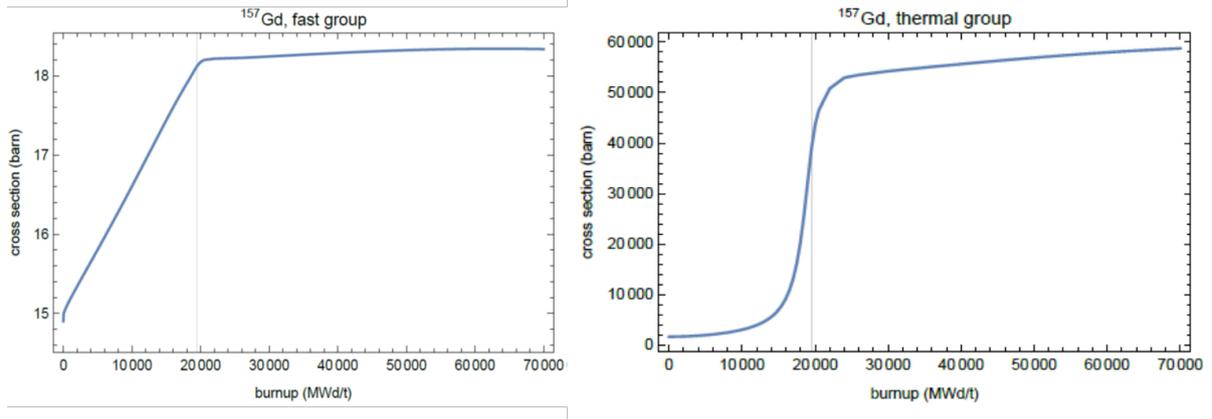


Figure 2.3: Section efficace d'absorption du gadolinium ^{157}Gd homogénéisée à 2 groupes en fonction du burnup. Figures de: I. Zmijarevic, "Some aspects of creating and using multi-parameter cross section library", DES/ISAS/DM2S/SERMA/LLPR/NT/2022-70741/A, Courtesy of Igor Zmijarevic.

mations de cette méthode consiste à supposer que le flux neutronique peut être représenté comme le produit d'une fonction macroscopique dépendant de la position et de la solution de transport détaillée 2D-infinie.

Un point crucial du calcul à deux niveaux réside dans la production de la bibliothèque paramétrique à quelques groupes, avec des cellules homogénéisées, pour un grand ensemble de *points d'état*. Chaque point d'état est défini par une collection de paramètres de fonctionnement, prenant des valeurs discrètes sur une grille multidimensionnelle. La bibliothèque générée est utilisée par un solveur de transport simplifié pour déduire la solution sur l'ensemble du réacteur. Plus précisément, dans la deuxième étape, les interdépendances de la neutronique, de la thermohydraulique et de l'évolution des isotopes sont prises en compte. Chaque simulation d'évolution est traitée comme un calcul en régime stationnaire, qui tient compte de la rétroaction de la thermohydraulique sur le transport des neutrons, ainsi que de la concentration en bore et de la position des barres de contrôle qui rendent le système critique. Afin de déduire les sections transversales pour l'état actuel du réacteur, c'est-à-dire l'ensemble des conditions d'exploitation qui définissent le système à l'heure actuelle, sur la base des données disponibles dans la bibliothèque de sections transversales homogénéisées à quelques groupes, une interpolation multiparamétrique est nécessaire. D'autre part, les sections paramétriques homogénéisées présentent de très fortes variations, nécessitant des polynômes d'interpolation d'ordre élevé pour leur représentation. Un exemple est fourni dans la Fig. 2.3, montrant la section efficace homogénéisée du gadolinium ^{157}Gd , qui a subi une condensation d'énergie sur $H = 2$ groupes, en fonction du burnup. Une forte variation est observée sur l'ensemble du cycle du combustible, ce qui peut être justifié en notant que la section transversale homogénéisée ne dépend pas seulement de l'ensemble des paramètres susmentionnés, \mathbf{p} , mais, en raison de la pondération par le flux, de l'ensemble du fonctionnement du réacteur. En d'autres termes,

$$\sigma_{hom}^{c,h} = \frac{\langle \sigma(\mathbf{p}), \varphi \rangle_{c,h}}{\langle 1, \varphi \rangle_{c,h}} \quad (2.1)$$

est la section efficace homogénéisée à quelques groupes, obtenue par intégration sur un maillage spatial/énergétique grossier, avec $\langle \cdot, \cdot \rangle_{c,h} = \int_{D_c} d^3r \int_{\Delta E_h} dE$. Sa variation élémentaire au-

tour d'un point (\mathbf{p}_0, ϕ_0) est définie par :

$$\delta\sigma_{hom} = (\nabla_{\mathbf{p}}\sigma_{hom})\Big|_{(\mathbf{p}_0, \phi_0)} \cdot d\mathbf{p} + \left(\frac{\partial\sigma_{hom}}{\partial\varphi}\right)\Big|_{(\mathbf{p}_0, \phi_0)} d\varphi, \quad (2.2)$$

où le premier terme s'écrit comme suit,

$$(\nabla_{\mathbf{p}}\sigma_{hom})\Big|_{(\mathbf{p}_0, \phi_0)} \cdot d\mathbf{p} = \sum_m \left(\frac{\partial\sigma_{hom}}{\partial p_m}\right)\Big|_{(\mathbf{p}_0, \phi_0)} dp_m. \quad (2.3)$$

Plusieurs nœuds d'interpolation sont nécessaires pour reconstruire la dépendance des sections transversales homogénéisées par rapport à l'ensemble des paramètres de fonctionnement. Les sections transversales estimées sont affectées par de nombreuses sources d'erreur, d'estimation difficile, y compris la rétroaction thermique-hydraulique et l'environnement d'assemblage.

Le présent travail de thèse propose une nouvelle méthode de simulation pour les calculs couplés de neutronique/thermique-hydraulique, dans une géométrie tridimensionnelle exacte, sans homogénéisation des sections efficaces. En particulier, l'accent est mis sur la solution de l'équation de transport des neutrons au moyen du code de transport intégral-différentiel (IDT) d'APOLLO3®. IDT résout l'équation de Boltzmann linéaire multigroupe par des ordonnées discrètes combinées à des matrices de probabilité dépendant de l'angle, avec une expansion P_N de la sections de diffusion. Plus précisément, IDT applique des *caractéristiques courtes* à des nœuds cartésiens $2D$ ou $3D$, hébergeant des cylindres hétérogènes, dans une géométrie modulaire. Les *cellules cartésiennes hétérogènes* (HCC) voisines sont couplées par des courants linéaires par morceaux, qui sont transmis d'une cellule à l'autre par un balayage spatial. De manière similaire à la méthode des probabilités de collision couplées aux courants d'interface (CCCP), les caractéristiques courtes restent finies lorsque le chemin optique s'approche de zéro, ce qui permet une représentation précise des milieux vides. Contrairement à d'autres codes basés sur les caractéristiques courtes, IDT n'approxime pas le flux angulaire sur les surfaces internes du HCC, mais seulement sur la grille d'interface.

La version actuelle d'APOLLO3® IDT a plusieurs limitations, qui peuvent empêcher ou restreindre son application aux calculs du cœur du réacteur. En particulier,

- Des études antérieures sur le code ont montré une faible précision numérique pour les systèmes multiplicatifs avec de grands gradients, induits par des milieux fortement absorbants ou fortement diffusives ;
- APOLLO3® IDT n'est pas capable de modéliser des maillages non structurés / non conformes / non extrudés et son utilisation pratique est limitée aux géométries des réacteurs à eau pressurisée (PWR) ;
- Le domaine spatial est discrétisé par un maillage cartésien conforme. L'insertion/ le retrait des barres de contrôle nécessiterait des plans axiaux fins, afin de représenter l'interface entre l'extrémité de la barre de contrôle et l'eau en dessous. Un tel choix n'est pas pratique et peut entraîner des temps de simulation importants. L'alternative consiste à effectuer une homogénéisation de la section transversale, par pondération par les volumes ou (de préférence) par le flux, ce qui est un choix courant dans les solveurs nodaux traditionnels. Si un tel modèle est mis en pratique, des discontinuités importantes sont observées dans la dérivée première de la courbe de réactivité, conduisant à des écarts significatifs entre le facteur de multiplication calculé et sa valeur réelle ;

- Le coût en mémoire et le temps de calcul augmentent rapidement à mesure que le nombre de régions augmente, limitant l'application des méthodes basées sur le CCCP aux grands systèmes, avec plusieurs milliers de milieux. Cela se justifie par le fait que la mémoire occupée par les *matrices de probabilité de collision* augmente linéairement avec le nombre de HCC, de groupes d'énergie et de directions S_N et quadratiquement avec le nombre de régions par cellule et avec le nombre de composantes de la base spatiale.

L'objectif de la thèse est de mettre en œuvre une nouvelle version du solveur IDT, capable de résoudre ou de réduire considérablement les problèmes énumérés, afin de permettre son utilisation pratique pour les calculs couplés neutroniques/thermohydrauliques sur une variété de systèmes industriels. Ceux-ci peuvent comprendre des réacteurs de différents types et de différentes géométries.

Le présent manuscrit comprend quatre parties et est organisé comme suit.

Dans la partie I, les méthodes numériques de pointe pour la résolution de l'équation de transport des neutrons sont rappelées.

Au chapitre 3, un aperçu rapide des sections efficaces des neutrons est fourni, ainsi que les principales hypothèses conduisant au problème de transport des neutrons intégral-différentiel dépendant du temps. Deux voies de solution, c'est-à-dire les méthodes Monte Carlo et les méthodes déterministes, sont comparées, en soulignant leurs avantages et leurs inconvénients.

Dans le chapitre 4, il est proposé de se concentrer davantage sur les méthodes déterministes. Chaque variable de l'espace des phases, c'est-à-dire \mathbf{r}, Ω, E , est discrétisée. Le formalisme multigroupe est introduit pour la variable énergie. La boucle imbriquée, comprenant les itérations de puissance, thermiques et internes, est illustrée. L'angle peut être discrétisé par des ordonnées discrètes ou par l'expansion P_N . Le P_N Simplifié (SP_N) est également mentionné en tant qu'approximation du P_N à plus faible occupation de mémoire. La solution spatiale peut être fournie par une diversité de techniques numériques. Un sous-ensemble de méthodes basées sur la forme intégrale de l'équation de Boltzmann est discuté, en particulier la méthode des probabilités de collision, la méthode des caractéristiques (longues) et la méthode des caractéristiques courtes. La dernière section du chapitre 4 présente une introduction aux techniques d'accélération. Dans ce cadre, une section spécifique présente la méthode CMFD (Coarse-Mesh Finite Difference).

Le chapitre 5 décrit les principaux défis posés par les calculs du cœur du réacteur. Trois approches sont comparées, c'est-à-dire la stratégie en deux étapes, le transport direct 3D et la fusion 2D/1D. Les simulations d'insertion/retrait des barres de contrôle sont examinées, en se concentrant sur l'effet de "rod cusping". Quatre techniques numériques sont mentionnées pour réduire les déviations de la valeur de la réactivité, notamment les techniques de pondération par le flux, l'approche d'équivalence et deux méthodes basées sur les sous-plans.

Dans le chapitre 6, une introduction à IDT est proposée. Le code étant équipé de plusieurs schémas de discrétisation spatiale, l'accent est mis sur l'application de caractéristiques courtes à des cellules cartésiennes hétérogènes. Un traçage modulaire permet l'intégration des matrices angulaire de probabilité. Enfin, IDT s'appuie sur la méthode de décomposition de domaine, pour réduire le temps CPU, grâce à un parallélisme massif. Une section dédiée fournit quelques

clés sur son implémentation particulière dans IDT.

Les parties II-IV retracent les étapes fondamentales de ce travail de thèse.

Dans la partie II, des détails sont fournis sur le nouveau module géométrique d'IDT.

Dans le chapitre 8, l'application d'IDT au mouvement des barres de contrôle est discutée. En utilisant les nouvelles capacités géométriques, qui permettent d'affiner la discrétisation locale des HCC et de générer des maillages XYZ non conformes, les mouvements des barres de contrôle sont simulés sans recours à l'homogénéisation. Les HCC hébergeant une barre de contrôle sont finement discrétisés, afin de permettre l'insertion de matériau absorbant par steps discrets, tandis que des discrétisations axiales plus grossières sont utilisées pour les cellules environnantes. De cette manière, la nature hétérogène de chaque nœud est préservée, avec une augmentation limitée du temps de calculs.

La partie III est consacrée aux progrès sur la méthode des caractéristiques courtes (MOSC).

Dans le chapitre 9, une nouvelle méthode asymptotique permet de réduire l'empreinte mémoire des matrices de probabilité, en réduisant le nombre de cellules calculées. Toute les HCC partageant la même géométrie et ayant des sections efficaces "similaires", pour chaque groupe d'énergie et région de calcul, sont considérée comme une "perturbation" d'une HCC "de référence". Cela signifie que les coefficients ne sont calculés que pour un sous-ensemble de configurations de cellules physiques "non perturbées", avec une réduction conséquente du temps CPU et de l'occupation de la mémoire des matrices de probabilité. La solution de l'équation de transport peut être reconstruite par un développement en série de Neumann, si la perturbation est "suffisamment petite" par rapport à la configuration de référence. La série peut être calculée par des balayages spatiaux consécutifs, en utilisant les matrices de probabilité "non perturbées", appliquées à des mises à jour successives de la source de neutrons. Le temps total de calculs est généralement supérieur à celui du MOSC standard, en raison du coût supplémentaire des itérations de Neumann, dans la boucle interne. D'après nos études préliminaires, cette technique est particulièrement adaptée aux réseaux autoprotégés ou aux calculs d'évolution, afin de soulager l'occupation de la mémoire des solveurs basés sur les probabilités de collision.

Au Chap. 10, une analyse est proposée pour les propriétés des coefficients de probabilité. Premièrement, le temps CPU est réduit par un développement par morceaux en série de Taylor de fonctions de type exponentiel, en fonction du chemin optique, apparaissant dans la définition des matrices angulaires de probabilité. Les coefficients des polynômes sont des paramètres précalculés, qui sont stockés en mémoire au début de chaque simulation. Si le chemin optique dépasse une certaine quantité (prédéfinie), les exponentielles se rapprochent de zéro et les coefficients se réduisent à des expressions plus simples. Deuxièmement, un ensemble de relations de conservation, reliant tous les degrés de liberté entrants et sortants, permet de vérifier les matrices angulaires de probabilité jusqu'à la double précision. Enfin, l'application des relations de symétrie et de réciprocité est discutée, afin de réduire l'empreinte mémoire et le temps de calcul des coefficients, en réduisant le nombre de directions stockées, pour des caractéristiques linéaires courtes.

Dans le Chap. 11, la source neutronique et le flux angulaire à l'interface sont approchés par des polynômes paraboliques par morceaux. Des études antérieures ont prouvé que les termes bilinéaires ne permettent pas de réduire significativement la discrétisation spatiale, ils ont donc été négligés. L'analyse présentée montre que les caractéristiques courtes paraboliques incomplètes (PSC) permettent une réduction importante du nombre de régions, tout en préservant une bonne précision numérique. Néanmoins, l'occupation mémoire est encore plus grande que les caractéristiques courtes d'ordre linéaire, car les termes du second ordre ne vérifient pas les relations de symétrie susmentionnées. Ainsi, une approche numérique hybride, utilisant des développements linéaires par morceaux sur le plan radial et une expansion parabolique par morceaux le long de l'axe z a été testée dans le cadre de PSC, afin de récupérer les transformations de symétrie radiale antérieures et, en même temps, de réduire le nombre de pas axiaux.

Une fois abordées les limites majeures de la version précédente du code, IDT est couplé à THEDI (THErmohydraulique DIphasique), App. A, et XSTOOL (Cross-Section TOOL), pour les calculs multiphysiques, dans la Partie IV.

Au Chap. 12, une nouvelle voie numérique est proposée pour les simulations couplées neutronique/thermohydraulique (N/TH), en géométrie tridimensionnelle exacte, sans homogénéisation de section efficace. La rétroaction thermique est inclus dans la boucle des itérations de puissance. Un paramètre défini par l'utilisateur permet d'affaiblir le couplage N/TH, en exécutant THEDI toutes les \mathcal{N} itérations externes. Pour l'instant, la thermohydraulique est résolue sous des hypothèses simplifiées, c'est-à-dire un écoulement monophasique stationnaire, avec des sous-canaux indépendants. Un script python permet de transférer les données d'IDT vers THEDI et, sur la base de la nouvelle nappe de température, met à jour le fichier d'entrée pour XSTOOL. Si la distribution de température a subi un changement significatif par rapport à l'itération multiphysique précédente, c'est-à-dire $\delta T^{(e)}(\mathbf{r}) > \epsilon$, avec $\delta T^{(e)}(\mathbf{r}) = T^{(e)}(\mathbf{r}) - T^{(e-\mathcal{N})}(\mathbf{r})$, e étant l'index d'itération externe et ϵ une tolérance donnée, XSTOOL est exécuté pour l'interpolation de section. En fonction de la variation des sections efficaces macroscopiques, la mise à jour des matrices angulaires de probabilité peut ne pas être nécessaire. Si tel est le cas, la "perturbation" est simplement mise à jour et les itérations de Neumann permettent de reconstruire le flux angulaire. A noter que, dans ce cadre, les itérations de fuite critique ne sont pas requises. De plus, les sections efficaces microscopiques montrent une dépendance plus légère aux paramètres opérationnels que les données nucléaires homogénéisées (Eq. 2.1-2.3), et ouvrent ainsi la voie à une réduction des nœuds d'interpolation (Fig. 2.4).

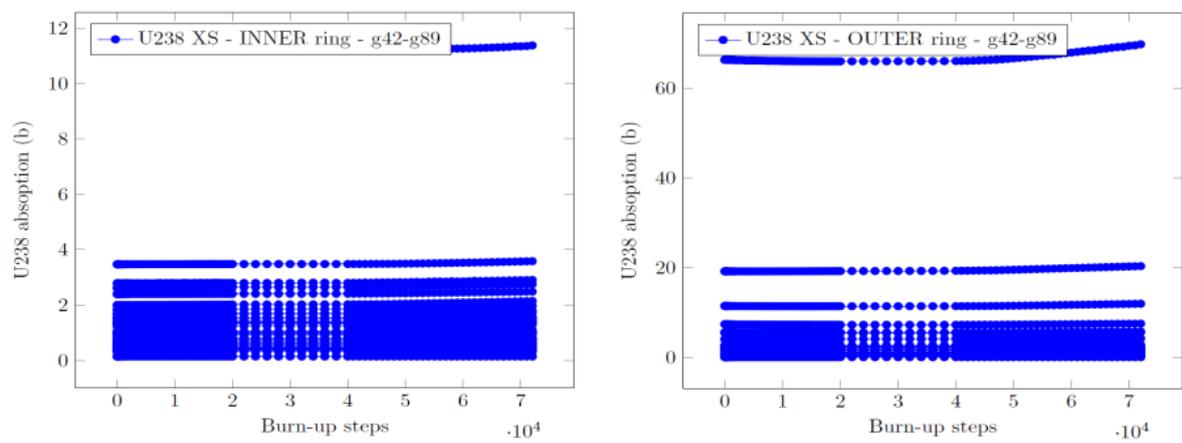


Figure 2.4: $\sigma_{U-238}^{g,in}$ et $\sigma_{U-238}^{g,out}$ en fonction du burnup, pour $g \in [42, 89]$, où *in/out* indique la couronne interne/externe de la cellule combustible.

Part I

SOLUTION OF THE BOLTZMANN EQUATION - STATE OF THE ART

3 - THE BOLTZMANN EQUATION

3.1 . CONTEXT

The history of nuclear power for civil applications dates back to 1947, with the NRX reactor in Chalk-River (Canada), and to 1955, with the Borax III reactor in Idaho (USA). Since then, several nuclear power plants have been designed and connected to the electrical grid, in order to serve the energy needs of tens of countries in the world. Currently, more than 400 fission reactors operate in 32 countries, including the USA, leading the ranking with 92 operating reactors, France (56), China (55), Russia (37), South Korea (25), and many others, for a total net capacity of about 378 GWe, [8]. Despite the accident at Fukushima in 2011, more than 50 reactors (~60 GW) are currently under construction, mostly in Asia, where China leads the ranking with 19 new units, [8].

The most common fission reactor type is the so-called **Pressurized Water Reactor** (PWR), which counts more than 300 operating units for energy production, as well as several hundreds more for naval propulsion, [9]. In a few words, the energy released by neutron-induced fission reactions on uranium 235 heats the water of a primary circuit, which in turns heats the water circulating in a secondary circuit, driving to steam production. The steam is responsible for the rotation of a set of turbines, whose mechanical energy is converted into electricity by an alternator, [9].

The fundamental particle which is responsible for reactor's functioning is the **neutron**, whose discovery, by James Chadwick, dates back to 1932. The study of the migration, collisions and evolution of the neutron population in the reactor core, which is addressed by **neutronics**, is the central problem of nuclear power plants design and safe operation.

The present chapter offers a brief overview of key aspects of neutron transport theory, focusing on the fundamental equation governing the time evolution and spatial distribution of the neutron population, namely the **Boltzmann** or **transport equation**. The discussion is organized as follows: in Sect. 3.2.1 and 3.2.2, the concept of cross section is introduced, together with the main neutron-matter reaction types; in Sect. 3.2.3, the neutron distribution is characterized in terms of particle density and angular flux; in Sect. 3.2.4, the neutron transport equation is derived; in Sect. 3.2.5, the boundary and initial conditions are recalled; finally, in Sect. 3.2.6, a comparison between Monte Carlo and deterministic methods is proposed, pointing out the advantages and drawbacks of both.

3.2 . NEUTRONICS

3.2.1 . Nuclear Reactions Induced by Neutrons

Let us illuminate a target of N nuclides per unit volume with a monodirectional beam of monoenergetic neutrons, with intensity $I [\frac{1}{cm^2s}]$. The target is orthogonal to the neutron direction and its width is equal to Δt . The expected number of nuclear reactions per unit surface and time, $R [\frac{1}{cm^2s}]$, reads as:

$$R = P \cdot I, \quad (3.1)$$

where $P = \sigma \cdot N \cdot \Delta t$ is the reaction probability, $\sigma [cm^2]$ being a proportionality coefficient, known as **microscopic cross section**. The microscopic cross section generally differs significantly from the geometric area of a nucleus, which may range from 10^{-30} to $10^{-26}cm^2$, depending on the target nuclide. For instance, for a (n, γ) reaction on $^{157}_{64}Gd$, where n is a neutron in the thermal range and $^{157}_{64}Gd$ is the isotope of gadolinium with mass number $A = 157$, the microscopic cross section reaches $4.9 \cdot 10^{-20}cm^2$. For historical reasons, a more natural unit for the microscopic cross section is the barn (b), where $1b = 10^{-24}cm^2$.

The product $\Sigma = \sigma \cdot N [\frac{1}{cm}]$ has a simple interpretation, as it represents the reaction probability per unit length, and is called **macroscopic cross section**. The reciprocal of the macroscopic cross section is the neutron **mean free path**, $\lambda [cm]$, which represents the average distance traveled by a neutron before collision.

Neutrons may interact with matter in different ways. For this reason, a **partial cross section** σ_j may be defined for each interaction type. Tab. 3.1 offers an overview of the main reaction types which take place in a fission reactor, [10]. A partial microscopic cross section is associated to each collision type. Some reactions may or may not have a threshold (abbr "Thr." in Tab. 3.1): for instance, neutrons of any energy may cause a fission reaction on a fissile nuclide; clearly, the same does not apply to fissionable nuclides. For the sake of clarity, the following notation has been adopted:

- Atomic Number, Z
- Mass Number, A
- Number of fission neutrons, ν
- Fission Products, P_1, P_2
- Reaction with threshold, \checkmark
- Reaction without threshold, $-$

Partial cross sections are additive, thus the sum over all interaction types simply returns the total cross section, i.e.,

$$\sigma_t = \sum_j \sigma_j. \quad (3.2)$$

Reaction type	Reaction	σ_j	Thr	Notation
Elastic scattering	$n + {}^A_ZX \rightarrow n + {}^A_ZX$	σ_e	–	(n, n)
Inelastic scattering	$n + {}^A_ZX \rightarrow n' + {}^A_ZX^*$	σ_i	✓	(n, n')
(n, xn)-reactions	$n + {}^A_ZX \rightarrow n_1 + \dots + n_x + {}^{A-1}_ZX$	$\sigma_{n,xn}$	✓	(n, xn)
Radiative capture	$n + {}^A_ZX \rightarrow \gamma + {}^{A+1}_ZX$	σ_c	–	(n, γ)
(n, α) reactions	$n + {}^A_ZX \rightarrow \alpha + {}^{A-4}_{Z-2}X$	σ_α		(n, α)
Fission	$n + {}^A_ZX \rightarrow P_1 + P_2 + \nu n + \gamma$	σ_f		(n, f)

Table 3.1: Main reaction types occurring in a fission reactor.

The same applies to macroscopic cross sections. In reactor physics, it is common practice to group the cross sections as follows [5]:

$$\sigma_s = \sigma_e + \sigma_i + \sum_{x \geq 2} \sigma_{n,xn}, \quad (3.3)$$

$$\sigma_a = \sigma_f + \sigma_c + \sigma_\alpha, \quad (3.4)$$

σ_s and σ_a being the scattering cross section, which accounts for elastic and inelastic scattering events as well as (n, xn)-reactions, and the absorption cross section, which addresses to all reactions involving the absorption of the incident neutron, respectively.

Cross sections strongly depend on the energy of the incident neutron. As shown in Figs. 3.1-3.2, cross sections display abrupt variations as a function of the energy, which take the form of sharp peaks, known as **resonances**, and correspond to distinct energy levels of the compound nucleus. One may also notice the peculiar $1/v$ behaviour, v being the projectile speed at relatively low energies. Heavy nuclides, ${}^{238}_{92}U$ typically exhibit very packed resonances in the epithermal range, i.e., from ~ 1 eV to 100 keV, whereas, for lighter nuclides, like hydrogen or oxygen isotopes, the nuclear energy structure is more loose, and the first resonance is observed at much higher energy (around 400 keV, for ${}^{16}_8O$, Fig.). Finally, heavier nuclides show narrower resonances, which represent one of the major matters of concern for deterministic calculations.

3.2.2 . Differential Scattering Cross Section and Fission Neutrons Distributions

In order to account for the dependence of the secondary neutron distribution on the incident neutron energy and direction in scattering events, let us introduce a probability density distribution, $p_s(\Omega', E' \rightarrow \Omega, E)$, such that

$$p_s(\Omega', E' \rightarrow \Omega, E) d^2\Omega dE \quad (3.5)$$

is the probability that a neutron with direction Ω and energy E is emitted in the differential element $d^2\Omega dE$, provided an incident neutron with direction Ω' and energy E' . For isotropic media, the scattering probability density does not depend separately on the incoming and outgoing angular directions, but only on the cosine of the deviation angle, namely $\mu = \Omega' \cdot \Omega$. A double differential cross section may be introduced, as follows, [5]:

$$\sigma_s(E' \rightarrow E, \mu) = \sigma_s(E) p_s(E' \rightarrow E, \mu). \quad (3.6)$$

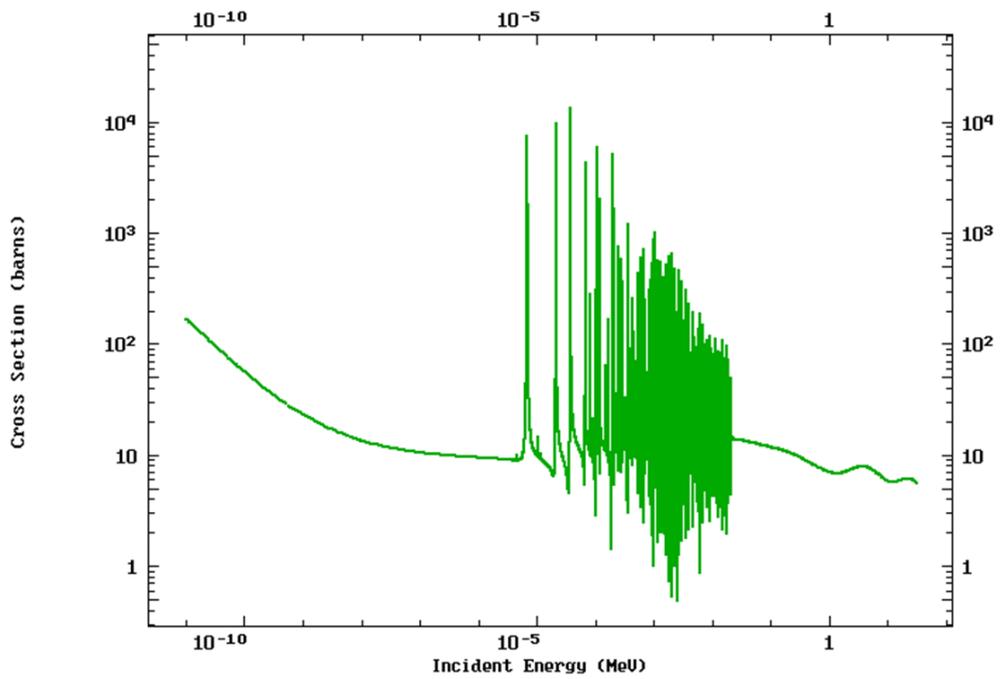


Figure 3.1: Total cross section of $^{238}_{92}\text{U}$. From <https://www-nds.iaea.org/exfor/endl.htm>, Evaluated Nuclear Data File (ENDF).

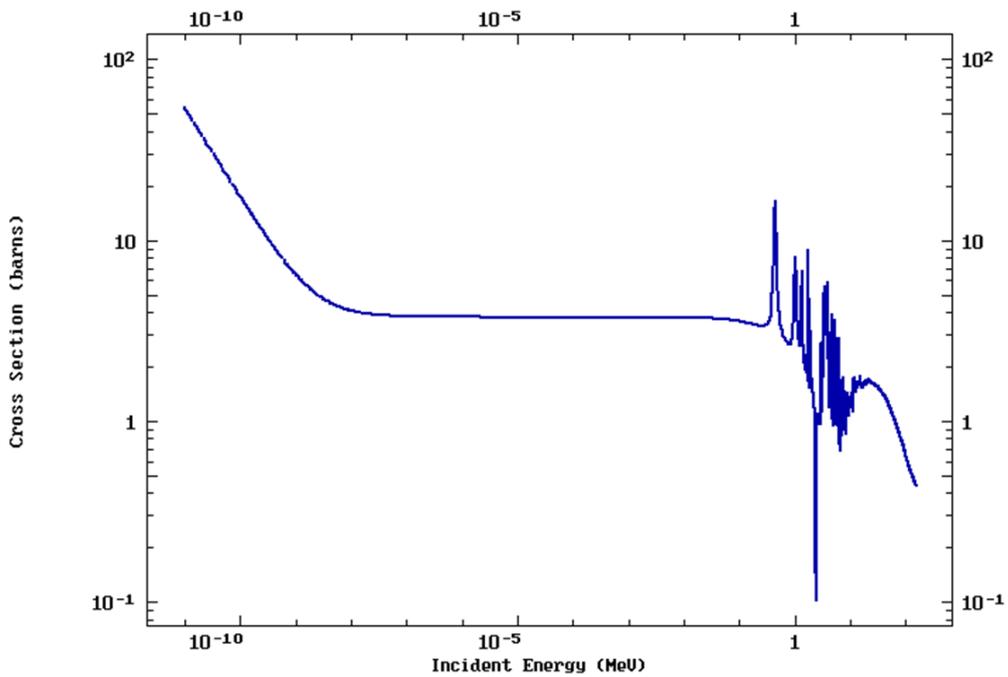


Figure 3.2: Total cross section of $^{16}_8\text{O}$. From <https://www-nds.iaea.org/exfor/endl.htm>, Evaluated Nuclear Data File (ENDF).

Eq. (3.6) can be easily extended to the macroscopic scattering cross section.

On the other hand, fission neutrons depend mildly on the direction of the incident neutron, thus fissions are generally assumed to be isotropic in the laboratory frame. The neutron multiplicity $\nu_f(E')$, i.e., the average number of neutrons emitted per fission event, and the fission neutron spectrum $\chi(E')$, i.e., the neutron energy distribution after a fission event, are assumed to depend only on the incoming and outgoing energy, respectively, [15]. Several functions have been proposed to fit the fission spectrum; a common choice is the Watt spectrum, which reads as

$$\chi^i = C(a_i, b_i) e^{-a_i E} \sinh \sqrt{b_i E}, \quad (3.7)$$

where a_i and b_i depend on the isotope i and C is a function of both a_i and b_i , [17].

3.2.3 . Phase-space Variables and Angular Flux

Despite the obvious difficulty to provide a general definition, fission reactor physicists usually concur that neutronics is the study of neutron travelling and collisions in matter, aimed determining the conditions for a self-sustaining chain reaction and the associated reaction rates distributions, [10]. Neutronics covers a broad range of applications, including criticality safety and kinetics, [11], perturbation theory and sensitivity analysis, [12], fuel cycle and multi-physics, [13], just to name a few.

Neutrons may be modelled as a statistical population of **density** $n(\mathbf{r}, \boldsymbol{\Omega}, E, t)$ [$\frac{1}{\text{cm}^3 \cdot \text{eV} \cdot \text{strad}}$], where $\mathbf{r}, \boldsymbol{\Omega}, E, t$ represent, respectively, the neutron position, angular direction, kinetic energy and time, [14]. The product $n(\mathbf{r}, \boldsymbol{\Omega}, E, t) d^3r d^2\Omega dE$ represents the number of neutrons around the point $\mathbf{r}, \boldsymbol{\Omega}, E$, within the phase-space element $d^3r d^2\Omega dE$, at time t . Provided that $v(E) = \sqrt{\frac{2E}{m}}$ is the neutron speed, the product $\varphi = nv$ [$\frac{1}{\text{cm}^2 \cdot \text{s} \cdot \text{eV} \cdot \text{strad}}$] defines the **angular flux**, which in turn depends on the whole set of phase-space variables. The angular flux is a mathematical definition, with no intuitive physical interpretation. Nevertheless, it is tightly related to the rate of collision events, and thus the reactor power distribution, as the number of interactions per unit time depends on both the particle density and speed, [5].

Let us consider a surface element dS , of normal unit-vector \mathbf{n}_s . The neutrons crossing dS between t and $t + dt$ are contained in a cylinder of base dS and generating lines parallel to $\boldsymbol{\Omega}$ and length equal to $v(E) \cdot dt$. Their number is defined by

$$n(\mathbf{r}, \boldsymbol{\Omega}, E, t) v(E) dt \boldsymbol{\Omega} \cdot \mathbf{n}_s dS. \quad (3.8)$$

The product $\mathbf{J} = n\mathbf{v}$, where $\mathbf{v} = v(E)\boldsymbol{\Omega}$ is the neutron velocity, represents the **angular current density**.

By integrating over all directions, one retrieves the scalar flux Φ and current density \mathbf{J} ,

$$\Phi(\mathbf{r}, E, t) = \int_{4\pi} \varphi(\mathbf{r}, \boldsymbol{\Omega}, E, t) d^2\Omega \quad (3.9)$$

$$\mathbf{J}(\mathbf{r}, E, t) = \int_{4\pi} \mathbf{J}(\mathbf{r}, \boldsymbol{\Omega}, E, t) d^2\Omega \quad (3.10)$$

Intuitively, the scalar flux $\Phi(\mathbf{r}, E, t)$ is the number of neutrons crossing a unitary sphere centered at \mathbf{r} , per unit energy and time.

It is common practice to split the scalar current in two contributions, i.e., a 'positive' term, Eq. (3.11), traversing the surface in the outward direction, and a 'negative' term, Eq. 3.12, traversing the surface in the inward direction:

$$J_+ = \int_{\Omega \cdot \mathbf{n}_s > 0} \mathbf{J}(\mathbf{r}, \Omega, E, t) \cdot \mathbf{n}_s d^2\Omega, \quad (3.11)$$

$$J_- = - \int_{\Omega \cdot \mathbf{n}_s < 0} \mathbf{J}(\mathbf{r}, \Omega, E, t) \cdot \mathbf{n}_s d^2\Omega, \quad (3.12)$$

where \mathbf{n}_s is the outward normal. Hence, the net current is provided by $J = J_+ - J_-$, [14].

3.2.4 . Neutron Transport Equation

In order to derive the neutron transport equation, a few hypotheses are introduced, [10]:

- The de Broglie wavelength reads as $\lambda_B = \frac{h}{p}$, where $h = 4.1356 \cdot 10^{-15} \text{ eV} \cdot \text{s}$ is the Planck constant and $p = mv$ is the module of the linear momentum. Both for thermal and fast neutrons, λ_B is much smaller than the inter-atomic distances, hence neutrons' wave nature can be neglected;
- The rest mass of a neutron ($\sim 1 \text{ GeV}$) is much larger than the maximum energy of the Watt spectrum ($\sim 20 \text{ MeV}$). As a consequence, fission reactor neutrons can be treated as non-relativistic particles;
- Neutrons undergo β -decay with a half life $T \sim 10 \text{ min}$. Nevertheless, as the neutron lifetime in a reactor is far lesser ($\sim \text{ms}$). Therefore, neutrons can be modelled as stable particles;
- Neutron-neutron collisions are negligible, as the average neutron density is of the order of 10^9 cm^{-3} , while the atomic density of a medium is of the order of 10^{23} cm^{-3} , leading to a linear formulation of the transport equation.

In conclusion, neutrons can be treated as rarefied gas of classical particles, which undergo stochastic collisions, with linear probability equal to the macroscopic cross section.

This said, let us choose a Lagrangian frame of reference, i.e., the control volume moves with respect to the laboratory frame, following the neutrons, [15], [16]. Dividing by the phase-space volume element $d^3r d^2\Omega dE$ and taking the limit as $\Delta t \rightarrow 0$, the balance equation reads as

$$\begin{aligned} \frac{dn(\mathbf{r}, \Omega, E, t)}{dt} + \Sigma_t(\mathbf{r}, E) v(E) n(\mathbf{r}, \Omega, E, t) = \\ \int_{4\pi} \int_0^\infty d^2\Omega' dE' \nu_s \Sigma_s(\mathbf{r}, \Omega', E' \rightarrow \Omega, E) v(E') n(\mathbf{r}, \Omega', E', t) + \\ \sum_i \frac{\chi^i(\mathbf{r}, E)}{4\pi} \int_{4\pi} \int_0^\infty d^2\Omega' dE' \nu_f^i \Sigma_f^i(\mathbf{r}, E') v(E') n(\mathbf{r}, \Omega', E', t), \end{aligned} \quad (3.13)$$

where

- Ω', E' are the incoming neutron direction and energy, respectively;
- the second term on the left hand side of the equation represents the total collision rate;
- the first term on the right hand side of the equation represents the rate of secondary neutrons produced by scattering interactions, where ν_s is the mean number of neutrons per diffusion collision and depends on both the position and the incident neutron energy;
- the second term on the right hand side of the equation represents the rate of secondary neutrons produced by fission, where $i = 1, \dots, N_f$, N_f being the number of fissile isotopes, χ^i is the Watt spectrum, and ν_f^i is the mean number of neutrons per fission and depends on both the position and the incident neutron energy.

The relation between the Lagrangian (or material) and Eulerian time derivatives is given by

$$\frac{d}{dt} = \frac{\partial}{\partial t} + v\mathbf{\Omega} \cdot \nabla. \quad (3.14)$$

Hence, by injecting $\varphi = nv$ and Eq. (3.14) into (3.13), one obtains the integro-differential transport equation,

$$\begin{aligned} & \frac{1}{v} \frac{\partial \varphi}{\partial t} + \mathbf{\Omega} \cdot \nabla \varphi + \Sigma_t(\mathbf{r}, E) \varphi(\mathbf{r}, \mathbf{\Omega}, E, t) = \\ & \int_{4\pi} \int_0^\infty d^2\Omega' dE' \nu_s \Sigma_s(\mathbf{r}, \mathbf{\Omega}', E' \rightarrow \mathbf{\Omega}, E) \varphi(\mathbf{r}, \mathbf{\Omega}', E', t) + \\ & \sum_i \frac{\chi^i(\mathbf{r}, E)}{4\pi} \int_{4\pi} \int_0^\infty d^2\Omega' dE' \nu_f^i \Sigma_f^i(\mathbf{r}, E') \varphi(\mathbf{r}, \mathbf{\Omega}', E', t). \end{aligned} \quad (3.15)$$

Eq. (3.14) is responsible for the appearance of $\mathbf{\Omega} \cdot \nabla \varphi$, i.e., the **streaming term**, and may be related to the number of neutrons escaping from a phase-space volume element, per unit time.

3.2.5 . Initial and Boundary Conditions

The solution space may be restricted to particular solutions by imposing initial and boundary conditions. The solution existence and uniqueness have been investigated by several authors and a rigorous treatment may be found in [18]. Eq. (3.15), together with albedo boundary conditions¹

$$\varphi(x^-, t) = \beta \varphi(x^-, t), \quad x^- \in \partial D^-, \quad (3.16)$$

and initial condition

$$\varphi(x, t = 0) = f(x), \quad x \in D, \quad (3.17)$$

forms a Cauchy problem, where

- $D = V \times [0, 4\pi] \times [0, \infty)$, with $V \subseteq \mathbb{R}^3$, is a phase space region,

¹The term *Albedo*, derived from the Latin word *albus* ('white'), has been introduced into optics by Lambert, in order to define the fraction of incident light reflected isotropically by a surface, [19]. The choice of albedo boundary conditions is common in reactor physics. In fact, they relate incoming and outgoing surface angular fluxes, acting as a neutron reflector surrounding the reactor core. From a physical standpoint, albedo boundary conditions allow simulating both the exiting neutrons leaving the multiplying system and the neutrons that are reflected into the reactor core.

- $\partial D^- = \{(\mathbf{r}, \boldsymbol{\Omega}) \in \partial V \times [0, 4\pi] : \boldsymbol{\Omega} \cdot \mathbf{n}_{\partial V}(\mathbf{r}) < 0\}$, $\mathbf{n}_{\partial V}(\mathbf{r})$ being the outward unit normal vector at $\mathbf{r} \in \partial V$,
- $\partial D^+ = \{(\mathbf{r}, \boldsymbol{\Omega}) \in \partial V \times [0, 4\pi] : \boldsymbol{\Omega} \cdot \mathbf{n}_{\partial V}(\mathbf{r}) > 0\}$,
- β is the **albedo operator** and reads as

$$\beta\varphi(x^+, t) = \int_0^\infty dE' \int_{\partial D^+} d^2\Omega' d^3r' \beta(\mathbf{r}', \boldsymbol{\Omega}', E' \rightarrow \mathbf{r}, \boldsymbol{\Omega}, E) \varphi(\mathbf{r}', \boldsymbol{\Omega}', E', t), \quad (3.18)$$

- f is a known function of $x = \{\mathbf{r}, \boldsymbol{\Omega}, E\}$ in D .

Vacuum and reflection boundary conditions may be derived from Eq. (3.16), setting the albedo operator β as indicated in 3.2.5 and 3.2.5.

Vacuum boundary conditions

Vacuum (or non-reentering) boundary conditions are obtained by setting β equal to zero. As a result, vacuum boundary conditions may be expressed as

$$\varphi(x^-, t) = 0, \quad x^- \in \partial D^-. \quad (3.19)$$

Reflective boundary conditions

Let us consider a surface Γ and suppose that $\boldsymbol{\Omega} \cdot \mathbf{n}_\Gamma(\mathbf{r}) < 0$. Let us suppose that neutrons are reflected into the reactor core, as follows:

$$\varphi(\mathbf{r}, \boldsymbol{\Omega}, E, t) = \varphi(\mathbf{r}, \boldsymbol{\Omega}', E, t), \quad (3.20)$$

with

$$\boldsymbol{\Omega}' = \boldsymbol{\Omega} - 2\mathbf{n}_\Gamma(\mathbf{r})(\mathbf{n}_\Gamma(\mathbf{r}) \cdot \boldsymbol{\Omega}) \quad (3.21)$$

Eq. (3.20) is known as reflective boundary condition, and may be derived from Eq. (3.16), by setting

$$\beta(\mathbf{r}', \boldsymbol{\Omega}', E' \rightarrow \mathbf{r}, \boldsymbol{\Omega}, E) = \delta(\mathbf{r}' - \mathbf{r}) \delta(\boldsymbol{\Omega}' - (\boldsymbol{\Omega} - 2\mathbf{n}_\Gamma(\mathbf{n}_\Gamma \cdot \boldsymbol{\Omega}))) \delta(E' - E). \quad (3.22)$$

Eq. (3.20) assumes that the boundary is a perfect mirror, as neutrons may be only reflected with a given angle and no escapes are admitted. This is also equivalent to an infinite periodic medium, where the elementary unit is the multiplying system whose boundary has been provided with reflective boundary conditions. Both vacuum and reflective boundary conditions are commonly used in lattice calculations, as shown in Chapter 5.

3.2.6 . Monte Carlo vs Deterministic Methods

The solution of the Boltzmann equation can be attained by means of diverse numerical approaches, based either on Monte Carlo or deterministic methods:

- **Monte Carlo methods**, [20]. As the solution of the transport equation corresponds to the expected value $E[x]$ of the neutron distribution $P(x)$, a Monte Carlo game may be formulated by generating a sample $\{x_i\}_{i=1, \dots, N}$ obeying $P(x)$. An approximation of $E[x]$ can be then retrieved by computing the arithmetic mean, $\mu_N = \frac{1}{N} \sum_{i=1}^N x_i$, such that

$\lim_{N \rightarrow \infty} \mu_N = E[x]$. The primary source of error is due to the finiteness of the sample: one can prove that the statistical error goes as $1/\sqrt{N}$.

In practice, Monte Carlo methods simulate the particle histories, one by one. In this sense, Monte Carlo's point of view is *microscopic*, since it closely follows the trajectory of the particle as it migrates and undergoes collisions in matter. The ultimate objective is to compute the response of a 'detector' region D , by computing an ensemble average. In principle, no approximations are introduced, the only limiting factor being the computational time to achieve convergence. It can be proven that Monte Carlo become competitive with deterministic methods as the number of dimensions of the problem increases.

- **Deterministic methods.** Unlike Monte Carlo, deterministic methods adopt a *macroscopic* point of view, which consists in discretizing each variable of the phase space and applying numerical techniques to the multigroup formulation of the neutron transport equation. As a result, the limit of deterministic methods resides in the inherent error introduced by the particular discretization. More specifically, the approximation which mostly affects the numerical precision of deterministic solvers is the multigroup discretization, weighting the macroscopic cross sections by the angular flux. As the latter is an unknown, an approximated self-shielding model is adopted, to compute the multigroup cross sections. Despite the loss of precision with respect to continuous-energy Monte Carlo, deterministic methods praise much shorter computational times and, as a consequence, are generally preferred over Monte Carlo for industrial calculations. In the following, an overview of the main deterministic approaches will be provided, before introducing the **I**ntegro-**D**ifferential **T**ransport solver, **IDT**.

4 - SOLUTION OF THE BOLTZMANN EQUATION BY DETERMINISTIC METHODS

4.1 . TIME-DEPENDENT NEUTRON TRANSPORT EQUATION

4.1.1 . Prompt and Delayed Neutrons

When a neutron is absorbed by a fissile nucleus, a fission reaction may occur. For a split second, a compound nucleus is formed, whose shape rapidly evolves, as described in Fig. 4.1.

Due to Colombian repulsion, the peanut lobes (stage 3, Fig. 4.1) are pushed apart, until fission in two excited fragments occurs (binary fission). The fission fragments rapidly de-excite, emitting ν_p prompt neutrons and γ -rays, with a delay of $10^{-17}s$. As mentioned in 3.2.2, prompt fission neutrons are emitted with energy distribution χ_p , where χ_p is the Watt spectrum, with average energy of about $2 MeV$. Nevertheless, a fraction of neutrons ν_d ($\sim 1\%$) are emitted with much longer delay, due to the decay of some fission products, named **precursors**, with a half-life of tens of seconds. These neutrons are called **delayed neutrons** and are emitted with energy spectrum χ_d , of average $\sim 100 keV$. Thanks to delayed neutrons, the control of nuclear power plants is possible, allowing for civil applications of nuclear energy.

As a consequence, in order to assess the time-dependent solution of the Boltzmann equation, one has to split the neutron source in two terms, i.e., a prompt fission source q_p , Eq. (4.1), and a delayed source q_d , such that, Eq. (4.2)

$$q_p(\mathbf{r}, E, t) = \sum_i \frac{\chi_p^i(\mathbf{r}, E)}{4\pi} \int_{4\pi} \int_0^\infty d^2\Omega' dE' \nu_p^i \Sigma_f^i(\mathbf{r}, E') \varphi(\mathbf{r}, \Omega', E', t), \quad (4.1)$$

$$q_d(\mathbf{r}, E, t) = \sum_{j=1}^M \frac{\chi_d^j(\mathbf{r}, E)}{4\pi} \lambda_j c_j(\mathbf{r}, t). \quad (4.2)$$

Precursors are generally subdivided into M families, as indicated in Eq. (4.2), with different decay constants λ_j and concentrations c_j , $j = 1, \dots, M$, evolving as follows:

$$\frac{\partial c_j(\mathbf{r}, t)}{\partial t} = -\lambda_j c_j(\mathbf{r}, t) + \int_0^\infty dE \nu_d^j \Sigma_f^j(\mathbf{r}, E) \Phi(\mathbf{r}, E, t), \quad j = 1, \dots, M. \quad (4.3)$$

For the sake of completeness, let us introduce the operatorial form of the transport equation, which reads as

$$\frac{1}{v} \frac{\partial \varphi}{\partial t} + \mathbf{L}\varphi = \mathbf{H}\varphi + \mathbf{F}_p\varphi + q_d + q_e, \quad (4.4)$$

where \mathbf{L} is the transport operator,

$$\mathbf{L}\varphi = \boldsymbol{\Omega} \cdot \nabla \varphi + \Sigma_t \varphi, \quad (4.5)$$

\mathbf{H} is the scattering operator,

$$\mathbf{H}\varphi = \int_{4\pi} \int_0^\infty d^2\Omega' dE' \nu_s \Sigma_s(\mathbf{r}, \Omega', E' \rightarrow \boldsymbol{\Omega}, E) \varphi(\mathbf{r}, \Omega', E', t), \quad (4.6)$$

\mathbf{F}_p is the prompt fission operator,

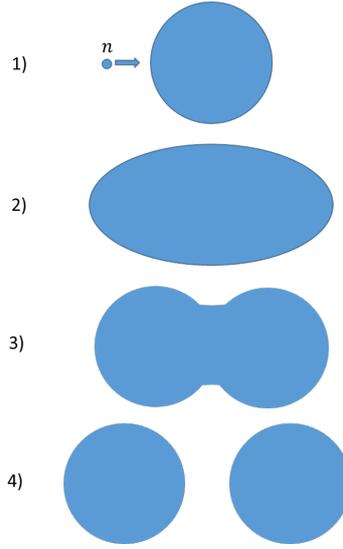


Figure 4.1: Fission reaction.

$$\mathbf{F}_p \varphi = \sum_i \frac{\chi_p^i(\mathbf{r}, E)}{4\pi} \int_{4\pi} \int_0^\infty d^2\Omega' dE' \nu_p^i \Sigma_f^i(\mathbf{r}, E') \varphi(\mathbf{r}, \Omega', E', t), \quad (4.7)$$

q_d is the delayed neutron source (Eq. (4.2)) and q_e is an external source.

4.1.2 . Flux Factorization and Point-Kinetics

Let us suppose that a steady-state reactor is perturbed at time $t = 0$. As a consequence, the reactor undergoes a quick adjustment, followed by a slower evolution of the whole power map. With this in mind, let us factorize the angular flux as

$$\varphi(\mathbf{r}, \Omega, E, t) = a(t) \psi(\mathbf{r}, \Omega, E, t), \quad (4.8)$$

where $a(t)$ is a time-dependent amplitude, such that

$$a(t) = \left\langle \omega, \frac{1}{v} \varphi \right\rangle, \quad (4.9)$$

whereas $\psi(\mathbf{r}, \Omega, E, t)$ is a shape function, slowly varying in time, satisfying

$$\left\langle \omega, \frac{1}{v} \psi \right\rangle = 1, \quad (4.10)$$

where $\langle \cdot, \cdot \rangle$ indicates a scalar product over the whole phase space D and $\omega(\mathbf{r}, \Omega, E)$ is an arbitrary non-null weighting function. Let us integrate Eq. (4.4) over D , after multiplication by ω . Analogously, Eq. (4.3) is multiplied by $\chi_d^j \cdot \omega$ and integrated over D . For the sake of simplicity, let us neglect the external neutron source. The final result is a set of $M + 1$ equations, of the kind

$$\frac{da}{dt} = \frac{\rho - \beta}{\Lambda} a + \sum_j \lambda_j c^j \quad (4.11)$$

$$\frac{d\kappa^j}{dt} = -\lambda_j \kappa^j + \frac{\beta^j}{\Lambda} a, \quad (4.12)$$

where

$$\rho = \frac{\langle \omega, (\mathbf{F} - (\mathbf{L} - \mathbf{H}))\psi \rangle}{\langle \omega, \mathbf{F}\psi \rangle} \quad (4.13)$$

is the reactivity,

$$\beta^j = \frac{\langle \omega, \mathbf{F}_d^j \psi \rangle}{\langle \omega, \mathbf{F}\psi \rangle} \quad (4.14)$$

is the effective delayed neutron fraction for family j ,

$$\Lambda = \frac{1}{\langle \omega, \mathbf{F}\psi \rangle} \quad (4.15)$$

is mean neutron generation time,

$$\kappa = \langle \omega, \chi_d^j c_j \rangle \quad (4.16)$$

is the weighed precursor concentration for family j . In Eq. (4.14), \mathbf{F}_d^j is the delayed fission operator for family j and reads as

$$\mathbf{F}_d^j \varphi = \frac{\chi_d^j(\mathbf{r}, E)}{4\pi} \int_{4\pi} \int_0^\infty d^2\Omega' dE' \nu_d^j \Sigma_f(\mathbf{r}, E') \varphi(\mathbf{r}, \Omega', E'.t), \quad (4.17)$$

Finally, we also introduced the steady-state fission operator $\mathbf{F} = \mathbf{F}_p + \sum_{j=1}^M \mathbf{F}_d^j$.

Although the weighting function $\omega(\mathbf{r}, \Omega, E)$ is, in principle, arbitrary, it may be proven that the adjoint to the critical flux, [21], is an optimal choice, [22].

4.1.3 . Improved Quasi-Static Method

In 4.1.2, the point-kinetics equations have been derived, in order to predict the time evolution of the amplitude function $a(t)$ and of the weighted precursor concentrations $\kappa^j(t)$, with $j = 1, \dots, M$. Let us replace Eq. (4.8) in Eq. (4.4) and (4.3). We obtain

$$\frac{1}{v} \frac{\partial \psi}{\partial t} + \frac{1}{v} \frac{\psi}{a} \frac{da}{dt} + \mathbf{L}\psi = \mathbf{H}\psi + \mathbf{F}_p\psi + \frac{1}{a}(q_d + q_e) \quad (4.18)$$

$$\frac{\partial c_j}{\partial t} = -\lambda_j c_j + \frac{4\pi a}{\chi_d^j} \mathbf{F}_d^j \psi. \quad (4.19)$$

As $\frac{\partial \psi}{\partial t} \ll \frac{da}{dt}$, in order to solve Eq. (4.11) - (4.12) and (4.18) - (4.19) two time-discretization scales are used, a fine mesh $T = \{t_i\}_{i=1, \dots, n}$, on which the amplitude function is evaluated and a coarse mesh $\tau \subseteq T$, on which the shape function is evaluated. Eq. (4.11) - (4.12) and (4.18) - (4.19) are solved alternatively, where Eq. (4.11) - (4.12) return a , given ψ , and Eq. (4.18) - (4.19) return ψ , given a and $\frac{da}{dt}$. Note that ψ is needed in Eq. (4.11) - (4.12), as the equation coefficients are given by Eq. (4.13), (4.14) and (4.15). A scheme is proposed in Fig. 4.2 to visualize the algorithm, for one coarse step, [23]

4.2 . STEADY-STATE NEUTRON TRANSPORT

In neutronics, a multiplying system is said to be **critical**, if the production and disappearance rates counterbalance each other. If the reactor is close to criticality, one may get rid of the time dependence in (4.4) artificially, by dividing the fission source by a positive parameter k , namely the **effective multiplication factor**,

$$\mathbf{L}\psi_k = \mathbf{H}\psi_k + \frac{1}{k}\mathbf{F}\psi_k, \quad (4.20)$$

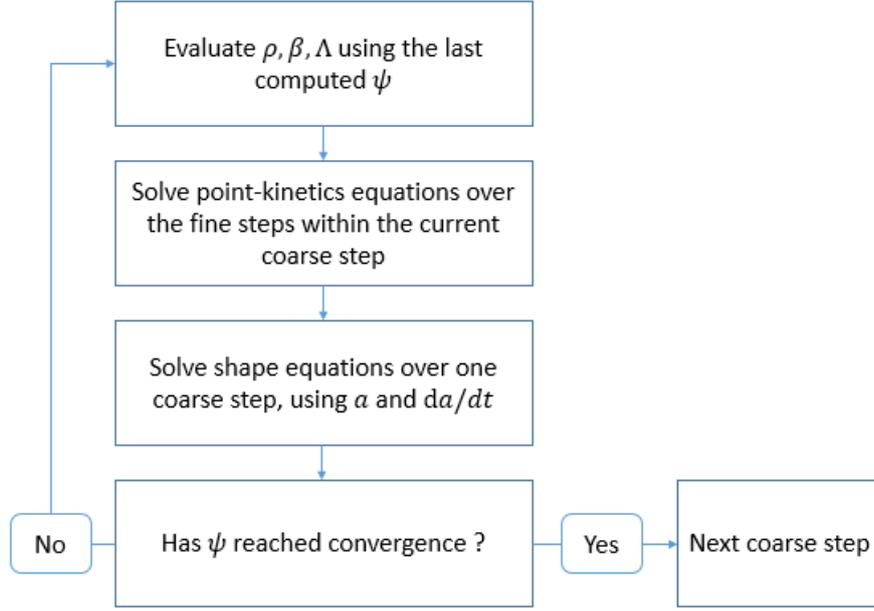


Figure 4.2: Improved Quasi-Static method.

where $\psi_k = \psi_k(\mathbf{r}, \Omega, E)$. Eq. 4.20 is an **eigenvalue problem**. We seek non-trivial solutions of the kind $\{k, \psi_k\}$, where ψ_k represents the eigenfunction corresponding to the eigenvalue k . As the only solution with a physical significance is the one corresponding to the **fundamental** (or **dominant**) **eigenvalue**, i.e., the maximum value of k satisfying Eq. (4.20), the present work will focus on searching the fundamental eigenpair.

If ψ_k is a solution of Eq. (4.20), then any function of the type $c \cdot \psi_k$, where c is a constant, is also solution of the k -eigenvalue problem. This constant may be set, so that the integral of the power distribution matches the power of the reactor, [5].

Finally, according to the value of k , the reactor may be critical ($k = 1$), sub-critical ($k < 1$) or super-critical ($k > 1$).

4.3 . ANGULAR DEPENDENCE OF THE SCATTERING CROSS SECTION

As mentioned in 3.2.2, for isotropic materials, the double-differential scattering cross section $\Sigma_s(\mathbf{r}, \Omega', E' \rightarrow \Omega, E)$ does not depend on Ω', Ω separately, but only on the cosine of the deviation angle, $\mu = \Omega' \cdot \Omega$. Thus,

$$\Sigma_s(\mathbf{r}, \Omega', E' \rightarrow \Omega, E) = \frac{1}{2\pi} \Sigma_s(\mathbf{r}, E' \rightarrow E, \mu), \quad (4.21)$$

where $\Sigma_s(\mathbf{r}, E, \mu)$ may be projected onto the Legendre polynomials, [24], as follows:

$$\Sigma_s(\mathbf{r}, E' \rightarrow E, \mu) = \frac{1}{2} \sum_{l=0}^{\infty} (2l+1) \Sigma_{s,l}(\mathbf{r}, E' \rightarrow E) P_l(\mu), \quad (4.22)$$

with

$$\Sigma_{s,l}(\mathbf{r}, E' \rightarrow E) = \int_{-1}^1 d\mu \Sigma_s(\mathbf{r}, E' \rightarrow E, \mu) P_l(\mu). \quad (4.23)$$

Thanks to the addition theorem,

$$P_l(\mu) = \sum_{m=-l}^{+l} A_{l,m}(\boldsymbol{\Omega}) A_{l,m}(\boldsymbol{\Omega}'), \quad (4.24)$$

the scattering source becomes

$$\mathbf{H}\varphi = \sum_{l=0}^{\infty} \frac{(2l+1)}{4\pi} \sum_{m=-l}^{+l} A_{l,m}(\boldsymbol{\Omega}) \int_0^{\infty} dE' \Sigma_{s,l}(\mathbf{r}, E' \rightarrow E) \Phi_{l,m}(\mathbf{r}, E'), \quad (4.25)$$

where

$$\Phi_{l,m}(\mathbf{r}, E) = \int_{4\pi} d^2\Omega A_{l,m}(\boldsymbol{\Omega}) \psi(\mathbf{r}, \boldsymbol{\Omega}, E) \quad (4.26)$$

are the flux angular moments and $A_{l,m}$ are the real spherical harmonics, [25]. If the expansion is truncated at $l = K$, the total number of spherical harmonics (and corresponding angular moments) will be $(K+2)(K+1)/2$ in 2D and $(K+1)^2$ in 3D.

4.4 . MULTIGROUP APPROXIMATION

4.4.1 . Multigroup Neutron Transport Equation

The time-independent neutron transport equation may be integrated over energy intervals of finite length. More specifically, the energy range from $E_G = 0$ to $E_1 = 20MeV$ is subdivided into G energy groups, namely $[E_g, E_{g-1}]$, where $g = G, \dots, 1$, inside which neutrons are assumed to be mono-energetic. The following group-dependent quantities may be defined:

$$\psi^g(\mathbf{r}, \boldsymbol{\Omega}) = \int_{E_g}^{E_{g-1}} \psi(\mathbf{r}, \boldsymbol{\Omega}, E) dE, \quad (4.27)$$

$$\Sigma^g(\mathbf{r}, \boldsymbol{\Omega}) = \frac{1}{\psi^g(\mathbf{r}, \boldsymbol{\Omega})} \int_{E_g}^{E_{g-1}} \Sigma(\mathbf{r}, E) \psi(\mathbf{r}, \boldsymbol{\Omega}, E) dE. \quad (4.28)$$

Equation 4.28 cannot be evaluated, as the angular flux is unknown. Although Σ^g is, in principle, a function of the angle, this dependence is usually neglected. The analysis of the ensemble of approximations allowing for the calculation of multigroup cross sections is beyond the scope of the present manuscript. Nevertheless, some introductory elements are provided in Sect. 12.3.1. For more details, the interested reader is referred to [5], [26], [27].

The multigroup k -eigenvalue formulation of the transport equation with P_N -scattering is given by

$$\begin{aligned} & \boldsymbol{\Omega} \cdot \nabla \psi^g(\mathbf{r}, \boldsymbol{\Omega}) + \Sigma^g(\mathbf{r}) \psi^g(\mathbf{r}, \boldsymbol{\Omega}) = \\ & \frac{1}{4\pi} \sum_{g'=1}^G \sum_{l=0}^K (2l+1) \Sigma_{s,l}^{g' \rightarrow g}(\mathbf{r}) \sum_{m=-l}^{+l} A_{l,m}(\boldsymbol{\Omega}) \Phi_{l,m}^{g'}(\mathbf{r}) + \\ & \frac{1}{4\pi k} \sum_i \chi_f^{i,g}(\mathbf{r}) \sum_{g'=1}^G \nu_f^{i,g'} \Sigma_f^{i,g'}(\mathbf{r}) \Psi^{g'}(\mathbf{r}), \end{aligned} \quad (4.29)$$

where

$$\Sigma_{s,l}^{g' \rightarrow g}(\mathbf{r}) = \frac{1}{\Phi_{l,m}^g(\mathbf{r})} \int_{E_g}^{E_{g-1}} \Sigma_{s,l}(\mathbf{r}, E' \rightarrow E) \Phi_{l,m}(\mathbf{r}, E') dE', \quad (4.30)$$

$$\chi_f^{i,g}(\mathbf{r}) = \int_{E_g}^{E_{g-1}} \chi_f^i(\mathbf{r}, E) dE, \quad (4.31)$$

$$\nu_f^{i,g} \Sigma_f^{i,g}(\mathbf{r}) = \frac{1}{\Psi^g(\mathbf{r})} \int_{E_g}^{E_{g-1}} \nu_f^i \Sigma_f^i(\mathbf{r}, E) \Psi(\mathbf{r}, E) dE. \quad (4.32)$$

4.4.2 . Iterations

For the sake of conciseness, let us introduce the operatorial form of Eq. (4.29),

$$\mathbf{L}^g \psi^g = \sum_{g'} \mathbf{H}^{g' \rightarrow g} \psi^{g'} + \frac{1}{k} q_f^g, \quad g = 1, \dots, G, \quad (4.33)$$

where

$$q_f^g = [\mathbf{F}\psi]^g \quad (4.34)$$

is the multigroup fission source. Let us rewrite the scattering source as the sum of three contributions, namely the down-scattering, self-scattering and up-scattering, respectively,

$$\sum_{g'} \mathbf{H}^{g' \rightarrow g} \psi^{g'} = \sum_{g' < g} \mathbf{H}^{g' \rightarrow g} \psi^{g'} + \mathbf{H}^{g \rightarrow g} \psi^g + \sum_{g' > g} \mathbf{H}^{g' \rightarrow g} \psi^{g'}. \quad (4.35)$$

Eq. (4.33) can be formulated in matrix form:

$$\begin{pmatrix} L^1 - H^{11} & -H^{12} & \dots & -H^{1g} & \dots & -H^{1G} \\ -H^{21} & L^2 - H^{22} & \dots & -H^{2g} & \dots & -H^{2G} \\ \dots & \dots & \dots & \dots & \dots & \dots \\ -H^{g1} & -H^{g2} & \dots & -L^g - H^{gg} & \dots & -H^{gG} \\ \dots & \dots & \dots & \dots & \dots & \dots \\ -H^{G1} & -H^{G2} & \dots & -H^{Gg} & \dots & L^G - H^{GG} \end{pmatrix} \begin{pmatrix} \psi^1 \\ \psi^2 \\ \dots \\ \psi^g \\ \dots \\ \psi^G \end{pmatrix} = \frac{1}{k} \begin{pmatrix} q_f^1 \\ q_f^2 \\ \dots \\ q_f^g \\ \dots \\ q_f^G \end{pmatrix} \quad (4.36)$$

where $H^{gg'} = H^{g' \rightarrow g}$.

Matrix $\mathbf{B}^g = (\mathbf{L}^g - \sum_{g'} \mathbf{H}^{g' \rightarrow g}) = (\mathbf{L}^g - \sum_{g'} \mathbf{H}^{gg'})$, where $\mathbf{B} = \mathbf{L} - \mathbf{H}$ will be referred to as **Boltzmann operator**, has a simple interpretation:

- The lower triangular part contains the multigroup down-scattering operator;
- The diagonal contains the within-group transport operator;
- The upper triangular part contains the multigroup up-scattering operator. Several terms are equal to zero, as the probability to be promoted to a higher energy group is non-negligible only for neutrons in the thermal range, i.e., neutrons from group $g > g_{th}$, where $g_{th} \sim 1eV$.

Hence, equation $\mathbf{B}^g \psi^g = \frac{1}{k} q_f^g, g = 1, \dots, G$, may resemble a lower triangular system.

In order to solve Eq. (4.36), a loop of three nested iterations is generally applied. Starting from a trial fission integral I_0^g and eigenvalue k_0 , where $g = 1, \dots, G$, Eq. (4.36) is solved by Gauss-Siedel iterations, starting from group $g = 1$, for which

$$(L^1 - H^{11})\psi^1 = \frac{1}{k_0} q_0^1, \quad (4.37)$$

as neutrons from group $g = 1$ do not undergo up-scattering and may only be refilled by fission. In the following, the ensemble of nested iterations is discussed in some detail.

Outer Iterations

Let us consider the k -eigenvalue problem,

$$\mathbf{B}\psi(\mathbf{r}, \Omega, E) = \frac{1}{k} \mathbf{F}\psi(\mathbf{r}, \Omega, E). \quad (4.38)$$

The fission source may be factorized as

$$\mathbf{F}\psi = \sum_i \chi^i(\mathbf{r}, E) I^i(\mathbf{r}), \quad (4.39)$$

where I^i is the **fission integral** and reads as

$$I^i(\mathbf{r}) = \int dE \nu_f^i \Sigma_f^i(\mathbf{r}, E) \Psi(\mathbf{r}, E). \quad (4.40)$$

At iteration $e = 0$, the eigenvalue and fission integral are initialized, $k^{(0)} = k_0$, $I^{(0)}(\mathbf{r}) = I_0(\mathbf{r})$. At iteration e , the Boltzmann operator is implicitly inverted, as

$$\mathbf{B}\psi^{(e+1)} = \frac{1}{k^{(e)}} \mathbf{F}\psi^{(e)}. \quad (4.41)$$

In order to invert the Boltzmann operator, two nested cycles are needed, namely the thermal and inner iterations. Once computed, $\psi^{(e+1)}$ allows for updating the eigenvalue and fission integral. In particular,

$$k^{(e+1)} = k^{(e)} \frac{\langle \omega, \mathbf{F}\psi^{(e+1)} \rangle}{\langle \omega, \mathbf{F}\psi^{(e)} \rangle}, \quad (4.42)$$

ω being an arbitrary weight function.

Thermal iterations

As thermal neutrons may be up-scattered up to $g = g_{th} - 1$, where g_{th} is the so-called **thermal cut-off** and amounts to ~ 1 eV, in order to assess the up-scattering source feeding groups $g > g_{th}$, thermal iterations, based on the Gauss-Siedel method, are implemented. If t is the thermal iteration index, the slowing-down equations read as

$$(\mathbf{L}^g - \mathbf{H}^{gg})\psi^{g,(t+1)} = \sum_{g' < g} \mathbf{H}^{gg'} \psi^{g',(t+1)} + \sum_{g' > g} \mathbf{H}^{gg'} \psi^{g',(t)} + \frac{1}{k^{(e)}} \mathbf{F}\psi^{g,(e)}, \quad (4.43)$$

where $\sum_{g' < g} \mathbf{H}^{gg'} \psi^{g',(t+1)}$, $\sum_{g' > g} \mathbf{H}^{gg'} \psi^{g',(t)}$ and $\frac{1}{k^{(e)}} \mathbf{F}\psi^{g,(e)}$ are the down-scattering, up-scattering and fission source, respectively. The self-scattering source $\mathbf{H}^{gg} \psi^{g,(t+1)}$ is kept on the left-hand side of Eq. (4.43) and solved by inner iterations.

At iteration $t = 0$, the angular flux moments $\Phi_{l,m}^{g,(t=0)}(\mathbf{r})$ are provided by the current outer iteration, $\Phi_{l,m}^{g,(e)}(\mathbf{r}) = \int_{4\pi} d^2\Omega A_{l,m}(\Omega)\psi^{g,(e)}(\mathbf{r}, \Omega)$.

As fast neutrons do not undergo up-scattering collisions, the corresponding term from Eq. (4.43) vanishes, thus allowing for skipping the thermal loop and computing ψ^g by inner iterations,

$$(\mathbf{L}^g - \mathbf{H}^{gg})\psi^g = \sum_{g' < g} \mathbf{H}^{gg'}\psi^{g'} + \frac{1}{k^{(e)}} \mathbf{F}\psi^{g,(e)}, g = 1, \dots, g_{th}. \quad (4.44)$$

Taking advantage of Eq. (4.44), the down-scattering source from groups $g' < g_{th}$ (fast groups) to $g_{th} < g < G$ (thermal groups) can be evaluated as $\sum_{g' < g_{th}} \mathbf{H}^{gg'}\psi^{g'}$ and injected into the thermal loop, which solves

$$\begin{aligned} (\mathbf{L}^g - \mathbf{H}^{gg})\psi^{g,(t+1)} &= \sum_{g_{th} < g' < g} \mathbf{H}^{gg'}\psi^{g',(t+1)} + \sum_{g < g'} \mathbf{H}^{gg'}\psi^{g',(t)} \\ &+ \left(\sum_{g' < g_{th}} \mathbf{H}^{gg'}\psi^{g',(e)} + \frac{1}{k^{(e)}} \mathbf{F}\psi^{g,(e)} \right), \quad g = g_{th}, \dots, G, \end{aligned} \quad (4.45)$$

where $\sum_{g_{th} < g' < g} \mathbf{H}^{gg'}\psi^{g',(t+1)}$ and $\sum_{g < g'} \mathbf{H}^{gg'}\psi^{g',(t)}$ represent the down-scattering from thermal group g' to thermal group $g > g'$ and up-scattering from thermal group g' to thermal group $g < g'$, respectively.

Finally, convergence is verified, by comparing the error with a given tolerance.

Inner iterations

Inner iterations solve the boundary-value problem

$$\mathbf{L}^g\psi^{g,(i+1)}(x) = \mathbf{H}^{gg}\psi^{g,(i)}(x) + q^g(x), \quad x \in D \quad (4.46)$$

$$\psi^{g,(i+1)}(x) = \beta\psi^{g,(i)}(x), \quad x \in \partial D^-, \quad (4.47)$$

where i is the inner loop index, x is a phase-space variable, β is the discretized albedo operator and q^g coincides with the right-hand side of Eq. (4.44) and (4.45), for fast and thermal neutrons, respectively.

At iteration $i = 0$, the angular flux moments $\Phi_{l,m}^{g,(i=0)}(\mathbf{r})$ are provided by

- the current outer iteration, i.e., $\Phi_{l,m}^{g,(e)}(\mathbf{r})$, for fast neutrons;
- the current thermal iteration, i.e., $\Phi_{l,m}^{g,(t)}(\mathbf{r})$, for thermal neutrons.

The total angular source is computed as

$$Q^{g,(i)} = \mathbf{H}^{gg}\psi^{g,(i)} + q^g \quad (4.48)$$

and a fixed-source problem of the kind

$$\mathbf{L}^g\psi^{g,(i+1)}(x) = Q^{g,(i)}(x), \quad x \in D \quad (4.49)$$

$$\psi^{g,(i+1)}(x) = \beta\psi^{g,(i)}(x), \quad x \in \partial D^- \quad (4.50)$$

is solved iteratively until convergence.

4.5 . ANGULAR DISCRETIZATION

Two approaches are commonly used to discretize the angular variable, namely the S_N (**or discrete ordinates) method** and the P_N (**or spherical harmonics) method**. The former consists in selecting a set of discrete directions and computing the angular integrals by means of a quadrature formula; the latter is based on projecting the angular flux on spherical harmonics. Sect. 4.5.1 and 4.5.2 provide some detail on the subject.

4.5.1 . Discrete Ordinates Method

The discrete ordinates method, also known as S_N method, calculates angular integrals by means of a numerical quadrature. In practice, given a function $g(\boldsymbol{\Omega})$,

$$\frac{1}{4\pi} \int_{4\pi} g(\boldsymbol{\Omega}) d^2\Omega \approx \sum_{d=1}^{N_d} g(\boldsymbol{\Omega}_d) \omega_d, \quad (4.51)$$

where $\{\boldsymbol{\Omega}_d, \omega_d\}$, with $d = 1, \dots, N$, represents a set of discrete directions and associated weights, N being the number of selected directions. Note that the weights are normalized to 1, i.e.,

$$\sum_{d=1}^N \omega_d = 1. \quad (4.52)$$

Each direction $\boldsymbol{\Omega}_d$ may be pictured as a point on the unit sphere Γ , with associated surface $\omega_d = \Delta\Gamma_d$. The flux moments $\Phi_{l,m}^g$ are reconstructed along inner iterations by angular quadrature, by cumulating the contributions from the selected directions,

$$\Phi_{l,m}^{g,(i+1)}(\mathbf{r}) = \int_{4\pi} d^2\Omega A_{l,m}(\boldsymbol{\Omega}) \psi^{g,(i+1)}(\mathbf{r}, \boldsymbol{\Omega}) \approx \sum_{d=1}^{N_d} \omega_d A_{l,m}(\boldsymbol{\Omega})_d \psi^{g,(i+1)}(\mathbf{r}, \boldsymbol{\Omega}_d). \quad (4.53)$$

Level Symmetric, Gauss-Legendre and Gauss-Chebyshev formulas are some of the most popular choices, [5]. Further details are beyond the scope of the present work.

4.5.2 . PN Method

The P_N method consists in projecting the angular flux onto spherical harmonics. If the series development is truncated at order N , we obtain

$$\psi^g(\mathbf{r}, \boldsymbol{\Omega}) \approx \frac{1}{4\pi} \sum_{l=0}^N (2l+1) \sum_{m=-l}^{+l} A_{l,m}(\boldsymbol{\Omega}) \Phi_{l,m}^g(\mathbf{r}). \quad (4.54)$$

As N goes to infinity, the series expansion converges to the solution, i.e., the angular flux.

Expansion (4.54) is injected into the transport equation. Taking advantage of the properties of the spherical harmonics, the initial problem can be rearranged, obtaining a system of $(N+1)^2$ coupled equations, where the angular moments $\Phi_{l,m}^g(\mathbf{r})$ are the unknowns.

Differently from S_N , the P_N method does not suffer from ray-effect, [28]. On the downside, it produces a large system of coupled equations, with the number of degrees of freedom growing as $(N+1)^2$. Also, the P_N approximation is less efficient when representing discontinuous solutions, which is often the case for reactor calculations, [29].

In order to reduce the memory requirements, the P_N method may be replaced by a simplified approach, namely the SP_N **method**, which only retains $2(N + 1)$ harmonics, as sketched in Tab. 4.1, for $N = 3$. In practice, if l is even, only moment $\Phi_{l,0}$ is preserved, the others are disregarded; on the contrary, if l is odd, three moments are preserved, namely $\Phi_{l,-1}$, $\Phi_{l,0}$ and $\Phi_{l,1}$. Unlike S_N and P_N , SP_N does not converge to the solution, as N goes to infinity. Despite this inconvenience, it provides significant improvements over diffusion theory, thus finding application in core calculations, [30].

	$m = -3$	$m = -2$	$m = -1$	$m = 0$	$m = 1$	$m = 2$	$m = 3$
$l = 0$	-	-	-	$\Phi_{0,0}$	-	-	-
$l = 1$	-	-	$\Phi_{1,-1}$	$\Phi_{1,0}$	$\Phi_{1,1}$	-	-
$l = 2$	-	$\Phi_{2,-2}$	$\Phi_{2,-1}$	$\Phi_{2,0}$	$\Phi_{2,1}$	$\Phi_{2,2}$	-
$l = 3$	$\Phi_{3,-3}$	$\Phi_{3,-2}$	$\Phi_{3,-1}$	$\Phi_{3,0}$	$\Phi_{3,1}$	$\Phi_{3,2}$	$\Phi_{3,3}$

Table 4.1: SP_N vs P_N method: moments in grey are neglected in SP_N .

4.6 . SPATIAL DISCRETIZATION

Several spatial discretization techniques are available in literature. For the sake of conciseness, the following analysis will focus on methods which are more closely related to the spatial discretization approach that is adopted in IDT, i.e., the method of short characteristics, which consists in projecting the Boltzmann equation on modular (Cartesian or tetrahedral) patterns and inverts the transport operator by means of angular probability matrices. Sect. 4.6.2 and 4.6.3 offer some detail on the method of collision probabilities and the MOC strategy, whereas a discussion on the method of short characteristics is available in Sect. 4.6.4. As all of them are based on the integral form on the neutron transport equation, Sect. 4.6.1 proposes a brief derivation of it².

4.6.1 . Integral Form of the Multigroup Transport Equation

Let us consider two points \mathbf{r} and $\mathbf{r} + x\boldsymbol{\Omega}$ on trajectory $t \parallel \boldsymbol{\Omega}$, x being the coordinate along the trajectory and $\boldsymbol{\Omega}$ an arbitrary angular direction. The angular flux at $\mathbf{r} + x\boldsymbol{\Omega}$ along direction $\boldsymbol{\Omega}$ reads as

$$\psi^g(\mathbf{r} + x\boldsymbol{\Omega}, \boldsymbol{\Omega}) = \psi^g(\mathbf{r}, \boldsymbol{\Omega}) e^{-\int_0^x dy \Sigma(\mathbf{r}+y\boldsymbol{\Omega})} + \int_0^x q^g(\mathbf{r} + y\boldsymbol{\Omega}, \boldsymbol{\Omega}) e^{-\int_y^x dz \Sigma(\mathbf{r}+z\boldsymbol{\Omega})} dy, \quad (4.55)$$

The first contribution accounts for neutrons departing from \mathbf{r} that arrive at $\mathbf{r} + x\boldsymbol{\Omega}$ without undergoing any collision, and thus changing neither direction of flight nor energy. The second term is related to the emission rate of neutrons with energy in g and angular direction $\boldsymbol{\Omega}$ along

²Methods based on the integral form of the Boltzmann equation introduce an approximation on the spatial and angular distribution of the neutron source. Integral methods generally assume flat and isotropic sources in the laboratory frame. On the contrary, methods based on the integro-differential form of the transport equation introduce an approximation on the angular flux, which is a much less regular function, particularly in the event of collisions (which is the case of nuclear reactors).

trajectory t between \mathbf{r} and $\mathbf{r} + x\boldsymbol{\Omega}$. Also Eq. 4.55 allows defining the optical path length as

$$\tau(x, y) = \int_y^x dz \Sigma(\mathbf{r} + z\boldsymbol{\Omega}), \quad (4.56)$$

Σ being the total cross section.

4.6.2 . Collision Probabilities

The integral form of the transport equation for an infinite system reads as

$$\psi(\mathbf{r}, \boldsymbol{\Omega}) = \int_0^\infty q(\mathbf{r} - s\boldsymbol{\Omega}, \boldsymbol{\Omega}) e^{-\tau(s)} ds, \quad (4.57)$$

with optical path

$$\tau(s) = \int_0^s \tau(\mathbf{r} - s'\boldsymbol{\Omega}) ds', \quad (4.58)$$

where energy group subscripts are omitted, for the sake of conciseness. Eq. 4.57 is integrated over the whole solid angle,

$$\Psi(\mathbf{r}) = \frac{1}{4\pi} \int_{4\pi} \int_0^\infty q(\mathbf{r} - s\boldsymbol{\Omega}, \boldsymbol{\Omega}) e^{-\tau(s)} ds. \quad (4.59)$$

Let us suppose that the neutron source is isotropic. Eq. 4.59 simplifies to

$$\Psi(\mathbf{r}) = \frac{1}{4\pi} \int_{4\pi} \int_0^\infty q(\mathbf{r} - s\boldsymbol{\Omega}) e^{-\tau(s)} ds. \quad (4.60)$$

Let us assume that the spatial domain is subdivided into Cartesian nodes (or *cells*) c , each finely discretized into multiple *regions*, of volume $V_{i,c}$. The neutron source is supposed to be constant over each $V_{i,c}$. A variable change is introduced, namely $\mathbf{r}' = \mathbf{r} - s\boldsymbol{\Omega}$. For the sake of simplicity, let us omit the cell index. Eq. 4.60 is multiplied by the total cross section Σ and integrated over each region V_i , thus obtaining,

$$\langle \Sigma, \Psi \rangle_{V_i} = \frac{1}{4\pi} \left\langle \Sigma, \sum_j q_j \int_{V_j} \frac{1}{s^2} e^{-\tau} d^3 r' \right\rangle_{V_i}. \quad (4.61)$$

Eq. 4.61 may be rearranged, by defining the **average neutron flux** over region V_i ,

$$\Psi_i = \frac{\int_{V_i} \Psi(\mathbf{r}) d^3 r}{V_i}, \quad (4.62)$$

and the **collision probabilities**,

$$P_{i,j} = \frac{1}{4\pi V_j} \int_{\mathbb{R}^3} d^3 r' \int_{V_i} \Sigma(\mathbf{r}) \frac{e^{-\tau(s)}}{s^2} d^3 r. \quad (4.63)$$

Eq. 4.63 can be interpreted as the probability for a neutron born uniformly and isotropically in region V_j to undergo its first collision in region V_i , as explained in Sect. 3.8 of Ref. [5]. Eq. 4.61 may be rewritten as

$$V_i \Sigma_i \Psi_i = \sum_j V_j P_{i,j} q_j, \quad (4.64)$$

where Σ_i is the flux-averaged cross section over region i , thus preserving the reaction rate,

$$\Sigma_i \Psi_i V_i = \langle \Sigma, \Psi \rangle_{V_i} = \int_{V_i} \Sigma(\mathbf{r}) \Psi(\mathbf{r}) d^3 r. \quad (4.65)$$

Assuming that cross sections are uniform over each region V_j , one may divide Eq. 4.63 by Σ_j , obtaining

$$p_{i,j} = \frac{1}{4\pi V_j} \int_{\mathbb{R}^3} d^3r' \int_{V_i} \frac{e^{-\tau(s)}}{s^2} d^3r. \quad (4.66)$$

Eq. 4.66 define the **reduced collision probabilities**. Note that Eq. 4.66 remains finite as Σ_i goes to zero. In other terms, the method of collision probabilities is able to model vacuum regions. By injecting Eq. 4.66 into Eq. 4.64, one gets

$$V_i \Psi_i = \sum_j V_j p_{i,j} q_j. \quad (4.67)$$

Collision probabilities satisfy reciprocity,

$$p_{i,j} V_j = p_{j,i} V_i, \quad (4.68)$$

and conservation relations,

$$\sum_i p_{i,j} \Sigma_i = \sum_i P_{i,j} = 1, \forall j. \quad (4.69)$$

Eq. 4.66 is computed numerically illuminating the geometry by parallel trajectories. This algorithm goes under the name of **ray-tracing** and will be described accurately in Sect. 6.3.4.

The method of collision probabilities finds application in legacy self-shielding, [32], and neutron flux calculations. The major limitations are the assumption of isotropic scattering and the size of the collision probability matrix, which scales quadratically with the total number of regions. One way to mitigate this limit consists in coupling the cells by neutron currents, thus introducing **current-coupled collision probabilities (CCCP)**, also known as Multicell method in the APOLLO documentation, [33], [34]. Cell boundary $\partial D = \partial D^- \cup \partial D^+$ is partitioned into sub-surfaces $s' \in \partial D^-$ and $s \in \partial D^+$, ∂D^- and ∂D^+ being the incoming and outgoing cell boundaries, respectively. Within each cell, for a given energy group, the following relation holds,

$$\Psi_i = \sum_j C_{i,j} q_j + \sum_{s'} I_{i,s'} j_{s'}^-, \quad (4.70)$$

where

- **C** is the collision matrix. $C_{i,j}$ represents the probability for a neutron born uniformly and isotropically in region j to undergo its first collision in region i ;
- **I** is the incoming matrix. $I_{i,s'}$ represents the probability for a neutron entering cell c through surface $s' \in \partial D^-$ to contribute to region i ;
- $j_{s'}$ is the average neutron current entering cell c through surface s' .

Cells are coupled by means of

$$j_s^+ = \sum_j E_{s,j} q_j + \sum_{s'} T_{s,s'} j_{s'}^-, \quad (4.71)$$

where

- \mathbf{E} is the escape matrix. $E_{s,j}$ represents the probability for a neutron born uniformly and isotropically in region j to exit cell c through surface $s \in \partial D^+$;
- \mathbf{T} is the transmission matrix. $T_{s,s'}$ can be interpreted as the probability for a neutron entering cell c through surface $s' \in \partial D^-$ to exit cell c through surface $s \in \partial D^+$;
- j_s^+ is the average neutron current exiting cell c through surface s . Note that j_s^+ matches the average incoming neutron current entering the neighbouring cell.

The calculation of the response matrices is not discussed in the present work, the interested readers are referred to [33].

Similarly to standard collision probabilities (CP), CCCP is able to handle unstructured geometries with high flexibility and numerical accuracy, [33]. Like CP, neutron scattering is supposed to be isotropic and anisotropy effects may be modelled, to some extent, by transport-corrected cross sections. The memory pressure is alleviated with respect to CP, but may still be a limiting factor, particularly for applications to large geometries comprising regions with different cross sections.

4.6.3 . Method of Characteristics

The method of characteristics (**MOC**) dates back to the seventies and is considered one of the most accurate tools for high-fidelity deterministic solutions. It consists in solving the transport equation along a set of parallel trajectories (or *characteristic curves*), along which the transport equation reduces to ordinary differential equations, for each direction of the quadrature formula. In MOC, the characteristic curves are continuous side-to-side straight lines, which are followed over the whole domain geometry³.

In order to introduce MOC, let us consider the integral form of the one-speed transport equation along a neutron trajectory $t \parallel \Omega_d$, where Ω_d belongs to the set of selected directions of the angular quadrature formula,

$$\psi(\mathbf{r}_t^- + x\Omega_d, \Omega_d) = \psi(\mathbf{r}_t^-, \Omega_d) e^{-\tau(x,0)} + \int_0^x q(\mathbf{r}_t^- + y\Omega_d, \Omega_d) e^{-\tau(x,y)} dy, \quad (4.72)$$

in which

- x is the coordinate along trajectory t ;
- \mathbf{r}_t^- is the incoming point of trajectory t into region $\alpha \subseteq \mathcal{G}$, \mathcal{G} being the assigned geometry, as shown in Fig. 4.3;
- $\tau(z_1, z_2) = \int_{z_1}^{z_2} \Sigma(\mathbf{r}_t^- + z\Omega_d) dz$ is the optical path between between $x = z_1$ and $x = z_2$.

In the following, the source and the macroscopic cross sections are assumed to be spatially constant within each region $\alpha \subseteq \mathcal{G}$,

$$q(\mathbf{r}, \Omega) = q_\alpha(\Omega), \forall r \in \alpha \quad (4.73)$$

$$\Sigma(\mathbf{r}) = \Sigma_\alpha, \forall r \in \alpha. \quad (4.74)$$

³Neutral particles are not sensitive to external long-range forces.

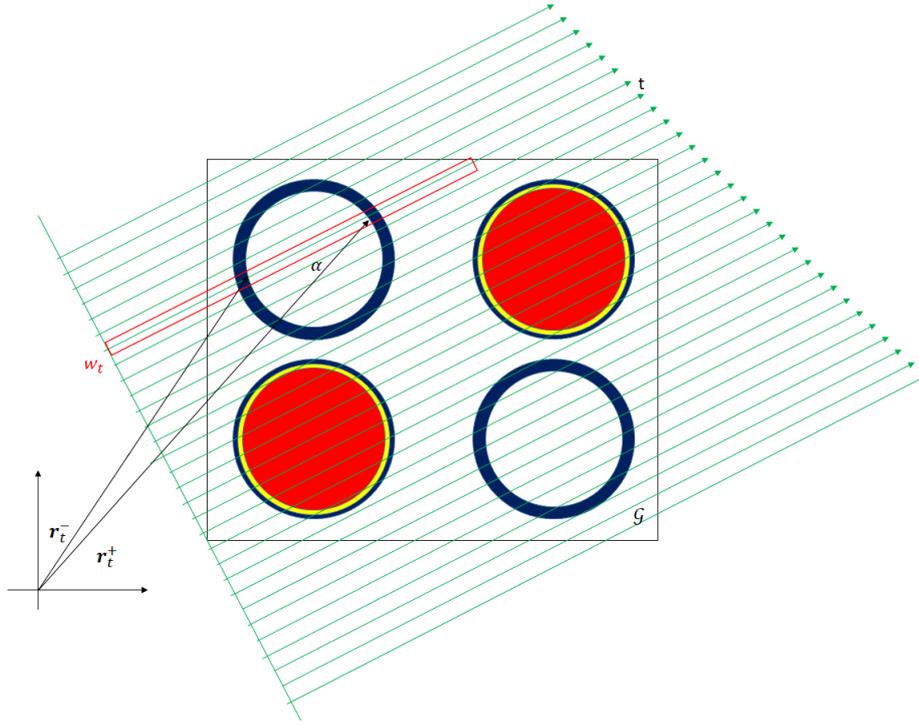


Figure 4.3: MOC tracking.

This approximation is often referred to as *Step Characteristics*, [35]. Eq. (4.72) between intersection points \mathbf{r}_t^+ and \mathbf{r}_t^- of trajectory t with region boundary $\partial\alpha$ reduces to

$$\psi(\mathbf{r}_t^+, \boldsymbol{\Omega}_d) = \psi(\mathbf{r}_t^-, \boldsymbol{\Omega}_d) e^{-\tau_\alpha(l,0)} + q_\alpha(\boldsymbol{\Omega}_d) \frac{1 - e^{-\tau_\alpha(l,0)}}{\Sigma_\alpha}, \quad (4.75)$$

which, with a mild change of notation, may be simplified to

$$\psi_t^+ = \psi_t^- e^{-\Sigma_\alpha l} + q_{d,\alpha} \frac{1 - e^{-\Sigma_\alpha l}}{\Sigma_\alpha}, \quad (4.76)$$

$l = \|\mathbf{r}_t^+ - \mathbf{r}_t^-\|$ being the chord-length, Σ_α the total macroscopic cross section of region α and ψ_t^+/ψ_t^- the outgoing/incoming angular flux, respectively. Eq. (4.76) goes under the name of **transmission equation**.

In order to compute the flux in region α , a **balance equation** may be derived, by volume-averaging the integro-differential form of the transport equation over region α ,

$$\frac{1}{V_\alpha} \left\langle 1, (\boldsymbol{\Omega}_d \cdot \nabla + \Sigma) \psi \right\rangle_{V_\alpha} = \frac{1}{V_\alpha} \left\langle 1, q \right\rangle_{V_\alpha} \quad (4.77)$$

$$\frac{1}{V_\alpha} \int_{V_\alpha} (\boldsymbol{\Omega}_d \cdot \nabla + \Sigma) \psi d^3r = \frac{1}{V_\alpha} \int_{V_\alpha} q d^3r. \quad (4.78)$$

Eq. (4.73) and (4.74) are then introduced into Eq. (4.78). Finally, by resorting to

$$\boldsymbol{\Omega} \cdot \nabla \psi = \nabla \cdot \boldsymbol{\Omega} \psi, \quad (4.79)$$

and the divergence theorem, one obtains the final form of the balance equation

$$\Sigma_{\alpha} \psi_{d,\alpha} = q_{d,\alpha} - \Delta j_{d,\alpha}, \quad (4.80)$$

where

$$\psi_{d,\alpha} = \frac{1}{V_{\alpha}} \int_{V_{\alpha}} d^3r \psi(\mathbf{r}, \boldsymbol{\Omega}_d) \quad (4.81)$$

is the volume-average of the flux over region α and

$$\begin{aligned} \Delta j_{d,\alpha} = & \frac{1}{V_{\alpha}} \int_{V_{\alpha}} d^3r \nabla \cdot \boldsymbol{\Omega}_d \psi(\mathbf{r}, \boldsymbol{\Omega}_d) = \\ & \frac{1}{V_{\alpha}} \oint_{\partial V_{\alpha}^+} d^3r j(\mathbf{r}, \boldsymbol{\Omega}_d) \cdot \mathbf{n}_s + \frac{1}{V_{\alpha}} \oint_{\partial V_{\alpha}^-} d^3r j(\mathbf{r}, \boldsymbol{\Omega}_d) \cdot \mathbf{n}_s \end{aligned} \quad (4.82)$$

is the net current over the boundary domain ∂V_{α} , along direction $\boldsymbol{\Omega}_d$. Eq. (4.82) is evaluated by numerical quadrature,

$$\Delta j_{d,\alpha} \approx \frac{1}{V_{\alpha}} \sum_{t \parallel \boldsymbol{\Omega}_d} w_t^{\perp} (\psi_t^+ - \psi_t^-), \quad (4.83)$$

in which

$$V_{\alpha} \approx \sum_{t \parallel \boldsymbol{\Omega}_d} w_t^{\perp} l_t \quad (4.84)$$

w_t^{\perp} and l_t being respectively the numerical weight and chord-length within domain α of trajectory t .

MOC is often considered a numerical reference for other spatial approximations. The error introduced by the numerical quadrature, i.e., Eq. (4.83) and (4.84), converges to zero, as the trajectory spacing w_t^{\perp} becomes smaller and smaller. It allows treating unstructured geometries [36] in an exact way and is also advantageous in terms of memory requirements, as, differently from CP and CCCP, it does not involve the calculation and storage of large response matrices. The use of cyclic trajectories [37] allows for an exact treatment of the angular flux at the domain boundary, for specular reflection, rotation, translation and other boundary conditions.

On the downside, the computational cost of MOC is strongly dependent on the system dimensions and number of geometrical regions. In fact, differently from Short Characteristics, which are discussed in section 4.6.4, MOC trajectories are continuous lines which cover the whole system. As a consequence, the trajectory spacing w_t^{\perp} is limited by the dimensions of the smallest computational regions in the geometry. In order to minimize the number of trajectories, a different approach may be implemented, which is, instead of using a constant quadrature weight, one may project the geometrical discontinuities on a plane perpendicular to the assigned direction. The space between two successive projections is then covered by n Gauss points, in order for the trajectories to be dense enough, [38]. Other powerful tracking strategies are discussed in [37].

4.6.4 . Method of Short Characteristics

The Method of Short Characteristics (**MOSC**) is based on current-coupled collision probabilities. Differently from CCCP, MOSC is able to handle anisotropic scattering sources, based on P_N representation. Examples of MOSC are common to both astrophysics and neutronics, with [39], [40], setting milestones for MOSC application to fission reactor physics. Short Characteristics generally assume piecewise constant (CSC) or linear (LSC) distributions of the neutron

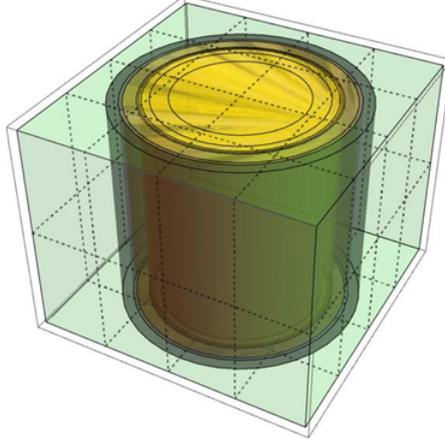


Figure 4.4: Heterogeneous Cartesian Cell (HCC).

source and surface angular flux, with some notable exceptions like [41], which assumes an exponential shape of the neutron source over the whole cell (ESC). Similarly to CCCP, the spatial domain is discretized by a *macro-mesh*, hosting homogeneous or heterogeneous nodes, which may be variously shaped, tetrahedral and Cartesian cells being a common choice, [40], [42]. In turn, each node may be equipped with an inner grid, comprising rings, [43], sectors or other geometries. In order to improve the representation of the interface angular flux, the cell sides may be partitioned into sub-surfaces. For the sake of brevity, the present analysis focuses on Heterogeneous Cartesian Cells (HCCs), of which Fig. 4.4 provides an example. The following discussion is restricted to polynomial bases. A more general formulation may be found in [43], pp. 259-260.

In the present section, the one-speed one-direction neutron transport operator \mathbf{L}_d^g is inverted within each Cartesian node by using linear short characteristics, LSC. Taking x as the coordinate along the trajectory, the integral form of the Boltzmann equation along trajectory t takes the form

$$\psi_d(\mathbf{r}_t^- + x\boldsymbol{\Omega}_d) = \psi_d(\mathbf{r}_t^-)e^{-\tau_t(x,0)} + \int_0^x q_d(\mathbf{r}_t^- + y\boldsymbol{\Omega}_d)e^{-\tau_t(x,y)} dy, \quad (4.85)$$

$\tau_t(x, y)$ being the optical path length along t from point $z = y$ to $z = x$,

$$\tau_t(x, y) = \int_y^x dz \Sigma(\mathbf{r}_t^- + z\boldsymbol{\Omega}_d). \quad (4.86)$$

Let us restrict Eq. (4.85) to one HCC and, analogously to MOC, suppose that

- The neutron source distribution is approximated by piece-wise linear functions in each region α within the HCC, i.e.,

$$q_{d,\alpha}(\mathbf{r}) = \sum_c P_{\alpha,c}(\mathbf{r}) q_{d,\alpha,c} = \mathbf{P}_\alpha(\mathbf{r}) \cdot \mathbf{q}_{d,\alpha}, \quad (4.87)$$

where \mathbf{P}_α is the linear basis and c identifies three/four spatial components in 2D/3D geometries, respectively. Note that \mathbf{P}_α is given by

$$\mathbf{P}_\alpha(\mathbf{r}) = \begin{bmatrix} 1 \\ \mathbf{r} - \mathbf{r}_\alpha \end{bmatrix}, \quad (4.88)$$

\mathbf{r}_α being the centre of mass of region α .

- Regions are homogeneous, i.e., cross-sections are represented by piece-wise constant functions, Eq. (4.74).

Due to Eq. (4.88), the constant and linear moments are orthogonal by construction and, as a consequence, the volume mass matrix of region α takes the form

$$\underline{\mathbf{M}}_\alpha = \langle \mathbf{P}_\alpha, \mathbf{P}_\alpha \rangle = \begin{bmatrix} 1 & \mathbf{0}^T \\ \mathbf{0} & \langle \mathbf{r} - \mathbf{r}_\alpha, \mathbf{r} - \mathbf{r}_\alpha \rangle \end{bmatrix}. \quad (4.89)$$

The spatial moments of the neutron source are defined as

$$\mathbf{q}_{d,\alpha} = \frac{[\underline{\mathbf{M}}_\alpha]^{-1}}{V_\alpha} \int_{V_\alpha} d\mathbf{r} \mathbf{P}_\alpha(\mathbf{r}) q_d(\mathbf{r}) = [\underline{\mathbf{M}}_\alpha]^{-1} \langle \mathbf{P}_\alpha, q \rangle, \quad (4.90)$$

where V_α represents both the integration domain and its measure. The characteristic approximation of MOSC consists in assuming that the interface angular flux may be reconstructed as piece-wise polynomial functions, one for each surface s of the boundary mesh,

$$\psi_{d,s}^\pm = \sum_b P_{s,b}(\mathbf{r}_t^\pm) \psi_{d,s,b}^\pm = \mathbf{P}_s(\mathbf{r}_t^\pm) \cdot \psi_{d,s}^\pm, \quad (4.91)$$

in which $\mathbf{P}_s(\mathbf{r}_t^\pm)$ is the linear basis associated to surface s , while b indexes two/three spatial components in 2D/3D geometries, respectively. Note that \mathbf{P}_s is provided by

$$\mathbf{P}_s(\mathbf{r}_t^\pm) = \begin{bmatrix} 1 \\ \mathbf{r}_t^\pm - \mathbf{r}_s \end{bmatrix}, \quad (4.92)$$

where \mathbf{r}_s is the centre of mass of surface s . Due to Eq. (4.92), the constant and linear surface moments are orthogonal, thus elements $m_{i,1}$ and $m_{1,j}$, with $i, j > 1$, of the surface mass matrix are equal to zero. Furthermore, as the boundary grid consists of Cartesian meshes and each local frame of reference is centered at \mathbf{r}_s , all surface mass matrices are diagonal.

The surface moments of the interface angular flux read as

$$\underline{\mathbf{M}}_s \psi_{d,s}^\pm = (\mathbf{P}_s, \psi_d)^\pm = \frac{|\boldsymbol{\Omega}_d \cdot \mathbf{n}_s|}{A_s} \int_{A_s} d\mathbf{r}^\pm \mathbf{P}_s(\mathbf{r}^\pm) \psi_d(\mathbf{r}^\pm), \quad (4.93)$$

where A_s represents both the sub-surface integration domain and its measure and (\cdot, \cdot) indicates a surface scalar product. Approximating the integral in Eq. (4.93) by numerical quadrature over the ensemble of trajectories $T_s(\boldsymbol{\Omega}_d)$ that intersect surface s and inverting the corresponding mass matrix, Eq. (4.93) becomes

$$\psi_{d,s}^+ = \frac{[\underline{\mathbf{M}}_s]^{-1}}{A_{s,d}} \sum_{t \in T_s(\boldsymbol{\Omega}_d)} w_t^\perp \mathbf{P}_s(\mathbf{r}_t^+) \psi_d(\mathbf{r}_t^+), \quad (4.94)$$

in which the trajectory weight w_t^\perp is provided by

$$w_t^\perp = |\boldsymbol{\Omega}_d \cdot \mathbf{n}_s| \Delta A_{s,t} = |\boldsymbol{\Omega}_d \cdot \mathbf{n}_{s'}| \Delta A_{s',t}, \quad (4.95)$$

where s and s' represent the outgoing and the incoming surfaces, respectively, while $\Delta A_{s,t}$ denotes the area of the projection of surface s onto the transverse plane. Note that \mathbf{n}_s coincides with one of the unit normal vectors of the Cartesian basis, i.e., \mathbf{e}_x , \mathbf{e}_y or \mathbf{e}_z , depending on the

orientation of the given HCC side.

Introducing Eq. (4.87) and (4.91) into Eq. (4.85) specialized for one HCC, one obtains the **balance** and **transmission equations** of MOSC, namely

$$\psi_{d,\alpha} = \sum_{s \in \Gamma^-(\Omega_d)} \mathbf{I}_{d,\alpha,s} \psi_{d,s}^- + \sum_{\beta \neq \alpha} \underline{\mathbf{C}}_{d,\alpha,\beta} \mathbf{q}_{d,\beta} + \underline{\mathbf{C}}_{d,\alpha,\alpha} \mathbf{q}_{d,\alpha}, \quad (4.96)$$

$$\psi_{d,s}^+ = \sum_{s' \in \Gamma^-(\Omega_d)} \underline{\mathbf{T}}_{d,s,s'} \psi_{d,s'}^- + \sum_{\alpha} \underline{\mathbf{E}}_{d,s,\alpha} \mathbf{q}_{d,\alpha}, \quad (4.97)$$

with incoming, collision, transmission and escape angular probability matrices given by

$$\mathbf{I}_{d,\alpha,s} = \frac{[\mathbf{M}_\alpha]^{-1}}{V_{\alpha,d}} \sum_{t \in T_\alpha(\Omega_d)} w_t^\perp \sum_{i \in I_{t,\alpha}(\Omega_d)} [\mathbf{G}_{d,t,\alpha,i} \times \mathbf{P}_s(\mathbf{r}_t^-)] e^{-\tau_{t,i}^<}, \quad (4.98)$$

$$\underline{\mathbf{C}}_{d,\alpha,\beta} = \frac{[\mathbf{M}_\alpha]^{-1}}{V_{\alpha,d}} \sum_{t \in T_\alpha(\Omega_d)} w_t^\perp \sum_{i \in I_{t,\alpha}(\Omega_d)} \sum_{\substack{j < i \\ j \in I_{t,\beta}(\Omega_d)}} [\mathbf{G}_{d,t,\alpha,i} \times \mathbf{F}_{d,t,\beta,j}] e^{-\tau_{i,j}}, \quad (4.99)$$

$$\begin{aligned} \underline{\mathbf{C}}_{d,\alpha,\alpha} &= \frac{[\mathbf{M}_\alpha]^{-1}}{V_{\alpha,d}} \sum_{t \in T_\alpha(\Omega_d)} w_t^\perp \sum_{i \in I_{t,\alpha}(\Omega_d)} (\underline{\mathbf{H}}_{d,t,\alpha,i} \\ &+ \sum_{\substack{j < i \\ j \in I_{t,\alpha}(\Omega_d)}} [\mathbf{G}_{d,t,\alpha,i} \times \mathbf{F}_{d,t,\alpha,j}] e^{-\tau_{i,j}}), \end{aligned} \quad (4.100)$$

$$\underline{\mathbf{T}}_{d,s,s'} = \frac{[\mathbf{M}_s]^{-1}}{A_{s,d}} \sum_{t \in T_\alpha(\Omega_d)} w_t^\perp [\mathbf{P}_s(\mathbf{r}_t^+) \times \mathbf{P}_{s'}(\mathbf{r}_t^-)] e^{-\tau_t}, \quad (4.101)$$

$$\underline{\mathbf{E}}_{d,s,\alpha} = \frac{[\mathbf{M}_s]^{-1}}{A_{s,d}} \sum_{t \in T_s(\Omega_d)} w_t^\perp [\mathbf{P}_s(\mathbf{r}_t^+) \times \sum_{i \in I_{t,\alpha}(\Omega_d)} e^{-\tau_{t,i}^>} \mathbf{F}_{d,t,\alpha,i}], \quad (4.102)$$

where

- $T_\alpha(\Omega_d)$ denotes the ensemble of trajectories along Ω_d intersecting region α ;
- $I_{t,\alpha}(\Omega_d)$ is the set of chords of trajectory t within region α ;
- $\Gamma^-(\Omega_d)$ is the HCC incoming boundary along direction Ω_d ;
- $\tau_{t,i}^> = \sum_{j > i} \tau_{t,j}$ is the down-stream optical path for chord i ;
- $\tau_{t,i}^< = \sum_{j < i} \tau_{t,j}$ is the up-stream optical path.

Eq. (4.98), (4.99), (4.101), (4.101) and (4.102) contain two four-component integral vectors, namely

$$\mathbf{G}_{d,t,\alpha,i} = \int_{x_{i-1}}^{x_i} \mathbf{P}_\alpha(\mathbf{r}_t^- + x\Omega_d) e^{-\Sigma_\alpha(x-x_{i-1})} dx, \quad (4.103)$$

$$\mathbf{F}_{d,t,\alpha,i} = \int_{x_{i-1}}^{x_i} \mathbf{P}_\alpha(\mathbf{r}_t^- + y\Omega_d) e^{-\Sigma_\alpha(x_i-y)} dy, \quad (4.104)$$

and one four-component integral matrix,

$$\underline{\mathbf{H}}_{d,t,\alpha,i} = \int_{x_{i-1}}^{x_i} \mathbf{P}_\alpha(\mathbf{r}_t^- + x\boldsymbol{\Omega}_d) \times \mathbf{F}_{d,t,\alpha,i}(x) dx, \quad (4.105)$$

$[x_{i-1}, x_i]$ being the chord interval along t .

The method of short characteristics ensures accurate representation of the neutron flux, [43]. Its modular geometry, consisting of current-coupled (heterogeneous) cells, allows for a major saving in terms of tracking, as, differently from MOC, trajectories are limited by the cell size. Cartesian regions are integrated in an exact way. The possibility to project discontinuities onto the transverse plane and introduce Gauss points between projections allows for a refined numerical quadrature in cells hosting more complex geometries, while, thanks to the locality of MOSC tracking, relaxing the trajectory spacing elsewhere. The *macro-mesh* hosting HCCs is also advantageous for domain decomposition implementations, which, in turns, allows for massive parallelization and, eventually, 3D direct-transport, [44].

On the downside, the method of short characteristics suffers from large memory occupation, due to the size of the angular probability matrices. In addition to this, MOSC introduces an approximation at the cell boundary, represented by Eq. (4.91), whose error may be reduced by increasing the number of sub-surfaces per cell side or the interface expansion order.

4.7 . ACCELERATIONS

Transport iterations (Sect. 4.4.2) generally suffer from slow convergence, which significantly affects the computational time. In order to boost the performance of transport solvers, several acceleration methods are available in literature. Sect. 4.7.1 provides a brief overview on linear and non-linear accelerations. Sect. 4.7.2 describes the Coarse Mesh Finite Difference (CMFD) method for outer iterations, as necessary for illustrating a subsequent part of the present work (Sect. 12.7).

4.7.1 . Linear vs Non-Linear Acceleration Schemes

Linear Accelerations. Once performed a transport iteration, namely

$$\mathbf{L}\psi^{(l+1/2)} = \mathbf{H}\psi^{(l)} + q, \quad (4.106)$$

where l denotes the iteration index, the residual of the scattering source is evaluated as

$$\rho^{(l+1/2)} = \mathbf{H}(\psi^{(l+1/2)} - \psi^{(l)}) \quad (4.107)$$

and injected into

$$(\widetilde{\mathbf{L} - \mathbf{H}})\Delta\Phi^{(l+1)} = \rho^{(l+1/2)}. \quad (4.108)$$

In Eq. (4.108), $(\widetilde{\mathbf{L} - \mathbf{H}})$ is a synthetic operator, defined as

$$(\widetilde{\mathbf{L} - \mathbf{H}})^{-1} \sim (\mathbf{L} - \mathbf{H})^{-1}. \quad (4.109)$$

Once solved Eq. (4.108), the angular flux is corrected as follows:

$$\psi^{(l+1)} = \psi^{(l+1/2)} + \Delta\Phi^{(l+1)} \quad (4.110)$$

An example of linear acceleration scheme is provided in [45], for the Boundary Projection Acceleration (BPA).

Non-Linear Accelerations. Once determined $\psi^{(l+1/2)}$ from Eq. (4.106), the cross sections are homogenized in space and condensed in energy, namely

$$\Sigma_\kappa = \frac{\int_\kappa d\kappa \Sigma \psi^{(l+1/2)}}{\int_\kappa d\kappa \psi^{(l+1/2)}}, \quad (4.111)$$

where κ is a coarse spatial/energy node. A non-linear diffusion equation of the kind

$$\left(\nabla \cdot \left(-\frac{1}{3\Sigma_\kappa} \nabla + \mathbf{v}_\kappa \right) + \Sigma_\kappa - \mathbf{H}_\kappa \right) \Phi_\kappa^{(l+1)} = q_\kappa, \quad (4.112)$$

is solved, \mathbf{v}_κ being the non-linear drift velocity,

$$\mathbf{v}_\kappa = \frac{\int_\kappa d\kappa \Omega \psi^{(l+1/2)} - \nabla \int_\kappa d\kappa \psi^{(l+1/2)}}{\int_\kappa d\kappa \psi^{(l+1/2)}}. \quad (4.113)$$

Finally, the angular flux is renormalized as

$$\psi^{(l+1)} = \psi^{(l+1/2)} \frac{\phi_c^{(l+1)}}{\int_c \psi^{(l+1/2)}}. \quad (4.114)$$

An example of non-linear acceleration scheme is provided in Sect. 4.7.2.

4.7.2 . Coarse Mesh Finite Differences

The Coarse Mesh Finite Difference method is a non-linear technique preserving the particle balance, [46]. It consists in applying finite differences to the diffusion equation, [47], on a coarse spatial/energy grid, in order to speed-up the convergence of the transport iterations. For the sake of simplicity, let us suppose that CMFD is applied to MOSC. The *macro-mesh* introduced in Sect. 4.6.4 is homogenized in space, i.e., the CMFD spatial grid may contain one or more MOSC cell per node. The energy grid, initially consisting of G fine groups, is condensed into H coarse groups, in order to decrease the computational cost.

The multigroup P_1 approximation of the Boltzmann equation reads as

$$\nabla \cdot \mathbf{J}^g(\mathbf{r}) + \Sigma^g(\mathbf{r})\Phi^g(\mathbf{r}) - \sum_{g' \neq g} \Sigma^{gg'}(\mathbf{r})\Phi^{g'}(\mathbf{r}) = \frac{1}{k} \sum_{g' \neq g} (\chi^\nu \Sigma)_f^{gg'}(\mathbf{r})\Phi^{g'}(\mathbf{r}). \quad (4.115)$$

Let us integrate both sides of Eq. (4.115) over V_ζ , ζ being a CMFD node, comprising one or more HCCs, and divide by the volume V_ζ of node ζ . The energy variable is condensed into H coarse groups. By applying the divergence theorem on the escape term, namely

$$\int_{V_\zeta} d^3r \nabla \cdot \mathbf{J}^h = \oint_{\partial V_\zeta} d^3r \mathbf{J} \cdot \mathbf{n}, \quad h = 1, \dots, H, \quad (4.116)$$

one obtains

$$\sum_{s \in \partial V_\zeta} \frac{A_{\zeta,s}}{V_\zeta} \mathbf{J}_{\zeta,s}^h + \Sigma_\zeta^h \Phi_\zeta^h - \sum_{h' \neq h} \Sigma_\zeta^{hh'} \Phi_\zeta^{h'} = \frac{1}{k} \sum_{h' \neq h} (\chi^\nu \Sigma)_{f,\zeta}^{hh'} \Phi_\zeta^{h'}, \quad (4.117)$$

where $A_{\zeta,s}$ is the area of side s of node ζ , $\mathbf{J}_{\zeta,s}^h$ is the average neutron current of group h through surface s , Φ_ζ^h is the average scalar flux of group h in cell ζ and $\Sigma_{\zeta,r}^h$ is the flux-weighted homogenized cross section of group h in cell ζ for reaction r . More specifically, provided that $\Phi^{(e+1/2)}$

denotes the transport scalar flux, i.e., the zero-th order spatial moment of $\Phi_{l=0,m=0}$, the homogenized cross sections read as

$$\Sigma_{\zeta}^h = \frac{\sum_{g \in h} \sum_{n \in \zeta} \sum_{\alpha \in n} V_{n,\alpha} (\Sigma_{n,\alpha}^g - \Sigma_{n,\alpha,0}^{gg}) \phi_{n,\alpha}^{g,(e+1/2)}}{\sum_{g \in h} \sum_{n \in \zeta} \sum_{\alpha \in n} V_{n,\alpha} \phi_{n,\alpha}^{g,(e+1/2)}}, \quad (4.118)$$

$$\Sigma_{\zeta}^{hh'} = \frac{\sum_{g \in h} \sum_{g' \in h'} \sum_{n \in \zeta} \sum_{\alpha \in n} V_{n,\alpha} \Sigma_{n,\alpha,0}^{gg'} \phi_{n,\alpha}^{g',(e+1/2)}}{\sum_{g' \in h'} \sum_{n \in \zeta} \sum_{\alpha \in n} V_{n,\alpha} \phi_{n,\alpha}^{g',(e+1/2)}}, \quad (4.119)$$

$$(\chi \nu \Sigma)_{f,\zeta}^{hh'} = \frac{\sum_{g \in h} \sum_{g' \in h'} \sum_{n \in \zeta} \sum_{\alpha \in n} V_{n,\alpha} \sum_{i \in \alpha} \chi_i^g (\nu \Sigma)_{i,\alpha}^{g'} \phi_{n,\alpha}^{g',(e+1/2)}}{\sum_{g' \in h'} \sum_{n \in \zeta} \sum_{\alpha \in n} V_{n,\alpha} \phi_{n,\alpha}^{g',(e+1/2)}}, \quad (4.120)$$

in which $n \in \zeta$ is an index for the HCCs composing node ζ and $\alpha \in n$ is a region of n , while i indicates a fissionable isotope. Note that $\Sigma_{n,\alpha,0}^{gg'}$ represents the homogenized isotropic (P_0) scattering cross section from group g' to group g , in region $\alpha \in n$. $(\chi \nu \Sigma)_{f,\zeta}^{hh'}$ is the homogenized fission matrix. The neutron current $J_{\zeta,s}^h$ is obtained by an artificial finite-difference Fick's law,

$$J_{\zeta,s}^h = -D_{\zeta,s}^h (\phi_{\zeta,s}^h - \phi_{\zeta}^h) + \bar{D}_{\zeta,s}^h (\phi_{\zeta,s}^h + \phi_{\zeta}^h), \quad (4.121)$$

where $\phi_{\zeta,s}^h$ is the average interface scalar flux on surface $s \in \partial V_{\zeta}$, while $D_{\zeta,s}^h$ is the finite-difference diffusion coefficient,

$$D_{\zeta,s}^h = \frac{2}{3 L_{\zeta,s} \Sigma_{\zeta,tr}^h}, \quad (4.122)$$

where $L_{\zeta,s}$ is the node thickness along normal unit-vector of surface s . The transport-corrected cross section of node ζ reads as $\Sigma_{\zeta,tr}^h = \Sigma_{\zeta}^h - \Sigma_{\zeta,s,l=1}^{hh}$. By imposing current continuity between neighbouring cells, in few steps, one can get rid of the interface scalar flux in Eq. (4.121), thus obtaining

$$J_{\zeta,s}^h = \frac{(D_{\zeta',s}^h - \bar{D}_{\zeta',k}^h)(D_{\zeta,s}^h + \bar{D}_{\zeta,s}^h)}{(D_{\zeta,s}^h - \bar{D}_{\zeta,s}^h) + (D_{\zeta',s}^h - \bar{D}_{\zeta',s}^h)} \phi_{\zeta}^h - \frac{(D_{\zeta,s}^h - \bar{D}_{\zeta,s}^h)(D_{\zeta',s}^h + \bar{D}_{\zeta',s}^h)}{(D_{\zeta,s}^h - \bar{D}_{\zeta,s}^h) + (D_{\zeta',s}^h - \bar{D}_{\zeta',s}^h)} \phi_{\zeta'}^h. \quad (4.123)$$

As mentioned in the opening, CMFD aims at preserving the particle balance. To do this, the drift coefficient $\bar{D}_{\zeta,s}^h$ is computed on-the-fly to adjust Eq. (4.121) by successive iterations, by means of

$$\bar{D}_{\zeta,s}^h = \left[\frac{J_{\zeta,s}^{h,(i+1/2)} + D_{\zeta,s} (\phi_{\zeta,s}^{h,(i+1/2)} - \phi_{\zeta}^{h,(i+1/2)})}{(\phi_{\zeta,s}^{h,(i+1/2)} + \phi_{\zeta}^{h,(i+1/2)})} \right]. \quad (4.124)$$

5 - REACTOR CORE CALCULATION SCHEMES

5.1 . INTRODUCTION

A reactor core is a complex system, involving complicated three-dimensional geometrical patterns, often with high level of material heterogeneity. As mentioned in chapter 3, neutron interaction probabilities strongly depend on the relative kinetic energy of the collision, which is affected by materials' temperature. Also, due to nuclear reactions, material isotopic concentrations evolve in time, thus adding a further level of complexity to reactor calculations.

For the sake of simplicity, let us restrict the focus on neutronics. Let us provide an estimate of the total number of volumetric degrees of freedom (DOF) for a linear MOSC calculation, with P_1 scattering. The most widespread technology, i.e., the Pressurized Water Reactor (PWR), is about 400 – 450 *cm* high. Depending on the specific benchmark, the number of fuel assemblies may range between 250 and 260 units. Each assembly contains around 260 fuel rods, 25 guide tubes, in which the control rod tip may be located at different axial positions, and 1 guide tube hosting a fission chamber. Let us assume that

- the upper and lower axial reflectors measure about 10 – 20 *cm* each;
- the fuel plenum is approximately 15 *cm* high;
- the top and bottom nozzles measure 10 – 15 *cm*;
- the active core length is about 370 and 380 *cm*.

In the vicinity of the active region, the axial reflector has to be finely discretized, with a step size of the order of the neutron mean free path ($\mu \lesssim 1$ *cm*), i.e., the inverse of the total cross section. The active length generally requires a finer subdivision in the proximity of the axial reflector and in the spacer grids, where large flux gradients are observed, the total number of axial planes being approximately 200 – 250.

Each fuel pin-cell contains at least 7 radial regions, i.e., 4 UOX (or MOX) self-shielding rings, 1 region for the air-gap, 1 ring for the fuel clad and 1 region for the surrounding moderator. The control rods may require a finer radial mesh: they contain at least 9 radial regions, namely 4 self-shielding AIC/B₄C/Pyrex rings, the air-gap, the control-rod clad, the water gap between the control rod and the guide tube, the guide tube and the surrounding moderator. An inner Cartesian or azimuthal grid may be necessary to improve the numerical accuracy. Overall, one may estimate a total of $5 - 6 \cdot 10^5$ computational regions per assembly, and thus around $1.5 \cdot 10^8$ regions for the whole core. This can be reduced by an order of magnitude, if 1/8-th core symmetry is applied. In addition, each region comprises 4 spatial moments per harmonics. Assuming P_1 scattering, the total number of moments raises to 16. One may suppose that the solid angle is discretized by ~ 100 directions (for instance, the S_8 level-symmetric angular quadrature formula includes 10 directions per octant). Finally, considered an energy mesh of 100 – 300 energy groups, the overall number of DOF will be of the order of $10^{12} - 10^{13}$.

In order to deal with a problem of this size, two different strategies are used in industrial codes, i.e., the **two-step strategy** and the **3D direct full-core transport**. The former comprises two stages, the first one being a lattice simulation, which computes few-groups libraries

for a variety of operating conditions (or *physical states*) of a slice of the whole reactor core. Each state corresponds to a set of physical parameters, typically, the fuel and moderator temperature, the moderator mass density, the boron concentration and the burn-up. Each lattice calculation consists in solving the k -eigenvalue formulation of the transport equation over the infinite spatial repetitions of an elementary unit, i.e., the considered core slice, along the radial plane, with few hundreds of energy groups and angular directions (for discrete-ordinates transport solvers, like APOLLO3® IDT-MOSC and TDT-MOC) or spherical harmonics (for P_N -based transport solvers, like APOLLO3® NYMO, [48]). A core step follows, consisting in the interpolation of the multi-parametric library data, in order to meet the actual state of the reactor core. A 3D full-core diffusion simulation (or, more generally, an approximate transport calculation, driven by a *synthetic* or *low-order operator*) is run on a coarse spatial mesh and energy grid, to determine the neutron flux and reaction rates of the whole reactor core, imposing the actual boundary conditions.

Conversely, the other modelling strategy, namely the 3D direct transport approach, consists in solving the Boltzmann equation for the full reactor core, with detailed geometry, the major source of error being the multi-group cross section library, due to the approximate self-shielding model. An example of 3D direct transport code is provided in [49]. In order to cope with a problem of this size, including a large number of DOF, one may resort to domain decomposition methods (DDM) [50], which allow for distributing the unknowns on several nodes in massive parallel calculations. DDM consists in breaking the original boundary-value problem on the whole domain into a set of coupled problems on smaller spatial domains. The global solution over the ensemble of smaller-sized problems converges to the one of the unpartitioned problem, by iterative steps, exchanging interface neutron currents between neighbouring sub-domains.

An alternative to 3D direct transport is the 2D/1D fusion method, [51], which combines radial 2D MOC and a lower order transport operator along the z -axis. The two solutions are coupled by axial and radial transverse currents and 3D coarse mesh finite differences, providing both acceleration and stability to the iterative solution, [52].

In this chapter, a brief overview of the main reactor core calculation schemes is provided. In Sect. 5.2, the two-step approach is discussed. Sects. 5.3-5.4 respectively list some remarkable limitations of the this strategy and propose 3D direct transport and 2D/1D fusion, as an alternative to two-step calculations. Finally, in Sect. 5.5, the rod-cusping effect is illustrated, together with some notable remedies, including the equivalence method and the subplane approach.

5.2 . TWO-STEP APPROACH

As mentioned in Sect. 5.1, a reactor core is a relatively large system, comprising several heterogeneous elements of high geometrical complexity. The direct transport solution, attained by high-fidelity deterministic or stochastic methods, is extremely expensive in terms of CPU time and memory requirements. The most established strategy for generating the power distribution is based on dimensionality reduction and goes under the name of **two-step approach**. This technique takes advantage of the high regularity of the reactor core, which, despite the inherent geometrical complexity, is composed of repetitions of elementary units, namely the **fuel-assemblies**. The first step, known as **lattice calculation** solves the detailed neutron transport equation on each fuel-assembly, with simplified boundary conditions, for the ensemble of all possible physical states. Each physical state is a combination of local and global

parameters, including the fuel and moderator temperature [K], the moderator mass density [g/cm^3], the boron concentration [g/cm^3], the burn-up [MWd/t], the control rod positions. The k -eigenvalue is adjusted afterwards, in order to match reactor's steady-state conditions (i.e., $k = 1$), by resorting to the B^2 -eigenvalue formalism. The output of the first step consists of homogenized and energy-condensed multi-parametric cross-section libraries to feed into the subsequent simulation step. Indeed, an intermediate calculation, based on equivalence theory, with superhomogénéisation (SPH) or flux-discontinuity factors, has to be performed, in order to preserve the reaction rates. Sect. 5.2.1 provides some more detail on lattice simulations. In Sect. 5.2.2, the second step, i.e., the **core calculation** is discussed. Taking advantage of the homogenized few-group cross sections provided by the lattice calculation, the reactor core is modelled as a collection of fuel-assemblies containing lumped pin-cells. The determination of the neutron flux and reaction rates distributions over the whole core is attained by discretizing a low-order transport operator, e.g., diffusion or SP_N , on a coarse spatial and energy mesh, and imposing high-fidelity boundary conditions on the reactor border.

5.2.1 . Descriptive Elements of Lattice Calculations

In the following paragraphs, a brief summary of the main stages composing state-of-the-art lattice calculations is proposed. The interested reader is referred to [5] for more details.

Cross-libraries and self-shielding. Differently from Monte Carlo, deterministic solvers are not able to treat continuous-energy cross sections. Hence, for each type of fission reactor, e.g., PWR, BWR, VVER, RBMK, etc., the experimental nuclear data are pre-processed, once for all, to provide reference multigroup cross section libraries, depending on a set of physical parameters, namely, the nuclide, the nuclear reaction, the energy group and the temperature. Due to cross section resonances (Fig. 3.1), the neutron flux exhibits abrupt variations, thus requiring the application of a preliminary **self-shielding** model for each type of fuel-assembly, for the main resonant isotopes. The objective of a self-shielding calculation is to provide multigroup cross sections, verifying reaction rate conservation, for the considered fuel-assembly. In practice, a 2D transport simulation is performed for every assembly type, assuming reflection boundary conditions. A mathematical model, like the Fine-Structure, [111], Tone or Subgroup methods, [5], is applied in order to comply such task. More details on the self-shielding models are beyond the scope of the present work, the interested reader is referred to [27].

Infinite lattice model. Once determined the self-shielded cross sections, the k -eigenvalue formulation of the multigroup transport equation is solved for each assembly-type, in real geometry, assuming reflection boundary conditions on the radial plane and thus neglecting the net exchange between assemblies. A 3D detailed model may be applied, imposing vacuum boundary conditions along the z-axis and introducing two layers of axial reflector, one per axial side, to "back-scatter" a fraction of neutrons towards the active region. Nevertheless, the axial variation of the neutron flux is generally disregarded, as less significant than the radial one, and a 2D infinite lattice model is applied, where the elementary unit is the considered fuel-assembly. The 2D multigroup Boltzmann equation may be solved by deterministic methods, a common choice being applying discrete ordinates for the angle and MOC for the spatial variable. In order to reduce the number of unknowns, one may take advantage of the geometrical properties of the fuel-assembly, thus applying 1/8-th symmetry for PWRs or 1/6-th symmetry for VVERs.

Buckling leakage model. At this stage, the effective multiplication factor k might be significantly different from 1. Nonetheless, a fission reactor is run in (quasi) steady state conditions. As neutron leakage has not been taken into account by explicit boundary conditions, the fine transport spectrum has to be adjusted artificially. A **critical leakage model** may be introduced in order to enforce steady-state regime, [5]. To do this, let us suppose that the actual flux distribution ψ may be factorized as follows:

$$\psi(\mathbf{r}, \boldsymbol{\Omega}, E) = \Phi(\mathbf{r}) \cdot \zeta(\mathbf{r}, \boldsymbol{\Omega}, E), \quad (5.1)$$

where ζ is the periodic fundamental mode, derived by MOC application to the infinite-lattice neutron transport problem. $\Phi(\mathbf{r})$ is a macroscopic spatial shape, satisfying a Laplace equation,

$$\nabla^2 \Phi(\mathbf{r}) + B^2 \Phi(\mathbf{r}) = 0, \quad (5.2)$$

in which the distribution curvature B^2 is the so-called **buckling**, that is adjusted iteratively up to $k = 1$. This curvature represents an average spatial profile of the neutron flux over the considered assembly in the full reactor core. Due to Eq. (5.1), Eq. (5.2) reads as

$$\psi(\mathbf{r}, \boldsymbol{\Omega}, E) = \zeta(\mathbf{r}, \boldsymbol{\Omega}, E) e^{i\mathbf{B} \cdot \mathbf{r}}. \quad (5.3)$$

At this point, a **homogeneous fundamental mode** approximation may be introduced, namely $\zeta \simeq \zeta(\boldsymbol{\Omega}, E)$, which reduces the heterogeneous assembly to a homogeneous paste, for computing the neutron leakage rates. The prior factorization is replaced into the Boltzmann equation for a finite and homogeneous system, with P_1 scattering. By few rearrangements, a set of B_1 equations is obtained, namely

$$\left(\Sigma(E) + D(B, E) B^2 \right) \zeta(E) = \int_0^\infty \Sigma_{s,0}(E' \rightarrow E) \zeta(E') dE' + \frac{\chi(E)}{k} \int_0^\infty (\nu_f \Sigma_f)(E') \zeta(E') dE', \quad (5.4)$$

$$D(B, E) = \frac{1}{\gamma(B, \Sigma) \Sigma(E)} \left(\frac{1}{3} + \frac{1}{\zeta(E)} \int_0^\infty \Sigma_{s,1}(E' \rightarrow E) D(B, E') \zeta(E') dE' \right), \quad (5.5)$$

where $\gamma(B, \Sigma)$ is a function of both the buckling and the total macroscopic cross section and may be approximated by a Taylor series expansion, as reported in [5]. Eq. (5.4) and (5.5) are approached by applying the multigroup formalism. As they contain three unknowns, namely ψ^g , D^g and B^2 , an additional condition is needed, which is provided by normalizing the fission reaction rate to 1. The buckling is thus determined by successive iterations, under the steady-state constraint $k = 1$. Once the fission reaction rate has converged to 1, within a given numerical tolerance, one has to verify that the steady-state condition is satisfied. If this is not the case, the total cross section is redefined as

$$\tilde{\Sigma}_t^g(\mathbf{r}) = \Sigma_t^g(\mathbf{r}) + D^g B^2, \quad (5.6)$$

and a heterogeneous calculation of the kind

$$\left(\boldsymbol{\Omega} \cdot \nabla + \tilde{\Sigma}_t^g \right) \psi^g = q^g \quad (5.7)$$

is performed, with reflection boundary conditions. The described procedure is repeated until convergence, [38].

Homogenization and energy condensation. The heterogeneous critical flux ψ^g , with $g = 1, \dots, G$, is used to define homogenized few-group cross section libraries, for every combination of a set of physical parameters, namely the fuel and moderator temperature, the moderator mass density, the boron concentration, the control rod positions and the burn-up. Note that the same set of problems is solved for each assembly type. Once simulated the last depletion step, a multi-parametric cross section library is returned. The reader should keep in mind that, despite for all media the homogenized cross sections contain the contributions of all nuclides, the distinction between microscopic cross sections and isotopic concentrations is preserved only for a few representative isotopes (i.e., the most significant fission products), whereas a fictitious nuclide summarizes the presence of all the others, [53].

5.2.2 . Descriptive Elements of Core Calculations

The objective of the second step is to determine the neutron flux and reaction rates distributions over the whole reactor core. A calculation over assemblies of homogenized pin-cells (or over homogenized assemblies) is performed, with the low-order transport operator being discretized by the finite-difference method, the analytical nodal (ANM) or nodal expansion method with transverse leakage approximation (NEM), [54], or the finite-element method (FEM), [55]. Differently from lattice simulations, no critical leakage model is required, as the boundary conditions may be represented with high accuracy.

Let us introduce the steady-state neutron balance over group h ,

$$\nabla \cdot \mathbf{J}^h(\mathbf{r}) + \Sigma^h(\mathbf{r})\Phi^h(\mathbf{r}) = \sum_{h' \neq h} \Sigma^{hh'}(\mathbf{r})\Phi^{h'}(\mathbf{r}) + \frac{1}{k} \sum_{h' \neq h} (\chi \nu \Sigma)_f^{hh'}(\mathbf{r})\Phi^{h'}(\mathbf{r}), \quad h = 1, \dots, H, \quad (5.8)$$

H being the number of condensed-energy groups (generally, $H = 2$, but may assume values up to a few tens). In order to solve Eq. (5.8), a relation between the multigroup neutron flux and current density is needed. Often, this condition is provided by the Fick law, leading to the diffusion approximation,

$$\mathbf{J}^h(\mathbf{r}) = -\mathbf{D}^h(\mathbf{r})\nabla\Phi^h(\mathbf{r}), \quad (5.9)$$

where $\mathbf{D}^h(\mathbf{r})$ is a tensor containing the directional diffusion coefficients.

The neutronics solver, based on 3D nodal diffusion with few energy groups, is coupled with a thermal-hydraulics code, whose solution may consist of 1D sub-channels exchanging heat and mass (Fig. 5.1).⁴ Each sub-channel may contain a certain number of fuel pins, discretized into multiple rings and surrounded by an average coolant, [56]. Despite the increasing interest for detailed thermal-hydraulics solutions, most reactor core calculations resort to coarse radial discretizations, where the fuel assembly is treated as a whole, without sub-channel refinement and transverse exchange. Moreover, heat conduction is often reduced to a single representative pin per assembly, with several rings to account for the sharp gradients of the radial temperature profile. The thermal-hydraulics feedback is responsible for a change in the cross sections of the neutron diffusion calculation. In return, the energy released by fission reactions impacts the temperature and mass density distributions, leading to a mutual interaction between neutronics and thermal-hydraulics, and thus to a multiphysics problem. Each steady-state solution requires, as a consequence, a coupled approach, where the boron concentration and control rod positions that make the core critical are searched. In PWRs, the boric

⁴An alternative strategy consists in implementing an inner thermal-hydraulic module.

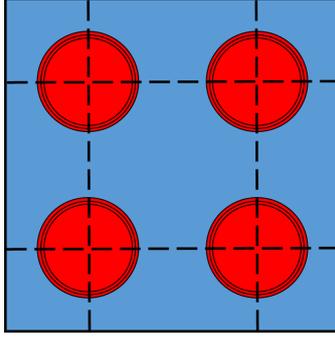


Figure 5.1: Thermal-hydraulics sub-channel mesh, represented by dashed lines.

acid is dissolved in the coolant. Assuming a uniform boron concentration C_B [ppm] over the whole moderator, the following formula holds, [57],

$$C_B = \frac{\rho_{H_3BO_3}(z) w_B}{\rho_{H_3BO_3}(z) + \rho_{H_2O}(z)} \cdot 10^6, \quad (5.10)$$

where $\rho_{H_3BO_3}$ and ρ_{H_2O} [g/cc] are respectively the boric acid density and the water density of the borated water, in the local coolant cell z , with $z = 1, \dots, N_z$, while w_B is the weight fraction of boron in the boric acid. The critical boron concentration can be determined by an iterative formula, involving the multiplication factor, i.e.,

$$C_B^{(n)} = \frac{C_B^{(n-1)} - C_B^{(n-2)}}{k^{(n-1)} - k^{(n-2)}} (1 - k^{(n-2)}) + C_B^{(n-2)}, \quad (5.11)$$

n being the iteration index.

5.3 . LIMITATIONS OF THE TWO-STEP STRATEGY AND 3D DIRECT TRANSPORT

As mentioned in Sect. 5.2, the two-step approach introduces a number of approximations, as, to name a few,

- The detailed transport solution is generally searched only for a 2D lattice, with simplified radial boundary conditions;
- In the lattice step, radial leakage is introduced artificially, by an approximate homogeneous critical model;
- The flux distribution is reconstructed as the product of a macroscopic profile and a periodic transport solution. This factorization is exact only for a core of identical fuel assemblies;
- The reaction rates over the whole core result from a diffusion approximation, which may fail for small reactors;
- The homogenized reflector cross sections, which require a dedicated approximate treatment, as no nuclear fission occur in this region and vacuum boundary conditions are imposed on at least one side.

The limitations of the two-step approach, together with the increasing availability of high performance computers, justify the growing interest of fission reactor physicists in full 3D transport-based simulations. The development of high-fidelity tools allows following local sharp gradients with great level of detail, at the expense of a large computational effort, the multigroup approximation and the self-shielding model being the sole factors which may impact the numerical precision. Due to the size of the problem, with thousands of billions of unknowns, the numerical solution, often based on the method of characteristics, relies on DDM for massive parallel calculations on high performance computers.

An example of 3D direct transport is provided by the OpenMOC solver, developed at the Massachusetts Institute of Technology (MIT). In [58], the numerical accuracy was tested by comparisons with a Monte Carlo reference solution. A total of 92480 CPU cores were needed for a runtime of 7.76 *h*, to simulate the full core 3D BEAVRS benchmark, [59].

Despite the growing interest in 3D direct transport, industrial codes still rely on the two-step approach or the 2D/1D fusion method, as less expensive in terms of computational (and economical) cost.

5.4 . 2D/1D FUSION METHOD

The 2D/1D Fusion method offers an alternative strategy to the two-step and 3D direct transport approach, for high-fidelity numerical solutions of the neutral-particle Boltzmann equation over the whole reactor core. Its first application dates back to the CRX code, [51], with several successive implementations, of which DeCART [60], nTRACER [61] and MPACT [52] are notable examples. The basic idea behind 2D/1D Fusion is to preserve a detailed solution on the x-y plane, while using a low-order transport model along the z-axis. This assumption is motivated by observing that the axial geometry is generally less heterogeneous than the radial one. The 2D/1D equations are derived from the multigroup Boltzmann equation, by approximating the streaming operator as follows

$$\boldsymbol{\Omega} \cdot \nabla \psi^g(\mathbf{r}, \boldsymbol{\Omega}) \simeq \left(\Omega_x \frac{\partial \psi^g(\mathbf{r}, \boldsymbol{\Omega})}{\partial x} + \Omega_y \frac{\partial \psi^g(\mathbf{r}, \boldsymbol{\Omega})}{\partial y} \right) + \frac{1}{4\pi} \frac{\partial J_z^g(\mathbf{r})}{\partial z} \quad (5.12)$$

for the radial equation and

$$\boldsymbol{\Omega} \cdot \nabla \psi^g(\mathbf{r}, \boldsymbol{\Omega}) \simeq \frac{1}{4\pi} \left(\frac{\partial J_x^g(\mathbf{r})}{\partial x} + \frac{\partial J_y^g(\mathbf{r})}{\partial y} \right) + \Omega_z \frac{\partial \psi^g(\mathbf{r}, \boldsymbol{\Omega})}{\partial z} \quad (5.13)$$

for the axial one. Replacing the streaming operator by Eq. (5.12) and integrating over a finite step Δz along the z-axis, one obtains the 2D equation,

$$\left(\Omega_x \frac{\partial \psi_{\Delta z}^g(x, y, \boldsymbol{\Omega})}{\partial x} + \Omega_y \frac{\partial \psi_{\Delta z}^g(x, y, \boldsymbol{\Omega})}{\partial y} \right) + \Sigma_{\Delta z}^g(x, y) \psi_{\Delta z}^g(x, y, \boldsymbol{\Omega}) = q_{\Delta z}^g(x, y, \boldsymbol{\Omega}) - L_{\Delta z}^g(x, y), \quad (5.14)$$

where $\psi_{\Delta z}^g(x, y, \boldsymbol{\Omega})$ and $q_{\Delta z}^g(x, y, \boldsymbol{\Omega})$ denote respectively the axial average of the angular flux and neutron source, over a slice Δz along the z-axis. Note that $L_{\Delta z}^g(x, y)$ in Eq. (5.14) represents the axial transverse leakage. Conversely, substituting Eq. (5.13) in the original multigroup transport equation, integrating over Δxy on the radial plane and over $[0, 2\pi]$ along the azimuthal angle, one obtains the 1D equation,

$$\frac{\mu}{2} \frac{\partial \psi_{\Delta xy}^g(z, \mu)}{\partial z} + \Sigma_{\Delta xy}^g(z) \psi_{\Delta xy}^g(z, \mu) = q_{\Delta xy}^g(z, \mu) - L_{\Delta xy}^g(z), \quad (5.15)$$

μ being the cosine of the polar angle, $\psi_{\Delta xy}^g(z, \mu)$ and $q_{\Delta xy}^g(z, \mu)$ the radial average of the angular flux and neutron source, respectively, and $L_{\Delta xy}^g(z)$ is radial transverse leakage term. Note that Eq. (5.14) and (5.15) are coupled through the radial and axial leakage. The solution is obtained iteratively, by solving Eq. (5.14) and computing the radial leakage, which is then provided to Eq. (5.15), whose solution allows to evaluate the axial leakage, which is, in turn, required in Eq. (5.14).

Eq. (5.14) has to be solved on a highly heterogeneous domain, i.e., the radial plane. For this purpose, the 2D Method of Characteristics is applied on discrete directions (S_N method), in order to attain high numerical accuracy on arbitrary geometries. Conversely, as the axial geometry is more homogeneous, Eq. (5.15) is generally solved by applying the diffusion approximation or by SP_N , with the latter being able to represent the angular dependence of the flux by Legendre polynomial expansion of order N .

In order to discuss multiphysics applications based on the 2D/1D Fusion method, let us restrict the focus on the VERA Core Simulator. This comprises three solvers, i.e., MPACT for neutron transport, COBRA-TF for thermal-hydraulics and ORIGEN for nuclide transmutations, [62]. Regarding the neutronics/thermal-hydraulics coupling, MPACT receives the temperature and density scalar fields produced by COBRA-TF, and, in return, it provides COBRA-TF with the power-density distribution. The spatial discretizations of MPACT and COBRA-TF are different: the former is based on combining Cartesian and azimuthal grids, while the latter relies on the sub-channel discretization depicted in Fig. 5.1. Hence, the physical quantities are mapped from one solver to another in such a way that the scalar fields (i.e., temperatures, concentrations, etc.) are conserved, as illustrated in [62]. To do this, a pin-cell coupling model is introduced, using an intermediate conformal pin-cell geometry to exchange data between MPACT and COBRA-TF.

One of the most innovative calculation methodologies proposed in the VERA Core Simulator is sketched in Fig. 5.2. The precomputed multigroup (MG) cross-section library, containing microscopic cross sections and resonance integrals, as a function of temperature and neutron kinetic energy, includes 295 isotopes and 51 energy groups, of which 22 are in the resonant region. The self-shielding calculation is performed for the resonant groups, for 49 resonant isotopes. The macroscopic cross section library is updated as depicted in Fig. 5.2 at each outer iteration, after updating the temperature and density fields. In other words, MPACT and COBRA-TF exchange information at each outer iteration, without converging separately.

The 2D/1D Fusion method is an attractive alternative to dimensionality reduction methods based on the two-step approach, as it does not require the use of homogenization and energy condensation techniques, which could degrade the numerical accuracy of the simulation. Nevertheless, the application of 2D/1D to highly heterogeneous axial geometries may suffer from some limitations, due to 1D diffusion. This is the case of

- BWR cores experiencing severe void fraction heterogeneities;
- Partially rodded assemblies, with heterogeneous AIC/B_4C control rods, with Fig. 7.27 providing a practical example;
- Axially heterogeneous fast breeder reactor cores.

In particular, in order to obtain accurate results, each MOC plane should be axially homogeneous. In principle, this task may be achieved by meshing the axial geometry, in such a way

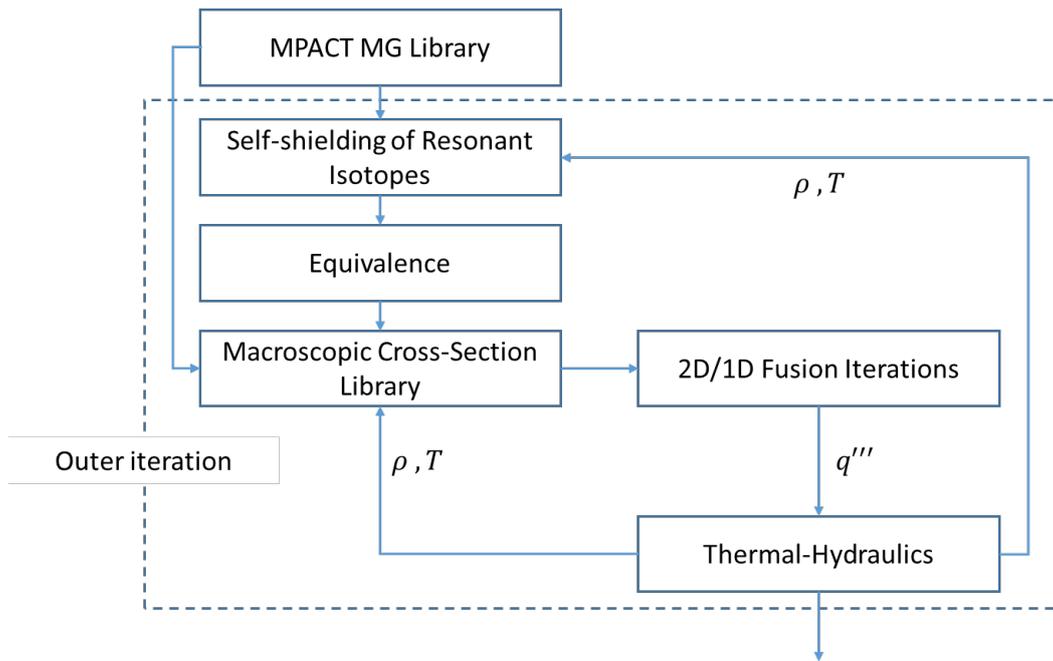


Figure 5.2: VERA Core Simulator: cross section update and outer iteration scheme.

that the vertical heterogeneities line up with the node boundaries. Nonetheless, this may cause some issues when simulating control rods successive positions, as control rods move with a step-size of $1 \div 2 \text{ cm}$. It is clear that MOC plane refinement is not a viable option, as the CPU time would increase dramatically. The same issue is experienced in other nodal solvers and calls for dedicated remedies, which are illustrated in Sect. 5.5.

5.5 . CONTROL ROD MOVEMENT AND CUSPING EFFECT

A control rod is a device that allows to regulate the evolution of the fission reaction chain. Its composition comprises nuclides like boron, cadmium, indium, silver, hafnium isotopes , which are strong neutron absorbers, with the lighter nuclides (e.g., ^{10}B , ^{11}B) undergoing (n, α) -reactions and the heavier nuclides (e.g, ^{113}Cd , ^{115}In , ^{107}Ag , ^{108}Ag) interacting by radiative capture.

As piece-wise cross sections are not allowed in traditional solver, when the control rod tip position does not align with the node boundary, the absorber has to be homogenized with the moderator underneath. If the node cross sections are homogenized by simple volume-weighting, an artificial depression is observed in the neutron flux axial profile, with a consequent under-prediction of the reactivity curve, as shown in Fig. 5.3. More details on the subject are provided in Chapter 8.

In order to reduce the cusping effect, several workarounds have been introduced in the last decades. In the following, a brief overview of some notable methods is proposed, [63], [64], [65].

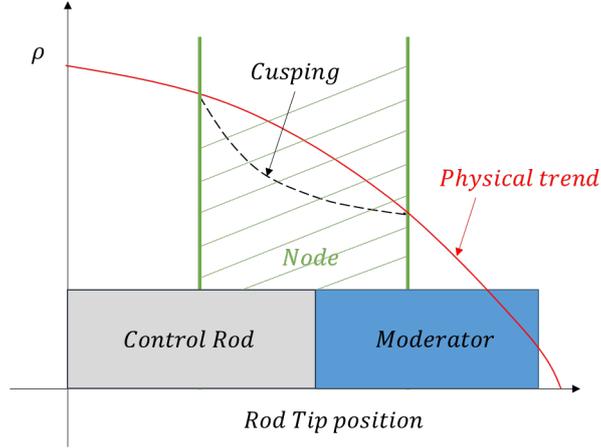


Figure 5.3: Rod cusping effect: reactivity vs control rod tip position.

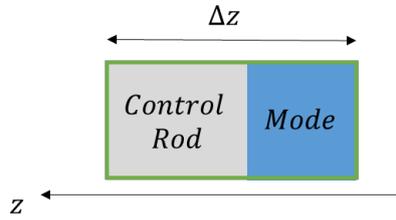


Figure 5.4: Rodded node.

5.5.1 . The Approximate Flux-Weighting Method

Given a partially rodded node k , the average flux in the unrodded part of the node is estimated as

$$\Phi_U^g = \frac{(z_{k-1} - z_{k-2})\Phi_{k-1}^g + (1 - \alpha)(z_k - z_{k-1})\Phi_k^g}{(z_{k-1} - z_{k-2}) + (1 - \alpha)(z_k - z_{k-1})}, \quad (5.16)$$

while the average flux in the rodded part of the node is approximated as

$$\Phi_R^g = \frac{(z_{k+1} - z_k)\Phi_{k+1}^g + \alpha(z_k - z_{k-1})\Phi_k^g}{(z_{k+1} - z_k) + \alpha(z_k - z_{k-1})}, \quad (5.17)$$

where $\alpha \in [0, 1]$ is the fraction of the node occupied by the control rod, z_k is upper axial coordinate of node k and Φ_k^g is the multigroup neutron flux in node k . This said, the average cross section of node k is computed by flux-weighting,

$$\Sigma_x^g = \frac{(1 - \alpha)\Sigma_{x,U}^g\Phi_U^g + \alpha(\Sigma_{x,U}^g + \Delta\Sigma_x^g)\Phi_R^g}{(1 - \alpha)\Phi_U^g + \alpha\Phi_R^g}, \quad (5.18)$$

x being an absorption, $(n, 2n)$ or removal reaction.

5.5.2 . The Equivalence Method

Let us consider two spatial meshes, namely c and f , with the former denoting a coarse axial discretization, where the node containing the control rod tip is treated as a whole, and the latter indicating a finer grid, where the rodded and unrodded parts are split in two separate nodes (Fig.5.4). The present method consists in determining a set of parameters to enforce an equivalence between c and f such that the average, linear and parabolic moments are preserved. For the sake of simplicity, we consider a two-group diffusion model and assume that

- The up-scattering source is negligible;
- Both fast and slow neutrons may induce a fission reaction;
- All fission neutrons are fast.

The multigroup diffusion equations read as

$$\nabla \cdot \mathbf{J}^1 + (\Sigma_a^1 + \Sigma^{2,1})\Phi^1 = \frac{1}{k} \sum_{g=1}^2 (\nu\Sigma)_f^g \Phi^g, \quad (5.19)$$

$$\nabla \cdot \mathbf{J}^2 + \Sigma_a^2 \Phi^2 = \Sigma^{2,1} \Phi^1, \quad (5.20)$$

$$\mathbf{J}^g = -D^g \nabla \Phi^g. \quad (5.21)$$

Let us solve the 1D problem along the axial direction and apply the Nodal Expansion Method, [66],⁵. Hence, Eq. (5.19)-(5.21) are integrated over Δx and Δy ,

$$\frac{dJ^1}{dz} + \left(\Sigma_a^1(z) + \Sigma^{2,1} \right) \Phi^1(z) = \frac{1}{k} \sum_{g=1}^2 (\nu\Sigma)_f^g(z) \Phi^g(z) - L^1(z), \quad (5.22)$$

$$\frac{dJ^2}{dy} + \Sigma_a^2(z) \Phi^2(z) = \Sigma^{2,1}(z) \Phi^1(z) - L^2(z), \quad (5.23)$$

$$J^g(z) = -D^g(z) \frac{d\Phi^g}{dz}, \quad (5.24)$$

where

$$L^g(z) = \frac{1}{\Delta x} \int_0^{\Delta x} J_y^g dx + \frac{1}{\Delta y} \int_0^{\Delta y} J_x^g dy. \quad (5.25)$$

represents the radial leakage. Let us project Eq. (5.22)-(5.23) onto a polynomial basis of the kind

$$\mathbf{P} = \begin{bmatrix} 1 \\ u \\ u^2 - \frac{1}{12} \end{bmatrix}, \quad u = \frac{z}{\Delta z} - \frac{1}{2}, \quad (5.26)$$

u being a reduced coordinate, assuming values in $[-1/2, 1/2]$. This formulation applies to both discretizations, namely c and f . However, for f , i.e., the finer mesh, the control rod tip position matches the node interface, thus the cross sections of the two (homogeneous) nodes may be defined as follows

$$\Sigma_x^g(u) = \begin{cases} \Sigma_{x,U}^g, & \alpha \in [-1/2, \xi] \\ \Sigma_{x,U}^g + \delta\Sigma_x^g, & \alpha \in [\xi, +1/2] \end{cases} \quad (5.27)$$

ξ being the reduced coordinate of the interface between the rodded and unrodded node and $\delta\Sigma_x^g = \Sigma_{x,R}^g - \Sigma_{x,U}^g$ being the difference between the rodded and unrodded cross sections for reaction type x . An analogous formula applies to the diffusion coefficients. Conversely, for c , i.e., the coarse mesh, the control rod tip is contained in the node. The cross section of the equivalent node may be defined as

$$\Sigma_x^g(u) = \Sigma_{x,U}^g + w^g \delta\Sigma_x^g, \quad (5.28)$$

where w^g is a group-dependent parameter. Eq. (5.22) and (5.23) are integrated over $[-1/2, +1/2]$ after substitution of Eq. (5.27), which allows to obtain the zero-order two-node diffusion equations. On the contrary, the zero-order equivalent-node equations are obtained by replacing Eq.

⁵The Nodal Expansion Method is not available in APOLLO3®.

(5.28) in Eq. (5.22)-(5.23) and integration over $[-1/2, +1/2]$. By enforcing an equality between the two two-node and equivalent-node formulations, one can derive w^g . Note that the leakage terms $L^g(z)$ are implicitly supposed to be the same for the two cases.

The equivalence between the high-order moments is enforced in a similar way, namely

$$\int_{-1/2}^{1/2} du P_n(u)[Eq.(g)]_{(5.27)} = \int_{-1/2}^{1/2} du P_n(u)[Eq.(g)]_{(5.28)} - c_n^g, \quad n = 1, 2, \quad (5.29)$$

where $Eq.(g)$ is alternatively the fast or thermal-group diffusion equation, in which, depending on the model, i.e., the two-node or the equivalent-node formulation, Eq. (5.27) or (5.28) are substituted. Note that a group and order-dependent correction c_n^g , with $n = 1, 2$, is added, in each case. In [64], it is shown that both the flux-approximate and equivalent method significantly reduce the control-rod cusping effect, where the latter outbests the results of the former. Nevertheless, none of them is able to eliminate completely the numerical wiggles in the reactivity curve.

5.5.3 . The Sub-plane Method

In [65], two novel subplane-based control rod decusping methods are illustrated for MPACT [52], using 2D/1D Fusion. As stated in Sect. 5.4, the 3D geometry is decomposed into a stack of radial planes, solved by MOC. The 2D planes are coupled along the z-direction, by the transverse leakage term. The solution of the 1D axial equation is attained by a low-order transport model.

In order to speed up the convergence of the transport iterations, MPACT relies on 3D CMFD: each pin-cell is homogenized into a single region, preserving total reaction rates, and the diffusion approximation is applied to obtain the scalar flux and renormalize the fine-mesh transport spectrum. As CMFD is based on pure 3D diffusion, which is not able to capture sharp axial gradients, a 1D NEM-P₃ solver is necessary to solve the spatial and angular dependence of the neutron flux. This allows to compute the axial currents coupling the MOC planes with good precision.

Each MOC plane is axially homogeneous. This clearly questions the applicability of 2D/1D to simulations where the control rods move along the vertical direction, to insert positive or negative reactivity, depending on the direction of motion. One may finely mesh the entire geometry, so that each radial plane is axially homogeneous. Nevertheless, this would significantly increase the computational cost of the whole simulation.

The **subplane-based methods** meet both requirements, i.e., preserving the numerical accuracy without dramatic repercussions on the runtime. Each radial plane is finely subdivided for CMFD and 1D NEM-P₃, but not for 2D MOC, which represents the most expensive part of the solution. In order to fix control rod cusping effects, two approaches are proposed in [65]. The first one uses rodded/unrodded nuclear data in the upper/lower sub-planes, for CMFD cross section homogenization. As a result, the low-order transport solution is able to capture the abrupt variation of the neutron flux around the control rod tip. Then, the radial equation is solved by 2D MOC with flux-weighted cross sections Σ_r , r being the radial mesh index, using the CMFD sub-plane flux,

$$\Sigma_r = \frac{\sum_{k \in plane} \Phi_{rad,r} \Phi_{axi,r}^k \Sigma_r^k \Delta^k}{\sum_{k \in plane} \Phi_{rad,r} \Phi_{axi,r}^k \Delta^k}, \quad (5.30)$$

where k is the sub-plane mesh index, Δ^k the sub-plane thickness, $\Phi_{axi,r}^k$ the cell-averaged sub-plane solution and $\Phi_{rad,r}$ the average radial flux. Note that $\Phi_{rad,r}$ is the same for all sub-planes within the MOC plane. This method allows to reduce the amplitude of the wiggles in the differential reactivity worth curve, but does not eliminate completely the control rod cusping effect.

The second sub-plane method presented in [65] is based on collision probabilities. In practice, the geometry of the partially rodded pin-cell is altered, by transforming the moderator region into a ring, but preserving the volume. The presence of the surrounding pin-cells is taken into account artificially, by introducing an additional ring around the moderator region, whose cross sections are obtained by flux-weighting homogenization. Monodimensional CP-based calculations are performed both with the rodded and the unrodded cross sections, thus providing a radial flux $\Phi_{rad,r}^k$ for each sub-plane k that can be used in Eq. (5.30) for cross section homogenization, in lieu of MOC. This approach prevents the occurrence of control rod cusping effects, but still shows some limitations. In fact,

- The method of collision probabilities is not able to handle anisotropic scattering sources;
- The application of the subplane-based methods is limited to stationary calculations;
- Numerical instabilities may be triggered by very thin cells.

6 - INTRODUCTION TO THE INTEGRO-DIFFERENTIAL TRANSPORT SOLVER (IDT)

6.1 . INTRODUCTION

The present chapter is completely devoted to the APOLLO3® Integro-Differential Transport solver (A3-IDT). The numerical capabilities of IDT are briefly recalled, with particular focus on heterogeneous geometries in two or three dimensions. This will serve as a preliminary basis to the new developments, which are illustrated in the upcoming chapters.

IDT is a neutral-particle transport code, designed to solve the multigroup Boltzmann equation, with discrete ordinates and P_N scattering, in planar, 2D and 3D geometries. It is able to treat an extensive range of boundary conditions, including non-reentering/vacuum, specular and isotropic reflection, π and $\frac{\pi}{2}$ -rotation, diagonal and multigroup albedo, axial symmetry, translation, etc. It is equipped with response-matrix spatial discretizations, including finite differences, nodal expansion and short characteristics methods. For 2D/3D heterogeneous geometries, A3-IDT relies on constant and linear short characteristics, described in Sect. 4.6.4. The spatial domain is subdivided into conformal Cartesian nodes, hosting an arbitrary number of extruded concentric cylinders. The transport operator is inverted by computing angular-dependent probability coefficients in each node and discrete direction of the angular quadrature formula, and propagating the numerical solution to neighbouring cells by piece-wise polynomial currents. A numerical spatial quadrature, based on a local (i.e., cell-by-cell) ray-tracing algorithm, ensures the computation of the collision probabilities for each direction. Finally, a spatial sweeping is conducted for each discrete direction, propagating the angular interface flux moments by front.

In A3-IDT, the fission, up-scattering and self-scattering source are converged by outer, thermal and inner iterations, respectively (Sect. 4.4.2). The use of synthetic operators (e.g., the CMFD method) permits a speed-up of the iterative scheme. In recent years, a non-overlapping domain decomposition method has been implemented, [70], allowing for massive parallel calculations.

The described solver is affected by some weaknesses, mainly attributable to the geometrical model and the memory requirements of the CP matrices (Sect. 4.6.4). First of all, no additional inner mesh refinement is available to boost the numerical accuracy, in case of peaked gradients, generated by strongly absorbing or diffusive media. Moreover, the geometrical representation is limited to conformal structured meshes, with extruded vertical cylinders. This does not allow including hexagonal patterns or horizontal fuel bundles, preventing the use of the solver to compute VVERs or CANDU reactors. In addition, the use of current-coupled collision probabilities (Sect. 4.6.2) poses some limitations to the application of A3-IDT to large multiplying systems or burn-up calculations, which generally include hundreds to thousands of different regions per assembly.

The present chapter is organized as follows. First, some generalities are provided on the main fields of applications of A3-IDT (Sect. 6.2). Then, the implemented numerical methods are briefly summarized (Sect. 6.3), with particular focus on the response-matrix formalism

and spatial discretization of heterogeneous geometries (Sect. 6.3.1-6.3.4). Sect. 6.4 offers a brief overview on the domain decomposition approach. Finally, a balance of pros and cons is drawn, opening to the necessary developments that are illustrated in the upcoming part of the manuscript (Sect. 6.5).

6.2 . HISTORICAL NOTE AND MAIN APPLICATIONS

Born in 1998 as a standalone solver, thanks to the work of [67], IDT has received several contributions over the last 25 years, with [68], [69], [70], [38] setting milestones in the development of the code. IDT is also part of APOLLO2 and APOLLO3®, [71]. At present, it is able to address the direct and adjoint formulation of the neutral-particle Boltzmann equation, for steady-state neutron and photon transport, with k -eigenvalue or fixed-source formalism.

IDT may target a variety of industrial applications, ranging from lattice calculations, homogenized pin-by-pin core simulations, 2D photon transport calculations, 1D-planar simulations with multi-group albedo computation and 2D/3D direct transport simulations, taking advantage of domain decomposition and OpenMP/MPI hybrid parallelism. It is also the subject of intense research activity, with recent applications to self-shielding problems with localized sources, [72], and to the generation of the importance function distribution, that is provided as an input to the TRIPOLI-4® Monte Carlo code, [73].

6.3 . SPATIAL DISCRETIZATION

IDT supports a relatively wide variety of spatial discretization schemes, including one-dimensional slabs as well as Cartesian geometries, in two or three dimensions, with the possibility to input an arbitrary number of concentric heterogeneous cylinders within each cell (Fig. 6.1-6.2). The spatial domain is partitioned into a conformal mesh of $N_x \cdot N_y \cdot N_z$ Cartesian nodes, where N_x , N_y and N_z represent the number of subdivisions for the x , y and z -axis, respectively. The spatial solution of the Boltzmann equation within each cell can be attained by several methods, in particular:

- Diamond difference and nodal expansion method in homogeneous Cartesian cells;
- Constant, linear, parabolic and cubic characteristics in 1D planar geometries;
- Constant, linear and bilinear short characteristics in homogeneous Cartesian cells;
- Constant and linear short characteristics in heterogeneous Cartesian cells.

IDT is also equipped with a number of acceleration techniques, based on the Chebyshev, BPA (Boundary Projection Acceleration), CMFD and Thermal-Rebalancing synthetic operators, [45], [46], [74].

6.3.1 . Response Matrix Approach

In IDT, all methods providing the spatial solution of the Boltzmann equation over a cell are based on a response matrix formalism, which consists of two equations of the kind

$$\psi(\Omega) = \underline{\mathbf{C}}(\Omega)\mathbf{q}(\Omega) + \sum_{s' \in \partial V^-} \underline{\mathbf{I}}_{s'}(\Omega)\psi_{s'}^-(\Omega), \quad (6.1)$$

$$\psi_s^+(\Omega) = \underline{\mathbf{E}}_s(\Omega)\mathbf{q}(\Omega) + \sum_{s' \in \partial V^-} \underline{\mathbf{T}}_{s'}(\Omega)\psi_{s'}^-(\Omega), \quad s \in \partial V^+, \quad (6.2)$$

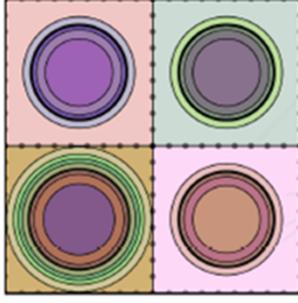


Figure 6.1: 2D heterogeneous cells.

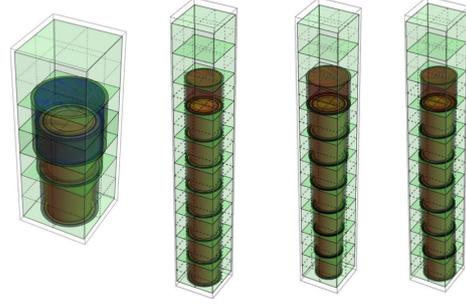


Figure 6.2: Examples of deformed 3D pin-cells.

where

- $\underline{\mathbf{C}}$, $\underline{\mathbf{I}}$, $\underline{\mathbf{E}}$ and $\underline{\mathbf{T}}$ represent the collision, incoming, escape and transmission angular probability matrices, respectively;
- ψ , ψ_s^\pm and \mathbf{q} denote the spatial moments of the flux, of the outgoing/incoming flux and of the neutron source, respectively;
- s and s' indicate the surface discretization on the outgoing and incoming boundaries ∂V^+ and ∂V^- , respectively, which depend on the angular direction illuminating the cell, Ω .

In other words, the response matrix formalism provides the solution to the neutral-particle transport equation over a mesh of spatial nodes coupled by particle currents, where Eq. (6.1) may be interpreted as a balance preserving the number of neutrons or photons, while Eq. (6.2) is based on the continuity of the cell-boundary currents. Both equations are obtained by projecting the source and interface fluxes onto polynomial bases, as discussed in Sect. 4.6.4.

6.3.2 . Heterogeneous Cartesian Cells

Let us narrow the focus on heterogeneous geometries. The spatial domain is decomposed into a modular Cartesian pattern, where each node may contain an arbitrary number of heterogeneous cylinders. This chunk of the original geometry goes under the name of Heterogeneous Cartesian Cell (HCC). A conforming surface mesh may be superimposed on the cell sides, in order to improve the representation of the interface angular flux. Note that all cylinders share the same axis, which is parallel to the vertical direction and contains the centre of mass of the Cartesian cell, as shown in Fig. 6.3. Due to the presence of the cylinders, the angular probability matrices of Eq. (6.1) and (6.2) cannot be integrated analytically. A ray-tracing algorithm permits computing the surface and volume integrals defining $\underline{\mathbf{C}}$, $\underline{\mathbf{I}}$, $\underline{\mathbf{E}}$ and $\underline{\mathbf{T}}$, by means of a quadrature formula. The weights are obtained by projection onto a transverse plane, which is perpendicular to the set of rays t_0, t_1, \dots, t_n , with $t_i \parallel \Omega$ (Fig. 6.5).

6.3.3 . Spatial Sweeping

Let us fix the angular direction, Ω . The spatial solution over the ensemble of cells composing the system geometry is obtained by applying Eq. (6.1) and (6.2) in each node. The discretized domain is swept along direction Ω by ordering the spatial cells, which are solved one after the other, following the interface currents. For heterogeneous cells, the sweeping proceeds by front, as indicated in Fig. 6.4. Once solved a HCC, its outgoing flux moments are transmitted to the down-stream HCC, by taking advantage of the continuity condition,

$$\psi_s^+|_{up} = \psi_{s'}^-|_{down}, \quad (6.3)$$

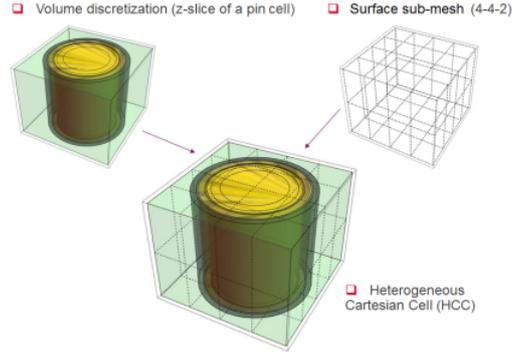


Figure 6.3: Heterogeneous Cartesian Cell (HCC).

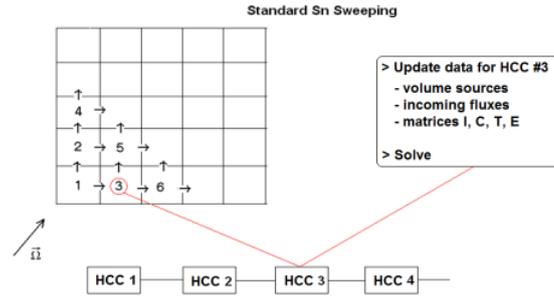


Figure 6.4: Sweeping by front.

where $\psi_s^+|_{up}$ and $\psi_{s'}^-|_{down}$ represent the outgoing and incoming angular flux moments, for the up-stream and down-stream cells, respectively. Note that the surface flux moments are transmitted with no approximation, thanks to the conforming surface mesh discretization. Finally, one should notice that at the system boundary the surface flux is provided by the boundary conditions.

6.3.4 . Ray-tracing

The angular probability matrices are computed by means of a modular tracking, where each ray-tracing module corresponds to a cell-geometry type. Generally, for industrial applications, the whole system may be reconstructed by repeating few geometry types, thus limiting the memory imprint of the tracking.

Assuming α, β, \dots as the indexes of the regions within the HCC and $T_\alpha(\Omega_d)$ as the set of trajectories intersecting region α along direction Ω_d , the ray-tracing module performs numerical transverse integration as

$$\int_{D_\alpha} d^3r f(\mathbf{r}) \simeq \sum_{t \in T_\alpha(\Omega_d)} w_t^\perp \int_{\xi_\alpha^-}^{\xi_\alpha^+} d\xi' f(\mathbf{r}_t^- + \xi' \Omega_d) \quad (6.4)$$

with

- ξ being the coordinate along the trajectory and f an arbitrary spatial function;
- $w_t^\perp = |\Omega_d \cdot \mathbf{n}_s| \Delta A_{s,t} = |\Omega_d \cdot \mathbf{n}_{s'}| \Delta A_{s',t}$ defining the weight of trajectory t , \mathbf{n}_s and $\Delta A_{s,t}$ being the unit-normal vector and the projection of surface s on the transverse plane, respectively;

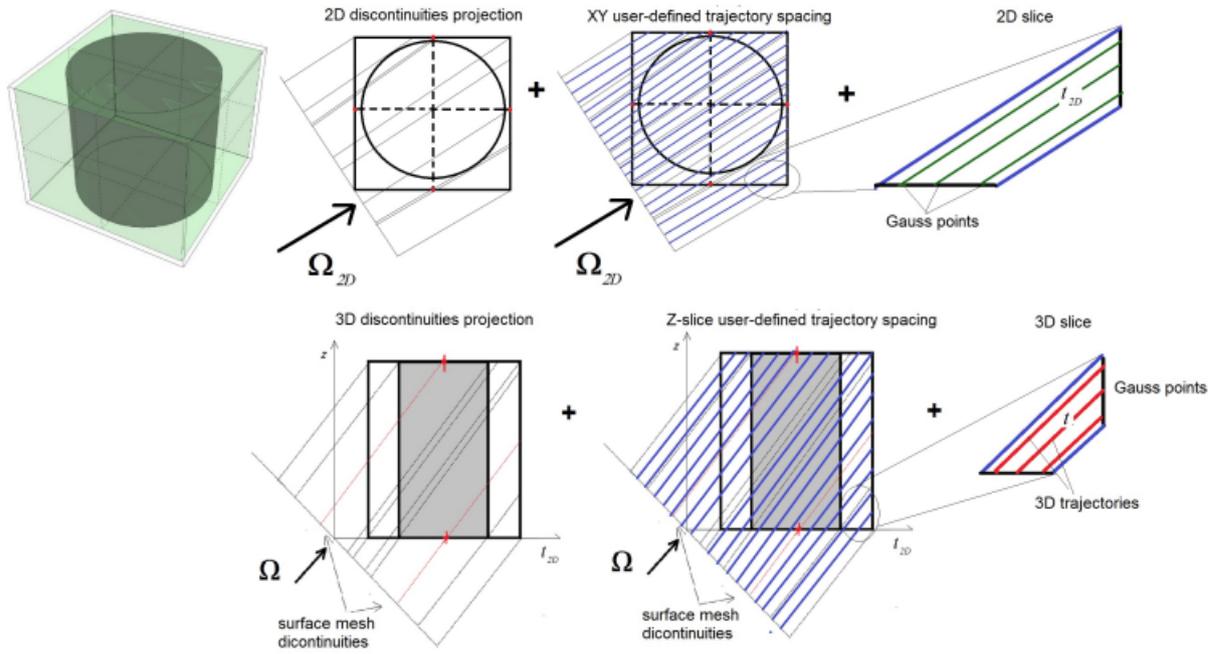


Figure 6.5: Ray-tracing.

- ξ_{α}^{\pm} denoting the coordinates of the trajectory intersections with the region boundary.

The local coordinate system, which is centered in the south-west corner of the HCC, is rotated along direction Ω_d . Three-dimensional ray-tracing is performed by taking advantage of the extruded nature of the geometry. Moreover, differently from MOC, as the rays are local to the HCCs, the geometrical discontinuities can be projected onto the transverse plane.

For 3D cells, the local tracking is performed by factorization in two planar ray-tracing modules (Fig. 6.5):

- First, a 2D-XY tracking is performed, where the discontinuities on the XY plane are projected onto the transverse plane. Then, the XY geometry is illuminated by a set of equally-spaced trajectories, with spacing Δ_{XY} defined by the user, parallel to the projection Ω_d^{XY} of Ω_d on the XY-plane. Finally, on each 2D slice between two consecutive trajectories, N_{XY} Gauss points are introduced, where, once again, N_{XY} is fixed in the input file;
- The prior algorithm is repeated for each tZ -slice, where t is a 2D-trajectory. Again, two user-defined parameters are required, i.e., a uniform spacing Δ_{tZ} and a number of Gauss points N_{tZ} .

The 3D weight w_{3D} of a 3D trajectory can be factorized as $w_{3D} = w_{XY} \cdot w_{tZ}$.

For each direction of the angular quadrature formula, the ensemble of trajectory data are stored in memory. These comprise:

- The 2D-angle order-number;
- The incoming surface s' ;

- The outgoing surface s ;
- The chord-to-region map;
- The trajectory weight w_{3D} ;
- The opposite of the cosine of the angle between the selected direction and the outward unit-vector of the incoming surface, $-\mathbf{\Omega} \cdot \mathbf{n}_s$; ⁶
- The cosine of the angle between the selected direction and the outward unit-vector of the outgoing surface, $\mathbf{\Omega} \cdot \mathbf{n}_s$; ⁶
- The local coordinates of the trajectory hang-point \mathbf{r}_t^- on the incoming cell-side; ⁶
- The local coordinates of the trajectory hang-point \mathbf{r}_t^+ on the outgoing cell-side; ⁶
- The chord-lengths.

6.4 . DOMAIN DECOMPOSITION IN IDT

6.4.1 . Problem Position

IDT relies on domain decomposition and massive parallelism to reduce the CPU time, [70], [76], [75]. Non-overlapping DDM has been implemented, in order to minimize data exchange. Provided D is the spatial domain of the whole system, DDM consists in decomposing D into a number of subdomains equal to U , i.e., $D = \bigcup_{u=1, \dots, U} D_u$. Let us introduce the following notation for the phase-space:

$$X_u \equiv (\mathbf{r} \in D_u, \mathbf{\Omega} \in S^2, E \in \mathbb{R}_G^+), \quad (6.5)$$

$$\partial X_u^\pm \equiv (\mathbf{r} \in \Gamma_u^\pm(\mathbf{\Omega}), \mathbf{\Omega} \in S^2, E \in \mathbb{R}_G^+), \quad (6.6)$$

$$\Gamma_u^\pm(\mathbf{\Omega}) \equiv (\mathbf{r} \in \partial D_u, \mathbf{n}_{u+}(\mathbf{r}) \cdot \mathbf{\Omega} \gtrless 0). \quad (6.7)$$

The solution of the whole problem may be reconstructed as

$$\psi(x) = \sum_u \chi_u(x) \psi_u(x), \quad x \in X, \quad (6.8)$$

where X denotes the phase-space domain of the whole system, while χ_u is the characteristic function of sub-domain u and reads as

$$\chi_u(x) = \begin{cases} 1, & \text{if } x \in X_u, \\ 0, & \text{otherwise.} \end{cases} \quad (6.9)$$

Eq. (6.8) is verified, provided the interface currents between neighboring sub-domains satisfy the continuity condition,

$$\psi_u^-(x) = \psi_v^+(x), \quad x \in \partial X_u^- \cap \partial X_v^+, \quad \forall u, v : u \cap v \neq \emptyset. \quad (6.10)$$

The transport problem is reformulated as U multi-group source problems of smaller size, namely

$$\begin{cases} (\mathbf{L} - \mathbf{H})_u \psi_u(x) = q_u(x), & x \in X_u, \\ \psi_u^-(x) = \psi_v^+(x), & x \in \partial X_u^- \cap \partial X_v^+, \quad \forall u \cap v \neq \emptyset, \\ \psi_u^-(x) = 0, & x \in \partial X_u^- \cap X^-, \end{cases} \quad (6.11)$$

⁶Data stored for linear (or higher) order development.

where the fission source distribution q_u for sub-domain u reads as

$$q_u(x) = \frac{\mathbf{F}_u \psi_u(x)}{k}. \quad (6.12)$$

6.4.2 . Power Iterations

Power iterations are performed to converge the fission source. First, the eigenvalue, the interface flux and the fission source are initialized for each sub-domain. Each power iteration solves U boundary-source problems of the kind

$$\left\{ \begin{array}{l} (\mathbf{L} - \mathbf{H})_u \psi_u^{(e+1)}(x) = q_u^{(e)}(x), \quad x \in X_u, \\ \psi_u^{-,(e+1)}(x) = \psi_v^{+,(e)}(x), \quad x \in \partial X_u^- \cap \partial X_v^+, \quad \forall u \cap v \neq \emptyset, \\ \psi_u^{-,(e+1)}(x) = 0, \quad x \in \partial X_u^- \cap X^-. \end{array} \right. \quad (6.13)$$

The local net disappearance operator $(\mathbf{L} - \mathbf{H})_u$ is inverted by thermal iterations, which are performed in each sub-domain, to converge the scattering source. Note that the sub-domains transmit the interface solution only at the end of the local thermal/inner iterations (Eq. (4.43)-(4.47)), in order to minimize the exchange of information. Once solved the local sub-domain problems, the k -eigenvalue and local fission sources are updated as follows:

$$k^{(e+1)} = k^{(e)} \frac{\sum_{u=1, \dots, U} (w, \mathbf{F}_u \psi_u^{(e+1)})_u}{\sum_{u=1, \dots, U} (w, \mathbf{F}_u \psi_u^{(e)})_u}, \quad (6.14)$$

$$q_u^{(e+1)}(x) = \frac{\mathbf{F}_u \psi_u^{(e+1)}(x)}{k^{(e+1)}}.$$

Successive iterations are run, until the convergence criteria on the eigenvalue, fission integral and incoming flux are simultaneously verified. More specifically, the errors are defined as follows:⁷

$$\varepsilon_k^{rel} = \left| 1 - \frac{k^{(e+1)}}{k^{(e)}} \right|, \quad (6.15)$$

$$\varepsilon_F^{L^\infty} = \left| 1 - \frac{\sum_g (F_u \psi_u^{(e+1)})_g}{\sum_g (F_u \psi_u^{(e)})_g} \right|, \quad u = 1, U \quad \mathbf{r} \in D_{u,R}, \quad (6.16)$$

$$\varepsilon_\psi^{L^\infty}(x) = \left| 1 - \frac{\int_{2\pi^-} d^2\Omega \, |\mathbf{n} \cdot \boldsymbol{\Omega}| \psi_u^{-,(e+1)}(x)}{\int_{2\pi^-} d^2\Omega \, |\mathbf{n} \cdot \boldsymbol{\Omega}| \psi_u^{-,(e)}(x)} \right|, \quad u = 1, U \quad x \in \partial X_u^-. \quad (6.17)$$

For each outer iteration, a multi-group low-order-transport problem is solved. In particular, IDT relies on Coarse Mesh Finite Differences (CMFD) (Sect. 4.7.2), to speed up the convergence of the transport iterations, where the coarse spatial/energy mesh is automatically generated, by simple inputs. Note that the CMFD spatial mesh is obtained by coarsening the HCC grid in each sub-domain. The mesh generator is able to preserve the global conformity of the coarse mesh.

⁷Error-norms used in IDT: L^∞ -norm on the flux and on the angular and spatial moments; L^∞ -norm on the border, with angular-integrated partial currents; L^2 -norm on the thermal source; L^∞ -norm on the fission integral.

Once determined the new transport iterate $\psi_u^{(e+1/2)}$ (Eq. (4.106)), $\forall u \in p$, with p being an MPI process, the CMFD eigenvalue and coarse-mesh scalar flux are initialized to

$$k^{CMFD,(0)} = k^{(e)}, \quad \Phi^{CMFD,(0)} = \int dx \psi_u^{(e+1/2)}. \quad (6.18)$$

The synthetic problem is approached by successive iterations, of index j , where, at each step, one solves the ensemble of subdomains $u \in p$, (Eq. (4.117)). A new iterate is computed, with eigenvalue

$$k^{CMFD,(j+1)} = k^{CMFD,(j)} \frac{\sum_{u=1,\dots,U} (w, \mathbf{F}_u \Phi_u^{CMFD,(j+1)})_u}{\sum_{u=1,\dots,U} (w, \mathbf{F}_u \Phi_u^{CMFD,(j)})_u}. \quad (6.19)$$

Once converged the low-order-transport problem, the fine transport spectrum is renormalized by means of the obtained coarse-mesh solution (Eq. (4.114)) and the effective multiplication factor $k^{(e+1)}$ is set equal to the converged CMFD eigenvalue.

6.4.3 . Pivot-Grid, CMFD-Mesh and Subdomain-to-Process-Map Automatic Generation

The fundamental algorithm responsible for the automatic construction of

- the pivot grid,
- the CMFD mesh,
- the Subdomain-to-Process mapping,

may be summarized as follows. Suppose N_k and M_k are respectively the number of fine steps and required coarse steps along axis k , with $k = x, y, z$ and $M_k \leq N_k$. Provided m_k is a coarse step along axis k , the number of fine steps contained in m_k is given by

$$n_{m_k} = \max \left(1, \left\lfloor \frac{N_k}{M_k} \right\rfloor \right) + 1, \quad (6.20)$$

for $m_k = 1, \dots, N_k(\text{ mod } M_k)$, and

$$n_{m_k} = \max \left(1, \left\lfloor \frac{N_k}{M_k} \right\rfloor \right), \quad (6.21)$$

for $m_k = N_k(\text{ mod } M_k) + 1, \dots, M_k$. In other words, the first $N_k(\text{ mod } M_k)$ coarse axial steps contain $\max \left(1, \left\lfloor \frac{N_k}{M_k} \right\rfloor \right) + 1$ consecutive fine axial steps, along axis k , whereas the remaining ones contain only $\max \left(1, \left\lfloor \frac{N_k}{M_k} \right\rfloor \right)$ consecutive fine axial steps.

This simple algorithm is used to define

- (a) the number of HCCs per subdomain. For subdomain (m_x, m_y, m_z) , the number of fine axial steps is defined by $n_{m_x} \cdot n_{m_y} \cdot n_{m_z}$, where $n_{m_x}, n_{m_y}, n_{m_z}$ are computed by Eq. (6.20) or (6.21);
- (b) the number of HCCs per CMFD node, in a given subdomain (as in Item (a)).

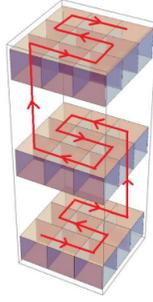


Figure 6.6: Serpentine path, for a 3x3x3 pivot grid.

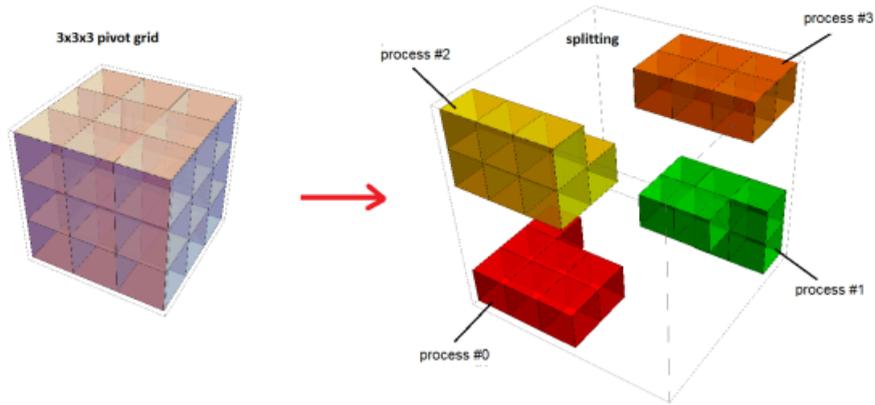


Figure 6.7: Sub-domain distribution on 4 MPI processes, for a 3x3x3 pivot grid.

Analogously, provided P is the required number of MPI processes, the number of sub-domains for process p is defined as

$$U_p = \max \left(1, \left\lfloor \frac{U}{P} \right\rfloor \right) + 1, \quad (6.22)$$

for $p = 1, \dots, U \pmod{P}$, and

$$U_p = \max \left(1, \left\lfloor \frac{U}{P} \right\rfloor \right), \quad (6.23)$$

for $p = U \pmod{P} + 1, \dots, P$. The spatial distribution of the sub-domains is depicted in Fig. 6.6 and Fig. 6.7, for a 3x3x3 pivot grid, distributed on 4 MPI processes. In practice, the pivot grid is swept following a serpentine path, which allows having sub-domains that share at least one side interface in each process.

Once decomposed the geometry, IDT is able to identify and gather the sub-domains sharing the same spatial mesh and material distribution into Generating Calculation Units (GCUs). In addition, each GCU has a unique set of solver options, such as the

- angular quadrature formula,
- inner/outer acceleration options,
- spatial homogenization and energy condensation inputs,

allowing for an optimal data distribution. The interested reader is referred to [70] for more details.

6.5 . DISTINCTIVE FEATURES, APPLICATION AND LIMITS OF APOLLO3® IDT

In this chapter, a general overview has been provided on the Integro-Differential Transport Solver of APOLLO3®. Before moving forward, it is worth mentioning that

- Unlike other solvers based on MOSC, the angular flux is not approximated on the inner surfaces of the cell, but only on the cell-boundary grid, enhancing the numerical accuracy of the solution;
- The sub-surface mesh is conformal, in order to transmit the interface flux moments with no approximation;
- For each angular direction and energy group, a set of volume and interface flux moments has to be computed, whose number depends on MOSC expansion order and geometry dimensionality as well as on the number of cell regions (cylinders + moderator) and sub-surfaces on each cell side. Unlike MOC (Sect. 4.6.3), the unknowns are computed by angular and energy-dependent angular coefficients, which, at constant order, may be interpreted as probabilities. Due to the presence of cylinders, their calculation requires a numerical quadrature, which is provided by ray-tracing. Differently from MOC, MOSC can benefit from local tracking, which can be calibrated cell by cell. This, together with discontinuity projection onto the transverse plane, allows to optimize the distribution and number of trajectories within the cell, as well as to compute volumes with higher numerical accuracy;
- APOLLO3® IDT applies domain decomposition, to run massive parallel simulations. In this frame, the solver can be and has already been employed for 3D direct transport calculations. It is worth mentioning the application to a 3D full PWR core, [44], with a 26-group P_3 -cross section library, using a S8 Level-Symmetric angular quadrature formula and Linear Short Characteristics. With 336 threads (12 threads x 28 nodes), IDT managed to compute the whole reactor core, with a total of $\sim 1.67 \cdot 10^{10}$ degrees of freedom, in 7.8 h.

Despite the mentioned pros, APOLLO3® IDT also suffers from some major issues, limiting its range of applications or numerical accuracy. Let us provide a detailed list of the most relevant problems and related consequences.

Geometrical modeling. A3-IDT is only able to model a conformal lattice of centered pin-cells hosting vertical heterogeneous cylinders. As a result, A3-IDT cannot:

- Introduce unstructured and/or non-conforming meshes;
- Model non-centered cylinders;
- Represent non-extruded cells;
- Account for hexagonal arrays or horizontal fuel-pin cells, preventing its application to VVERs or CANDU reactors;
- Refine the spatial mesh by sectors or Cartesian grids;
- Host non-cylindrical fuel pellets within the pin-cell (e.g., pebbles).

Control-rod insertion/extraction. As mentioned, A3-IDT as well as most A3 solvers cannot model non-extruded or non-conforming geometries. Generally, the axial size of each HCC ranges between 5 and 10 *cm*. On the other hand, control rods are inserted or withdrawn with a step length of the order of 1 – 2 *cm*. As 3D pin-cells are extruded components, one may either introduce very fine HCCs, with $\Delta h \sim 1$ *cm*, to exactly represent the control rod movement, or homogenize the control-rod tip with the moderator below. The first solution pathway is very expensive and is generally discarded in practical applications. On the other hand, if the control-rod tip is homogenized with the underlying moderator, a non-physical behavior is triggered, resulting in a discontinuity of the first derivative of the curve representing the reactivity worth (Sect. 5.5).

Numerical accuracy. Each HCC may comprise an arbitrary number of concentric cylinders, but cannot be equipped with azimuthal sectors or Cartesian grids. As the source and interface flux expansion is limited to linear order, this may entail a significant loss in the numerical accuracy of the code. This is generally observed for small multiplying systems hosting water holes or strong absorbers (e.g., Gadolinium, AIC, B4C rods) where large gradients may require a finer discretization or higher development order.

Memory burden. MOSC and, more generally, methods based on current-coupled collision probabilities suffer from large memory pressure. In fact, the size of the angular probability matrices increases linearly with the number of HCCs and quadratically with the number of regions per cell. This may raise significant problems for relatively large systems (e.g., a reactor core) or even smaller geometries comprising a high number of regions. The latter may be the case of lattice simulations that typically require fuel-depletion analysis. In fact, due nuclear transmutations, a fuel-assembly will contain thousands of regions with different total cross sections, thus severely increasing the memory requirements of CCCPs.

The mentioned problems may, in a way or another, limit or, in some cases, even prevent the use of A3-IDT. The work presented in the upcoming chapters aims at extending the range of applications of IDT, enhancing its numerical accuracy and easing the memory pressure, by dedicated algorithms, in order to prepare a robust solver for lattice and core simulations. In particular, the last chapter is centered on recent developments enabling coupled neutronics/thermal-hydraulics calculations, employing, on the one side, IDT for computing the reactor power distribution and, on the other, THEDI, [77], for determining the thermal feedback. For the sake of completeness, a brief presentation of THEDI is provided in App. A.

Part II

NEW MODELING CAPABILITIES IN IDT

7 - GEOMETRICAL NOVELTIES

7.1 . INTRODUCTION

IDT is a discrete-ordinates transport solver, based on angular-dependent collision probabilities and P_N expansion of the scattering source. The code decomposes the spatial domain in modular XYZ patterns, containing a chunk of the original system, [43], [69]. In recent years, IDT has been developed to model fuel pin-cells in real geometry, by introducing the possibility to input a set of concentric heterogeneous rings within each Cartesian node, [44], [80]. This geometrical unit represents the so-called Heterogeneous Cartesian Cell (HCC).

As mentioned in Chapter 6, the spatial solution of the transport equation is computed by Linear Short Characteristics (LSC). Unlike other solvers based on the same method, [39], [81], [82], the integral transport equation is projected over each HCC. This leads to a system of algebraic equations, where the unknowns are the spatial moments of the angular flux within each heterogeneous region and on each surface of the interface-boundary mesh. Note that the angular flux is not approximated on the inner surfaces of the cell. The solution is propagated transmitting the interface angular fluxes to the down-stream neighbouring cells, by HCC-front spatial sweeping, as in standard XY(Z)- S_N .

The present model only allows the application of IDT to conformal XYZ pin-cell positions, as the PWR lattice. Moreover, the lack of spatial refinement along the azimuthal angle limits the numerical accuracy of the code, especially for lattice configurations containing strong absorbers or water holes.

This chapter illustrates the geometrical novelties introduced in IDT. In particular, Sect. 7.2 provides a brief overview of the main geometric capabilities. Sect. 7.3 presents the fundamental algorithm allowing for the generation of unstructured, non-conformal and non-extruded geometries. Afterwards, a combinatorial ray-tracing technique is described (Sect. 7.4), together with a set of geometric patterns, aimed at verifying numerical quadrature (Sect. 7.4.1). In Sect. 7.5, focus is given to the mathematical model permitting HCC alignment to the x and y -axis. Finally, the new geometrical model is applied to solve a set of problems, in two and three dimensions, and the numerical precision is established by comparison with a Monte Carlo reference (Sect. 7.6-7.7).

7.2 . NEW HCC MODULAR GEOMETRY - A GENERAL OVERVIEW

Provided the new solver preserves the conformal nature of the macro-grid lodging the HCCs, one can now refine the local mesh of each Cartesian node. The new geometrical capabilities of IDT are able to combine cylinders and X/Y/Z planes within each HCC (Fig. 7.1). Neighbouring HCCs may thus be discretized in a different way. More in general, the latest developments allow to:

- Superimpose local non-conformal Cartesian grids onto the set of concentric cylinders (Fig. 7.2);
- Represent off-centered rings (Fig. 7.3);

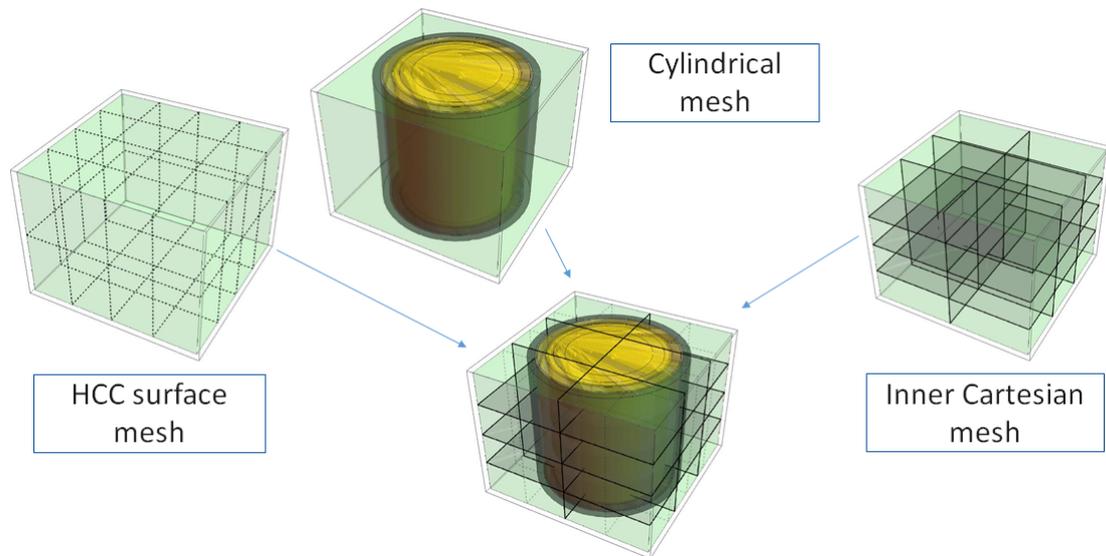


Figure 7.1: New Heterogeneous Cartesian Cell.

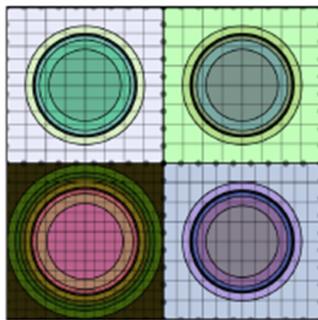


Figure 7.2: Non-conforming inner grids.

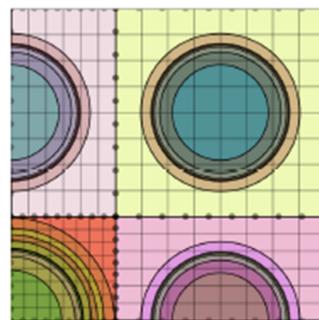


Figure 7.3: Quarter and half pin.

- Simulate non-Cartesian components, e.g., structural elements (reactor vessel, barrel, spacer grids, etc.) or hexagonal patterns (VVER reactors) (Fig. 7.4-7.5);
- Model unstructured and non-extruded geometries (Fig. 7.9). In this respect, note that any unstructured geometry composed by a set of cylinders and X/Y/Z planes can be decomposed in HCC patterns.

As shown in Fig. 7.2 and Fig. 7.3, the sub-surface mesh conformity is maintained, in order to propagate the piece-wise polynomial currents with no further approximation (black dots in Fig. 7.2-7.4). The interface border is generally discretized into equal-sized axial steps, namely $\Delta\xi_1^{sub} = \Delta\xi_2^{sub} = \dots = \Delta\xi_N^{sub}$, where $\xi = x, y, z$ and *sub* denotes the surface mesh. Nevertheless, a dedicated tool permits introducing surface steps of different length along the same axis. Note that the inner-volume axial steps may or may not coincide with the ones defining the interface mesh (e.g., Fig. 7.2).

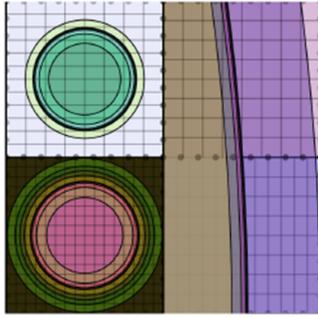


Figure 7.4: Chunk of structural component.

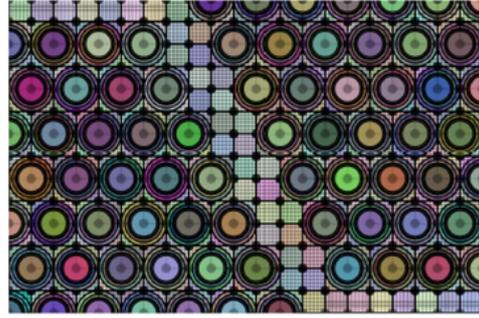


Figure 7.5: Hexagonal lattice.

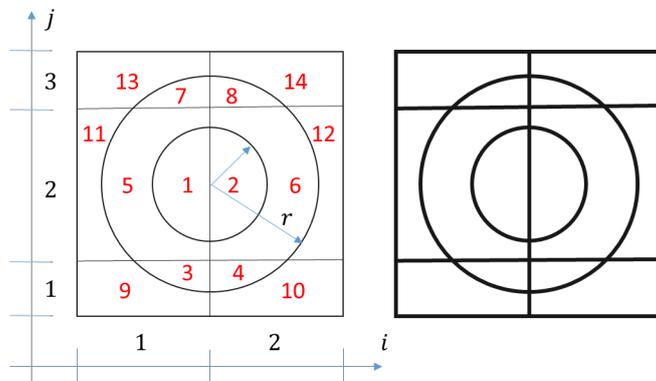


Figure 7.6: Automatic mesh generator nb. 1. Each ray-tracing region is mapped onto a different computational region.

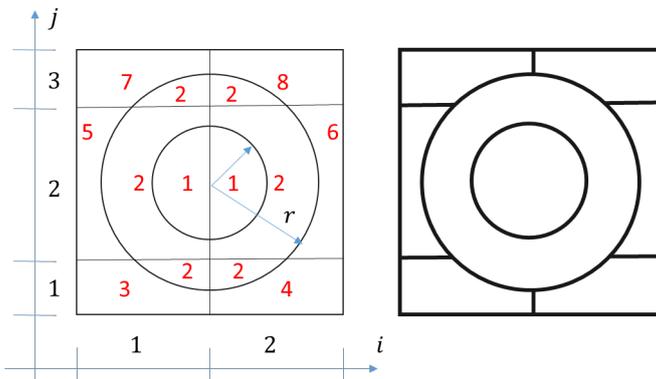


Figure 7.7: Automatic mesh generator nb. 2. The ray-tracing regions within the same circle are merged into a single computational region. The spatial refinement is maintained in the surrounding moderator ($r = 0$).

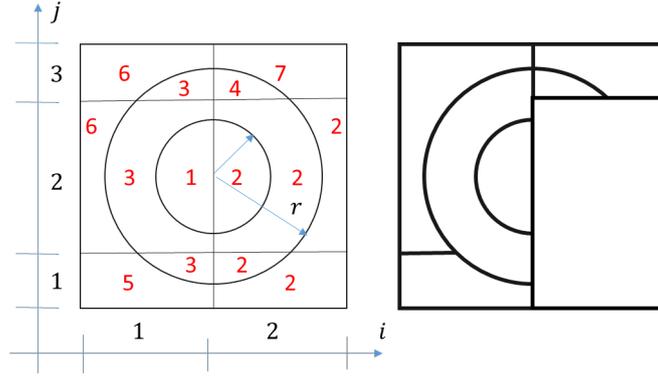


Figure 7.8: Non-automatic mesh generator. The number of computational regions has been reduced to $N_\alpha = 7$, by assigning the same α to different ray-tracing regions.

7.3 . TUPLE-TO-REGION MAP

As depicted in Fig. 7.6-7.8, the inner-volume mesh can be coarsened or customized, to meet specific user's requirements. In particular, one may reduce the mesh refinement, by collapsing some two or more ray-tracing regions into a single computational region. The mathematical tool which is responsible for the association between ray-tracing and computational regions is the Tuple-to-Region Map (TRM). To understand, let us consider a two-dimensional centered cell, containing 2 concentric rings ($N_r = 2$) and 2×3 Cartesian planes ($N_i = 2, N_j = 3$) (Fig. 7.6-7.8). In the following,

- $i \in \mathbb{N}_i \equiv \{1, \dots, N_i\}$ is the index identifying the steps along the x-axis, $j \in \mathbb{N}_j \equiv \{1, \dots, N_j\}$ is the index identifying the steps along the y-axis and $r \in \mathbb{N}_r \equiv \{0, \dots, N_r\}$ is the index identifying different rings, with the convention that the outermost area (normally occupied by the moderator) is indicated with $r = 0$. In 3D geometries, an additional index, namely $k \in \mathbb{N}_k \equiv \{1, \dots, N_k\}$, allows representing the steps along the z axis;
- $\alpha \in \mathbb{N}_\alpha \equiv \{1, \dots, N_\alpha\}$ is a tag for a computational region contained in the HCC.

(i, j, r, k) is a tuple of integer numbers representing a ray-tracing region. The Tuple-to-Region Map (TRM) is a function of the kind

$$f : \mathbb{N}_i \times \mathbb{N}_j \times \mathbb{N}_r \times \mathbb{N}_k \rightarrow \mathbb{N}_\alpha, \quad (7.1)$$

which handles the association between ray-tracing regions, defined as (i, j, r, k) , and computational regions, indicated by a single integer number (α), specified by the user or by an external mesh generator. In general, for geometries composed by planes and cylinders, only a subset of indexes within $\mathbb{N}_i \times \mathbb{N}_j \times \mathbb{N}_r \times \mathbb{N}_k$ is employed for region numbering. A simple example is provided by Fig. 7.6, where, for instance, no computational region is associated to the tuple $(1, 1, 1)$. Hence, we set

$$f(i, j, r, k) = \begin{cases} \alpha \in \mathbb{N}_\alpha, \\ 0, \text{ for any undefined region.} \end{cases} \quad (7.2)$$

By this tool, one can map each tuple defining a ray-tracing region onto a single integer, but the function is not necessarily one-to-one. The user may either choose two automatic maps

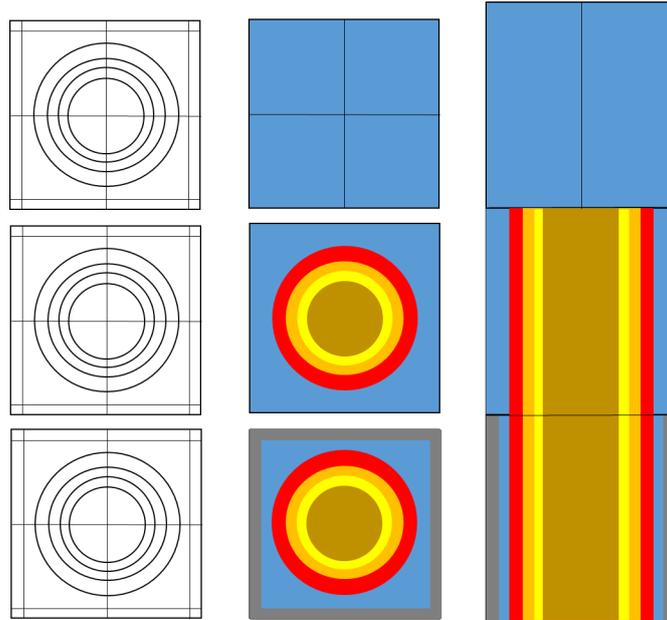


Figure 7.9: Non-extruded cell: on the left-hand-side, the plane-by-plane ray-tracing geometry; at the centre, the plane-by-plane computational geometry; on the right-hand-side, the transverse section.

(Fig. 7.6-7.7) or customize their own map (Fig. 7.8), by specific inputs. In Fig. 7.6-7.8, the computational regions are labeled with a red tag. The outcome of the mesh generator is provided by the picture on the right-hand side, showing the particular discretization chosen by the user. The first automatic TRM allows superimposing the Cartesian mesh onto the radial one, thus obtaining a computational grid of the type illustrated in Fig. 7.6. A second automatic option allows removing the XY grid meshing the pin, but leaves the surrounding moderator discretized, as depicted in Fig. 7.7. Finally, a third possibility allows an external mesh generator to input a customized map to coarsen the discretization (Fig. 7.8). Note that the second automatic TRM as well as the non-automatic option permit reducing the spatial refinement, and, as a consequence, the computational effort and memory imprint of the angular probability matrices.

Besides reducing the computational and memory requirements, the Tuple-to-Region Map enables the user to design non-extruded pin-cell geometries. An example is provided by Fig. 7.9, where the spatial discretization is customized plane by plane. The cell consists of three axial layers, comprising the same number of ray-tracing regions, but a variable number of computational regions. The Tuple-to-Region Map is not indicated, due to the lack of space. One should also notice that the possibility to model non-extruded geometries allows for a high-fidelity representation of the control rods, for which an application is illustrated in Chap. 8.

7.4 . COMBINATORIAL RAY-TRACING

The new version of IDT is equipped with a combinatorial ray-tracing module. Analogously to Sect. 6.3.4, numerical transverse integration is performed by Eq. (6.4). The local coordinate system, which is centered in the south/west corner of the HCC box (Fig. 7.10), is rotated along direction $\Omega_d \in S_N$. Three-dimensional ray-tracing is performed taking advantage of the locally-extruded nature of the cell. The chord-lengths are computed in two steps:

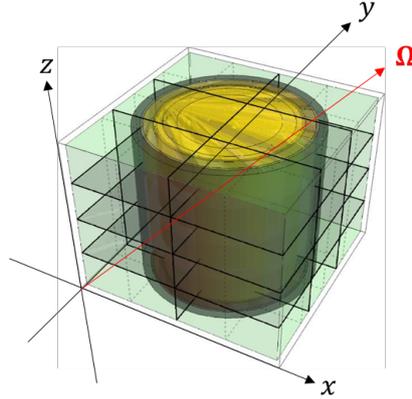


Figure 7.10: Original frame of reference (before rotation).

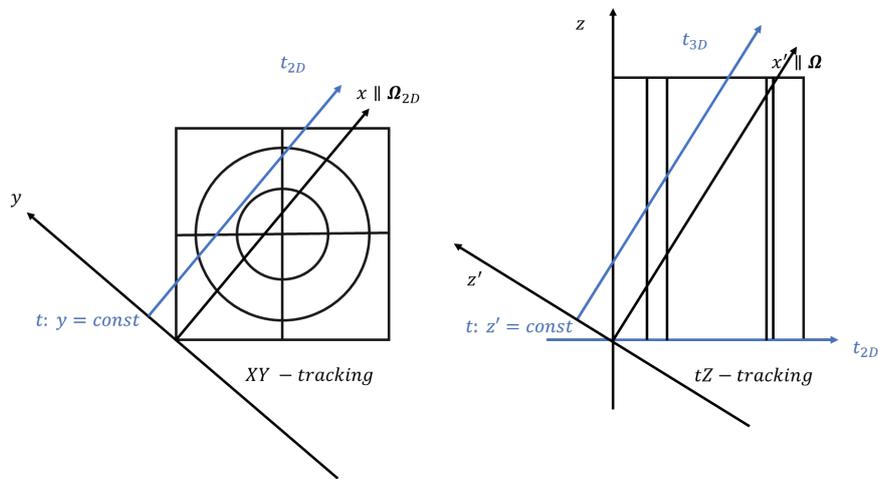


Figure 7.11: XY and tZ -tracking.

- a 2D-tracking along the horizontal component of direction Ω_d (left-hand-side of Fig. 7.11);
- a subsequent ray-tracing over the tZ -slice (right-hand-side of Fig. 7.11).

Thus, the (x, y) -coordinates describe points that are, respectively, parallel and perpendicular to $\Omega_{d,2D}$. First, the geometrical discontinuities (e.g., the grid-circle intersection points, the interface-boundary mesh points and the points delimiting the projections of the cell and its rings) are rotated. This ensemble of transverse y -points is enriched by a set of uniformly-distributed points, whose spacing is adjusted according to a user-defined tracking parameter. The ordering of these points defines a set of transverse intervals (Fig. 7.12) within which N trajectories are distributed, N being a fixed number of Gauss points. Hence, in general, the resulting transverse weight is not constant, i.e., it may change from one trajectory to another.

As depicted in Fig. 7.13, trajectory chord-lengths and region assignments are thereupon computed by combining the intersections of each trajectory with 3 geometrical patterns. These are, respectively,

- the vertical band pattern, defining segments of index i ,
- the horizontal band pattern, defining segments of index j ,

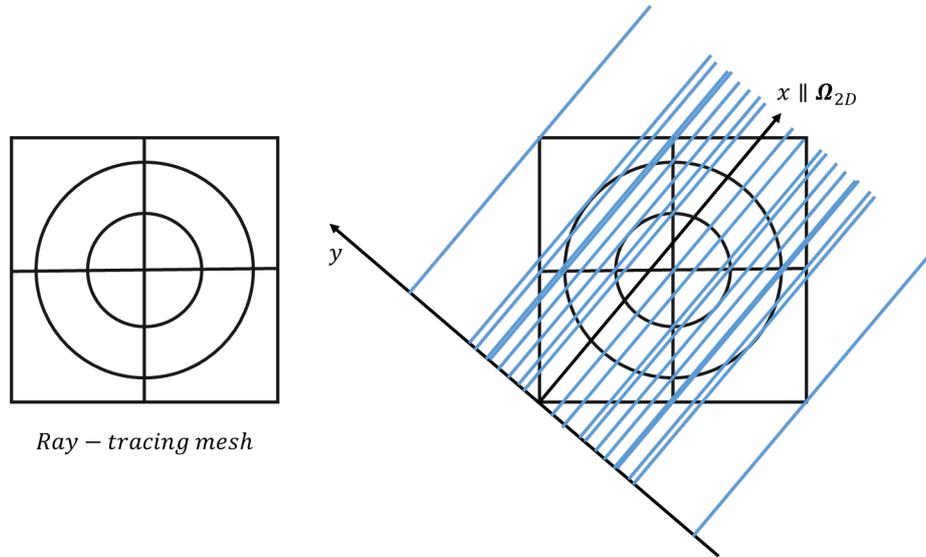


Figure 7.12: Step 1: Discontinuity projection + uniform ray spacing (omitted) + Gauss points (omitted).

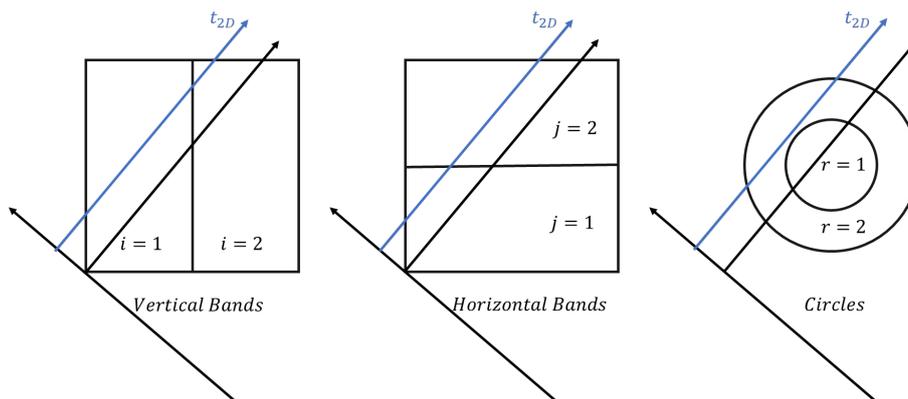


Figure 7.13: Step 2: Geometrical patterns.

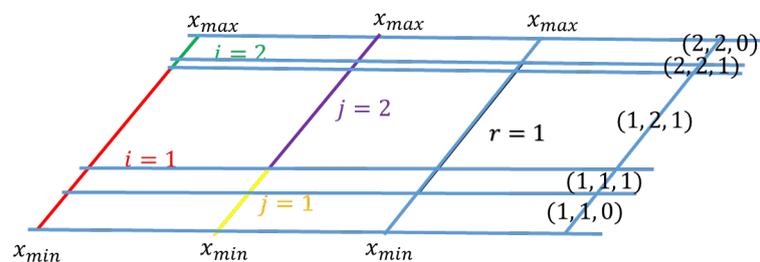


Figure 7.14: Step 3: Chord-lengths and TRM, with x_{min} and x_{max} being the coordinate of the incoming and outgoing point of the trajectory. Colour legend: red $\equiv i = 1$, green $\equiv i = 2$, yellow $\equiv j = 1$, purple $\equiv j = 2$, black $\equiv r = 1$.

- the set of rings, defining segments of index r .⁸

The ordering of the intersection points, together with the above-mentioned indexes, provides the 2D chord-lengths and the 2D tuple, of indexes (i, j, r) , associated to each ray-tracing region (Fig. 7.14).

The same 2D-tracking algorithm is applied to the horizontal and vertical bands composing the tZ -slice. The chord-lengths are finally computed by performing again Step-1-2-3, in Fig. 7.12-7.14, on the Cartesian slice and by combining the 2D TRM, associated to the vertical bands of the tZ -slice, with the index k , associated to the horizontal bands along Z . The region number, namely α , is then obtained by means of the TRM, as described in Sect. 7.3.

As mentioned in Sect. 7.3, despite the invariance of the ray-tracing mesh with respect to a translation along Z (Fig. 7.9), the TRM permits the construction of non-conforming/non-extruded 3D meshes, by adding up adjacent chords. Since two or more tuples can be assigned to the same computational region, the TRM may map the same computational-mesh index α onto several chord-lengths. Therefore, when performing the tZ -tracking, if two (or more) adjacent chord-lengths (l, l', l'', \dots) share the same computational region (α), but are associated to different ray-tracing regions, i.e., $f(i, j, r, k) = f(i', j', r', k') = (i'', j'', r'', k'') = \dots = \alpha$, with $(i, j, r, k) \neq (i', j', r', k') \neq (i'', j'', r'', k'') \neq \dots$, then the aforementioned chord-lengths are summed up and assigned to α .

7.4.1 . GEOMETRY TYPES

The accuracy of the new ray-tracing module has been verified by monitoring the maximum relative error with respect to the exact volume of 11 geometrical motifs, shown in Fig. 7.15. The test-cases include structured and unstructured geometries, based on heterogeneous Cartesian patterns, with a cell-pitch of 1.26 *cm*. The 11 geometries have been refined by superimposing a 6x6 Cartesian grid, with uniform step-length. Fig. 7.15 displays the relative error (*pcm*) obtained by refining the ray-spacing in two ways:

- no Gauss points and variable Δ_{XY} (uppermost figure);
- $\Delta_{XY} = 0.05$ *cm* and increasing number of Gauss points.

Note that the HCCs containing no circles, e.g., pattern nb. 5 in Fig. 7.15, are analytically integrated, i.e., the numerical error is systematically equal to zero.

⁸The structure of the implemented tracking module may allow for the introduction of other geometrical patterns, such as azimuthal sectors, spheres, oblique bands, to name a few. Hence, future developments could further enrich the ray-tracing capabilities of the code.

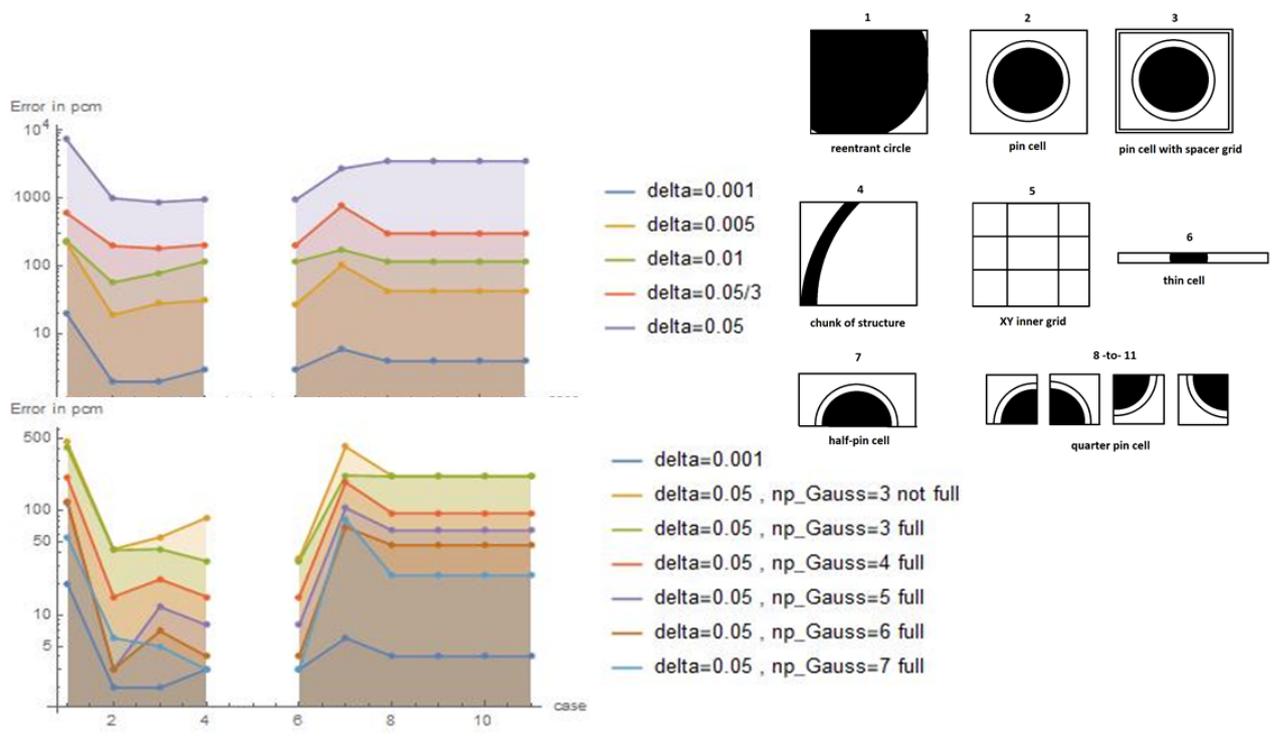


Figure 7.15: Volume computation for a set of heterogeneous Cartesian patterns. On the left-hand-side, the relative error obtained by varying the ray-spacing parameters (i.e., Δ_{XY} and number of Gauss points); on the right-hand-side, the geometrical motifs (labelled by an integer number, ranging from 1 to 11).

7.5 . HORIZONTAL HCCs

As anticipated in Sect. 7.1, the new version of IDT is able to simulate horizontal pin-cells. The user is allowed to dispose the cylinders parallel to either the x or y -axis, in addition to the more common orientation along z . Let us provide some more details about the mathematical model supporting this utility. In the following, two reference frames are introduced, namely

- a global frame $\mathcal{T} \equiv \{x, y, z\}$, where z is parallel to the vertical direction,
- a local frame, with coordinate system \mathcal{T}_{loc} defined by the triplet $\{x_{loc}, y_{loc}, z_{loc}\}$, where z_{loc} is parallel to the cylinder axis.

User's inputs provide the cell size and geometrical features with respect to $\mathcal{T}\{x, y, z\}$. In order to perform ray-tracing as in Sect. 7.4, IDT applies a set of affine transformations allowing coordinate change from \mathcal{T} to \mathcal{T}_{loc} . For the sake of brevity, let us consider a cell with cylinder axis aligned to x (Fig. 7.16). The angular direction is rotated by 90° around the y -axis, namely

$$\Omega_{loc} = \underline{\mathbf{R}}\Omega, \quad (7.3)$$

where

$$\Omega_{loc} = \begin{pmatrix} \Omega_{x_{loc}} \\ \Omega_{y_{loc}} \\ \Omega_{z_{loc}} \end{pmatrix}, \quad \Omega = \begin{pmatrix} \Omega_x \\ \Omega_y \\ \Omega_z \end{pmatrix}, \quad \underline{\mathbf{R}} = \begin{pmatrix} 0 & 0 & -1 \\ 0 & 1 & 0 \\ 1 & 0 & 0 \end{pmatrix} \quad (7.4)$$

are the components of Ω in the local/global frame of reference and the rotation matrix, respectively. The cylinder axis undergoes a rototranslation. Let \mathbf{r} be a point of coordinates (r_x, r_y, r_z) in $\mathcal{T}\{x, y, z\}$. In the local reference frame,

$$\mathbf{r}_{loc} = \underline{\mathbf{A}}\mathbf{r} \quad (7.5)$$

with

$$\mathbf{r}_{loc} = \begin{pmatrix} r_{x_{loc}} \\ r_{y_{loc}} \\ r_{z_{loc}} \\ 1 \end{pmatrix}, \quad \mathbf{r} = \begin{pmatrix} r_x \\ r_y \\ r_z \\ 1 \end{pmatrix}, \quad \underline{\mathbf{A}} = \begin{pmatrix} 0 & 0 & -1 & \sum_{k=1}^n \Delta z_k \\ 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}, \quad (7.6)$$

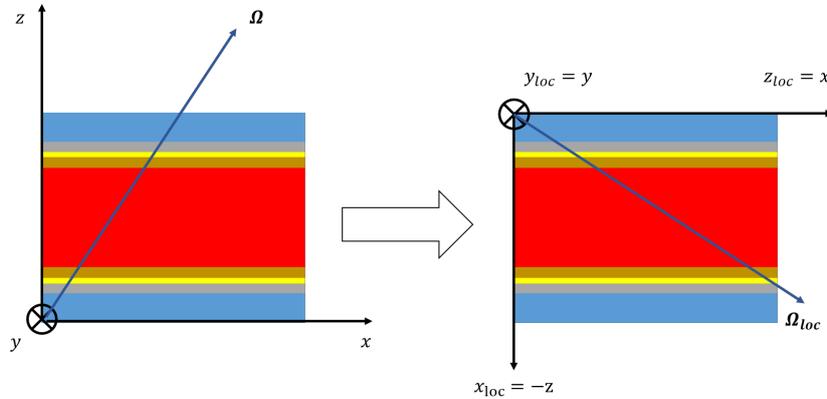


Figure 7.16: HCC orientation along x .

$\Delta \mathbf{z} = (\Delta z_1, \Delta z_2, \dots, \Delta z_n)^T$ being the axial steps of the inner Cartesian mesh. Thus, provided

$$\mathcal{C} \equiv \begin{cases} y = Y_c, \\ z = Z_c, \end{cases} \quad (7.7)$$

is the equation of the cylinder axis in the global frame of reference, one has

$$\mathcal{C}_{loc} \equiv \begin{cases} x_{loc} = \sum_{k=1}^n \Delta z_k - Z_c, \\ y_{loc} = Y_c. \end{cases} \quad (7.8)$$

in \mathcal{T}_{loc} . The Cartesian steps transform as follows:

$$\begin{cases} \Delta \mathbf{x}_{loc} = \mathbf{T} \Delta \mathbf{z}, \\ \Delta \mathbf{y}_{loc} = \Delta \mathbf{y}, \\ \Delta \mathbf{z}_{loc} = \Delta \mathbf{x}, \end{cases} \quad (7.9)$$

in which $\mathbf{T} = (\mathbf{e}_n; \mathbf{e}_{n-1}; \dots; \mathbf{e}_k; \dots; \mathbf{e}_1)$ inverts the order of the discretization steps. Note that \mathbf{e}_k is the k -th unit vector of the canonical basis in \mathbb{R}^n , with n being the number of Cartesian steps along the cylinder axis. Eq. (7.9) applies to both the cell-volume and interface-boundary discretization. Finally, the cell sides are mapped according to the law

$$\begin{cases} W_{loc} \rightarrow T, \\ S_{loc} \rightarrow S, \\ E_{loc} \rightarrow B, \\ N_{loc} \rightarrow N, \\ B_{loc} \rightarrow W, \\ T_{loc} \rightarrow E, \end{cases} \quad (7.10)$$

where W, S, E, N, B and T denote the west, south, east, north, bottom and top sides of the HCC.

An analogous set of isometric transformations allows to simulate horizontal HCCs, with the cell axis parallel to the y -direction.

Let us conclude this section with a practical example. Fig. 7.17 contains a set of non-extruded/non-conforming 3D geometries, which may be constructed by using the new version of IDT. Letter "I", "D" and "T" show respectively the possibility to

- represent non-extruded HCCs,
- model non-conforming Cartesian meshes inside the HCC and rotate the cylinder axis,
- combine horizontal and vertical cells in the same geometry.

7.6 . NUMERICAL RESULTS - 2D BENCHMARKING

The new version of IDT has been extensively tested, for several angular quadrature formulas and spatial discretizations, using constant and linear short characteristics. In the following, two benchmark types are illustrated, i.e., the 2D C5G7 benchmark and a set of 2D 3x3 UOX/MOX patterns. A dedicated chapter will illustrate the numerical precision and memory occupation of the angular probability coefficients.

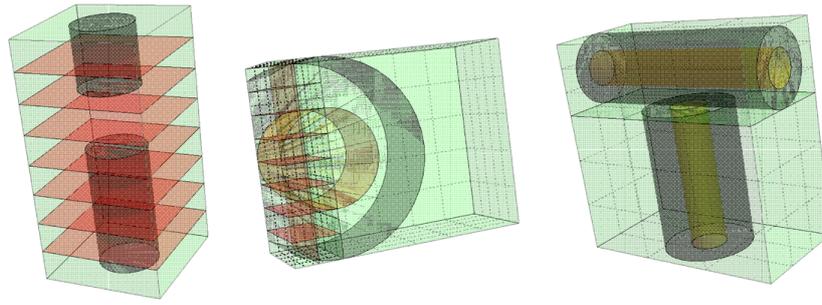


Figure 7.17: Non-conforming geometries and pin-cell orientation along different axes.

7.6.1 . Two-dimensional C5G7 Benchmark

The 2D C5G7 benchmark [83] is a small reactor core, comprising sixteen 17x17 UOX/MOX assemblies and a radial reflector. Each assembly consists of 264 fuel rods and 25 guide tubes, of which the one located in the assembly centre contains a fission chamber. A 7-group P_0 library is proposed to all participants, in order not to add any numerical error related to self-shielding. Note that the fuel rod and the surrounding clad as well as the guide tube and inner moderator/detector are homogenized. More details about the benchmark dimensions and MOX-enrichment distribution can be found in [83]. An intuitive sketch is provided in Fig. 7.18, in which each assembly is represented as a square, either containing UOX or MOX fuel rods. Reflection boundary conditions are imposed on the western and northern sides of Fig. 7.18, to profit of $1/4^{th}$ reactor symmetry.

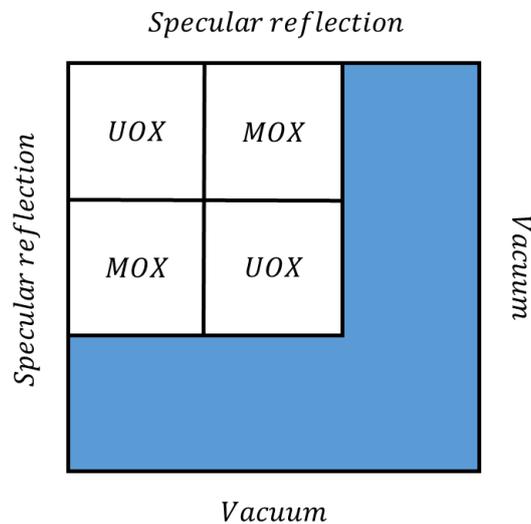


Figure 7.18: 2D C5G7 benchmark. Radial reflector in blue.

In Tab. 7.2, the solution provided by IDT is compared to the MCNP Monte Carlo reference, [83]. Both constant and linear short characteristics are applied. Tab. 7.1 provides details about the discretization used for CSC and LSC. Note that, for the latter, the non-automatic map generator is applied to coarsen the spatial mesh in the fuel rings, while keeping the surrounding moderator finely discretized.

Model	Nb. of Regions	Nb. of Surfaces per Side	Angular Quadrature
CSC	82331	6	$S_{16,8}$ Chebyshev-Legendre
LSC	14756	4	S_8 Level Symmetric

Table 7.1: Spatial and angular discretization.

Model	Δk_{eff}	Max Abs. Rel. Pin Error (%)	RMS (%)	CPU Time [min]
CSC	21	1.21	0.28	~ 4
LSC	41	0.59	0.17	~ 1.3

Table 7.2: Numerical results and CPU time on a single thread.

7.6.2 . Two-dimensional 3x3 UOX/MOX Patterns

IDT has been verified by comparison with APOLLO3® TDT MOC and TRIPOLI-4® continuous-energy Monte Carlo. The considered benchmark is a set of two-dimensional 3x3 motives, with specular reflection boundary conditions. The central cell alternatively hosts

- an AIC or B₄C control rod,
- a guide tube,
- a hafnium rod,
- a fuel rod containing Gadolinium.

Both IDT and TDT use 281-group P₃ cross-section library, obtained by Livolant-Jeanpierre self-shielding approximation. Generally, when comparing deterministic solutions to Monte Carlo, one should consider reaction rates. Nevertheless, in the present analysis, focus is given to the scalar flux per energy group and material. This may lead to significant numerical errors, as there are no compensations, due to the large number of energy groups.

Tab. 7.3 shows the number of regions for each simulation. Fig. 7.19 depicts the spatial mesh applied to UOX-AIC, when approximating the source and interface flux distributions by piece-wise polynomial functions (LSC). Analogously to Fig. 7.9, distinct computational regions are indicated in different colours. Note that TRM association allows to coarsen the underlying ray-tracing discretization, also represented in Fig. 7.19, thus reducing the memory footprint and the computational time. The solid angle is discretized into 256 discrete directions, using $S_{16,8}$ Chebyshev-Legendre angular quadrature.

Approx	AIC	Hf	GT	Gd	B ₄ C
CSC	387	401	472	358	383
LSC	128 (UOX) - 161 (MOX)	128	164	109	128

Table 7.3: Number of regions for each UOX/MOX pattern. Note that this number is the same for both fuel types, except for the AIC benchmark.

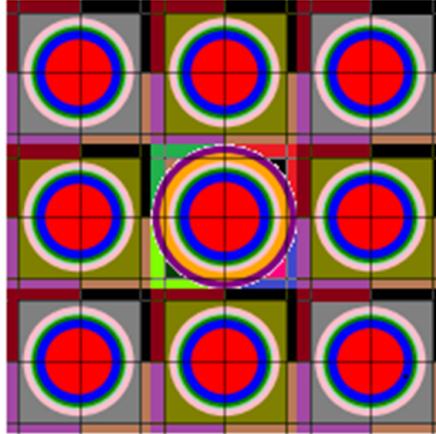


Figure 7.19: Spatial mesh for UOX/AIC LSC simulation. UOX cells comprise 11 computational regions (of which 4 UOX rings, 1 air-gap ring, 1 Zr-clad ring and 5 moderator subdivisions), while the central cell is finely discretized into 40 regions, where all rings (8, in total) are subdivided into 4 sectors and the surrounding moderator is finely discretized into 8 slices.

The Monte Carlo standard deviation of the scalar flux per energy group and material does not generally exceed 1%, except for a few energy groups in the innermost regions of the control rod, where the Monte Carlo score is too feeble. As displayed in Fig. 7.20, the errors observed for IDT and APOLLO3® TDT are of the same order. Their large magnitude is ascribable, on the one hand, to the fine energy grid and, on the other hand, to the self-shielding model, which may fail for sharp narrow resonances.

Fig. 7.21 displays the average root mean square relative error (RMS) per material. Six spatial approximations (namely, "NEW CONST IDT", "NEW LIN IDT", "OLD CONST IDT", "OLD LIN IDT" and "CONST A₃ MOC") are compared to the Monte Carlo reference. In particular, "OLD CONST/LIN IDT" is a synthetic label denoting all results obtained by means of the new version of IDT without introducing any Cartesian mesh. "NEW CONST/LIN IDT" refers to all calculations superimposing a XY grid onto the radial mesh and/or the surrounding moderator. Finally, "CONST A₃ MOC" is a label for APOLLO3® TDT MOC, with zero-order neutron source expansion.

Fig. 7.22 displays the relative error on the fundamental eigenvalue, for the whole set of calculations. Generally, the curves which show the largest difference with respect to the Monte Carlo reference are the ones not employing any Cartesian/azimuthal spatial refinement, i.e., "OLD CONST/LIN IDT", in grey and yellow, respectively. Fig. 7.22 also shows the CPU time on a single node for constant and linear short characteristics with XY-grid refinement. This typically ranges from 3.5 to 5 *min* for all UOX/MOX patterns.

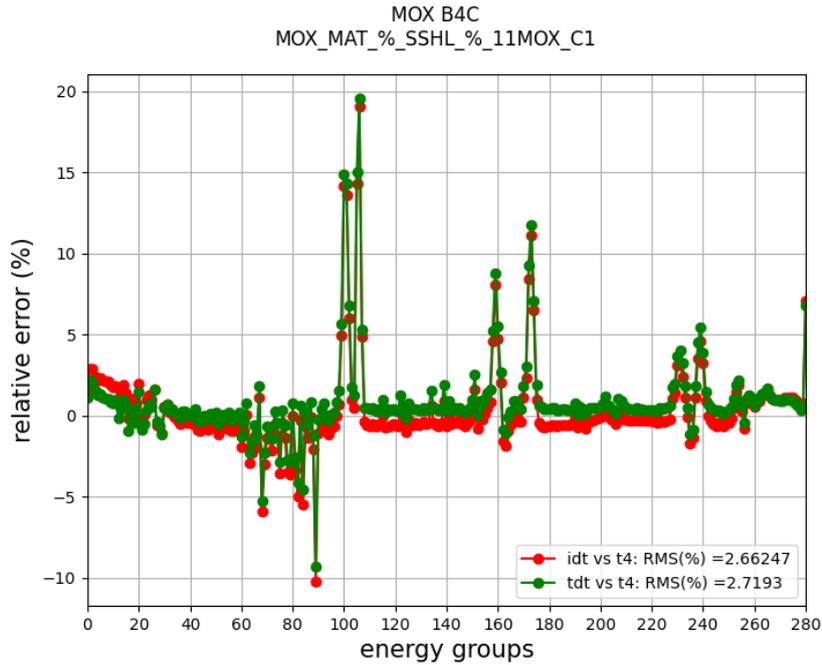


Figure 7.20: MOX-B₄C benchmark: relative error on the scalar flux per energy group and material region (reference provided by TRIPOLI-4® continuous-energy Monte Carlo). More specifically, the figure refers to the innermost fuel ring, within a cell in a corner position. The curve labeled "idt" (in red) is obtained with LSC spatial approximation. The error curves follow the same trend and show localized peaks within the same energy groups.

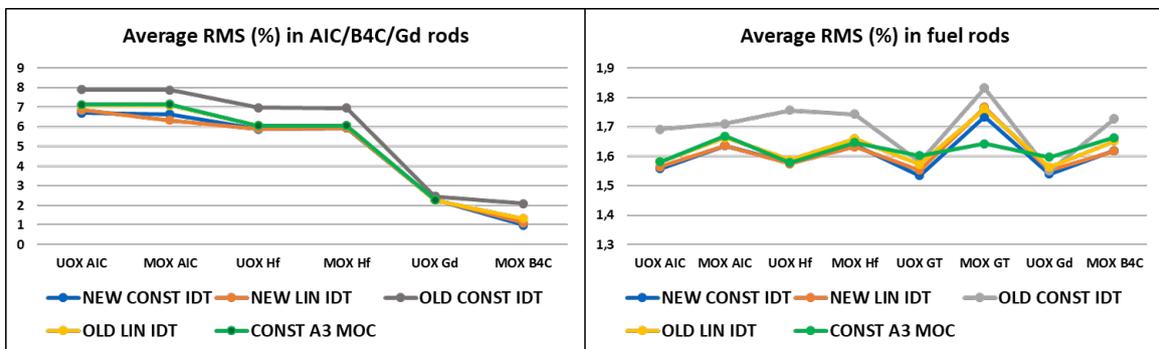


Figure 7.21: Average RMS relative error per material. The largest errors are observed in AIC and Hf rods. In the fuel rods, all calculations do not exceed 1.85% average RMS relative error.

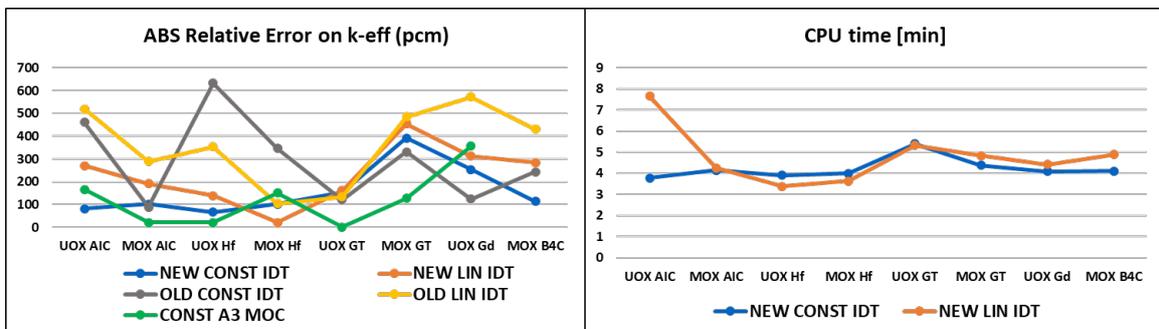


Figure 7.22: Relative error on the fundamental eigenvalue and CPU time.

7.6.3 . Hexagonal Patterns

The new version of IDT can model hexagonal assemblies, with little approximation. Such a geometry may be constructed by arranging quarter cells, as in Fig. 7.23, with the rings centered at a cell vertex. The fuel bundle is obtained as the result of consecutive "diagonal" cells, each one employing a circle of "infinite" radius, $\rho \rightarrow \infty$. In practice, one can require $\rho \gg l$, l being the cell pitch. The centre of this circle (\mathbf{c}) is located elsewhere and may be identified according to the following approach. First, let us define the chord

$$\mathbf{r} = \mathbf{p}_2 - \mathbf{p}_1, \quad (7.11)$$

where \mathbf{p}_1 and \mathbf{p}_2 are indicated in Fig. 7.24 The inward normal reads as

$$\mathbf{r}_\perp = \underline{\mathbf{R}}\mathbf{r}, \quad (7.12)$$

$\underline{\mathbf{R}}$ being the rotation matrix, namely

$$\underline{\mathbf{R}} = \begin{pmatrix} 0 & -1 \\ 1 & 0 \end{pmatrix}. \quad (7.13)$$

Let us introduce the distance between the centre of the circle (i.e., the fuel bundle) and the chord midpoint, \mathbf{m} ,

$$d = \text{dist}(\mathbf{c}; \mathbf{m}), \quad \mathbf{m} = \frac{\mathbf{p}_1 + \mathbf{p}_2}{2} \quad (7.14)$$

and request $d \gg l$. Hence, one has

$$\mathbf{c} = \mathbf{m} + \mathbf{n}d, \quad (7.15)$$

with

$$\mathbf{n} = \frac{\mathbf{r}_\perp}{\|\mathbf{r}_\perp\|}. \quad (7.16)$$

The bundle radius is given by

$$\rho = \text{dist}(\mathbf{c}; \mathbf{p}_2) \Rightarrow \rho^2 = d^2 + \left(\frac{\|\mathbf{r}\|}{2}\right)^2 \quad (7.17)$$

This algorithm allows, in a few steps, to build the fuel bundle of a hexagonal pattern, given

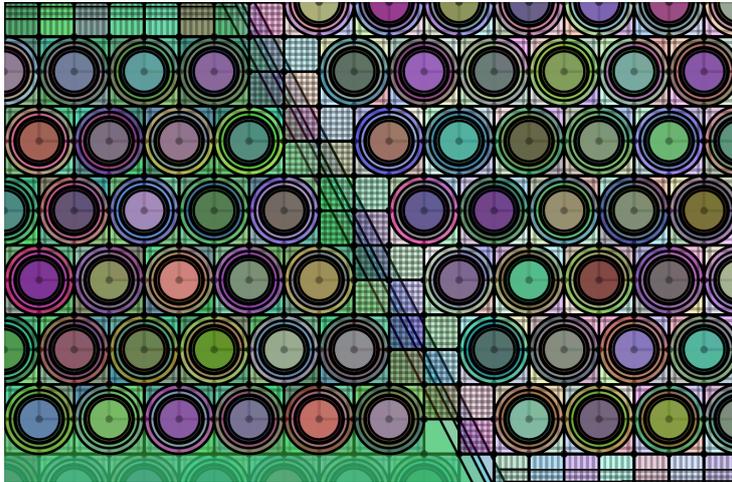


Figure 7.23: Hexagonal pattern, with fuel bundle.

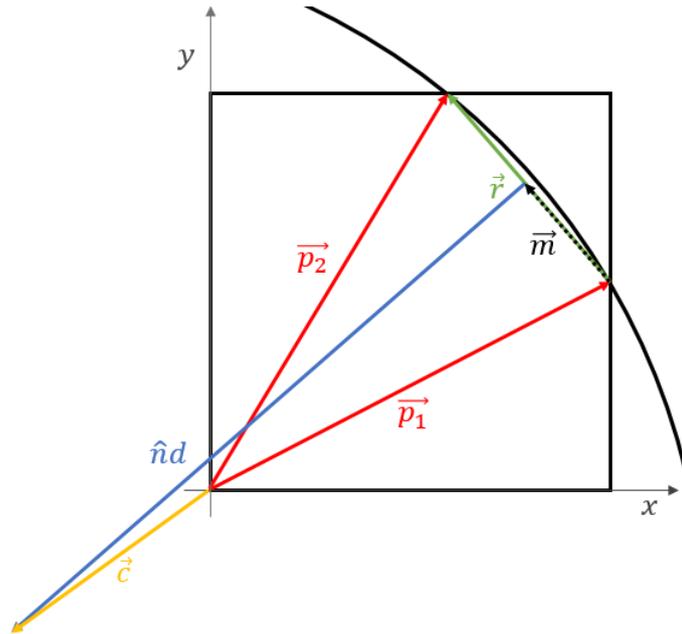


Figure 7.24: Fuel bundle construction for hexagonal patterns.

the position of the intersection points \mathbf{p}_1 and \mathbf{p}_2 , for the "diagonal" cells hosting the bundle. The sole approximation is the representation of the straight lines generating the fuel bundle by means of circular sections, albeit with a large radius.

Let us now present a simple test, involving a 2D hexagonal pattern, with UOX fuel and light water as moderator (Fig. 7.5). For the sake of simplicity, no fuel bundle is included. Specular reflection boundary conditions are imposed on the border. We use a 281-group P_3 cross section library. The solid angle is discretized by a $S_{16,8}$ Chebyshev-Legendre angular quadrature formula. The spatial discretization includes 21×14 quarter cells, arranged as in Fig. 7.5, a total of 3688 computational regions and 6×6 surfaces per cell. Linear Short Characteristics are applied to derive the spatial solution of the multigroup neutron transport equation.

Figures 7.25-7.26 display the fission integral and absorption reaction rate distributions, calculated by IDT. Note that both quantities are normalized to 1.

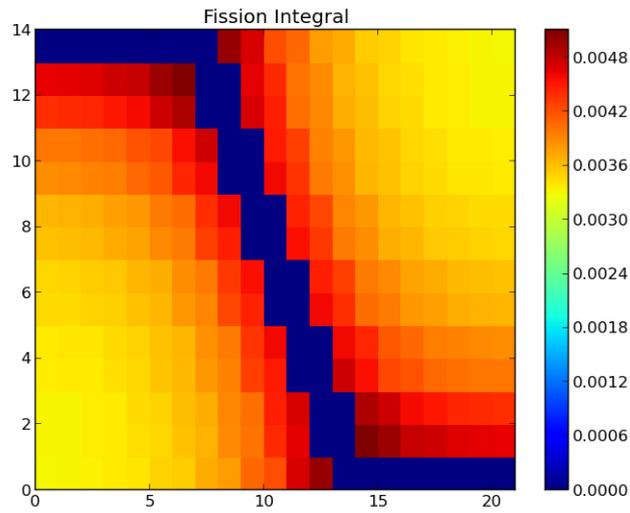


Figure 7.25: Fission integral distribution for a 21×14 two-dimensional hexagonal pattern, with UOX fuel and light water as moderator. Specular reflection boundary conditions are imposed on the border.

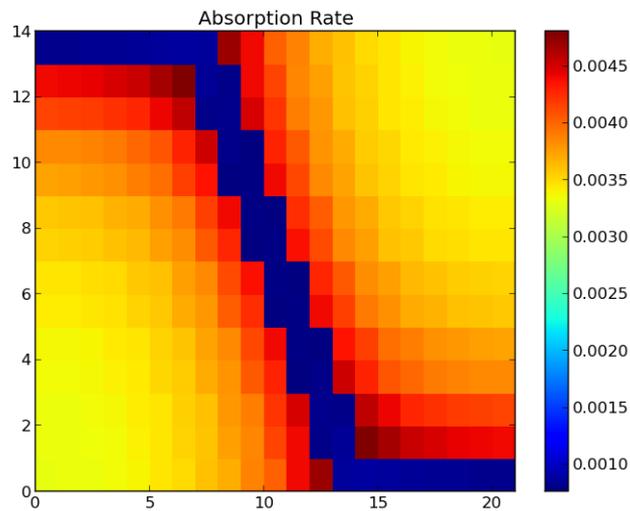


Figure 7.26: Absorption reaction rate distribution for a 21×14 two-dimensional hexagonal pattern, with UOX fuel and light water as moderator. Specular reflection boundary conditions are imposed on the border.

7.7 . NUMERICAL RESULTS - 3D 3x3 BEAVRS-LIKE BENCHMARK

The new version of IDT has also been verified on international and "home-made" 3D benchmarks. Several case studies are proposed in the present work, including the three-dimensional C5G7 benchmark (Sects. 8.3-11.5.1), the Watts Bar UOX/Pyrex assembly (Sect. 11.4.2) and a 3x3 BEAVRS-like lattice (Sect. 7.7.1-7.7.2). In this section, the latter is analyzed, in order to emphasize the possibility to represent all structural details in real 3D geometry (e.g., the spacer grids) and also to model non-conformal/non-extruded cells, which allow including geometrical discontinuities within the HCC. This generally improves the numerical accuracy, as the geometrical/material interface is not located on the cell border, where MOSC approximates the surface angular flux with piecewise polynomial functions.

In Sect. 7.7.1, the benchmark is illustrated, providing details on the lattice geometry and the self-shielding model. In Sect. 7.7.2, the numerical results are analyzed for each study case.

7.7.1 . Introduction

The numerical accuracy of IDT is tested by comparison with TRIPOLI-4® continuous-energy Monte Carlo, on a set of three-dimensional 3x3 lattices (Fig. 7.27). The central position is occupied by a control rod, which consists of two parts: a silver-indium-cadmium alloy in the bottom region and a B_4C rod in the upper region. The heterogeneous geometry is constructed taking inspiration from the BEAVRS benchmark [59], both in IDT and TRIPOLI-4®, including the spacer grids and the upper plenum springs (in green), the lower and upper nozzles (in pink), the fuel clad (in grey), the guide-tube dashpot, to name a few.

A premise on the self-shielding model is crucial. Indeed, the cross-section library (with 281 energy groups and P_3 transfer matrix) is entirely recycled from the set of 2D 3x3 UOX patterns, discussed in Sect. 7.6.2. This, of course, introduces an inherent modelling error, as a bunch of 2D planes (e.g., the one corresponding to the guide-tube dashpot and the ones hosting the spacer grids) contain inappropriate nuclear data.

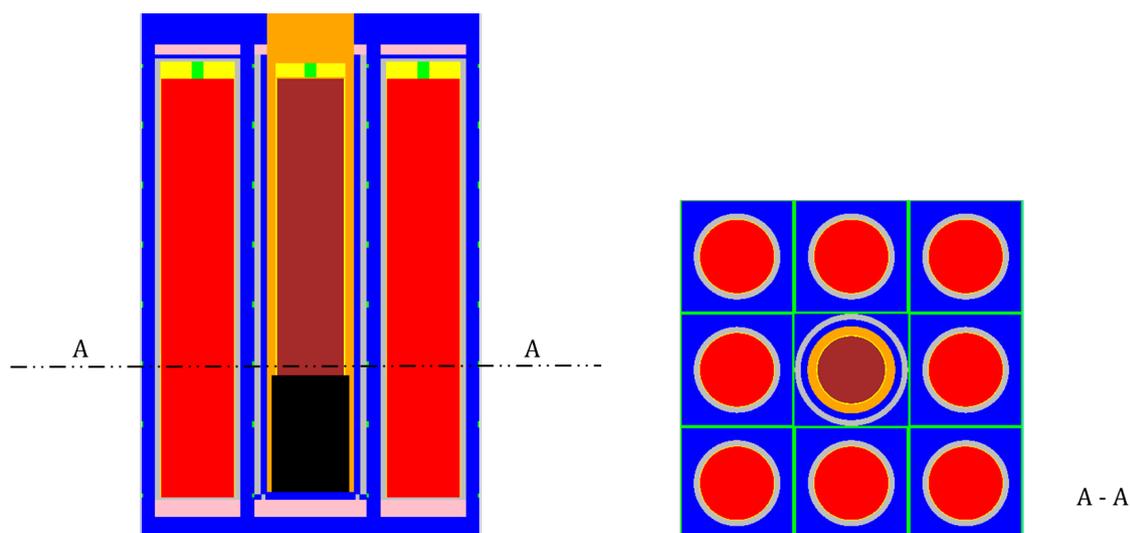


Figure 7.27: CR completely inserted. Image by TRIPOLI-4®.

In the following, three control-rod (CR) positions are simulated, i.e., CR completely inserted,

Configuration	Nb. of Cells	Ave Nb of Reg./Cell	Tot Nb of Dir.	Surf/Cell
CR all in	78	40	256	144
CR at half height	46	179	256	66
CR out	46	171	256	66

Table 7.4: IDT: spatial and angular discretization.

CR tip at half height and CR completely withdrawn (Fig. 7.28).

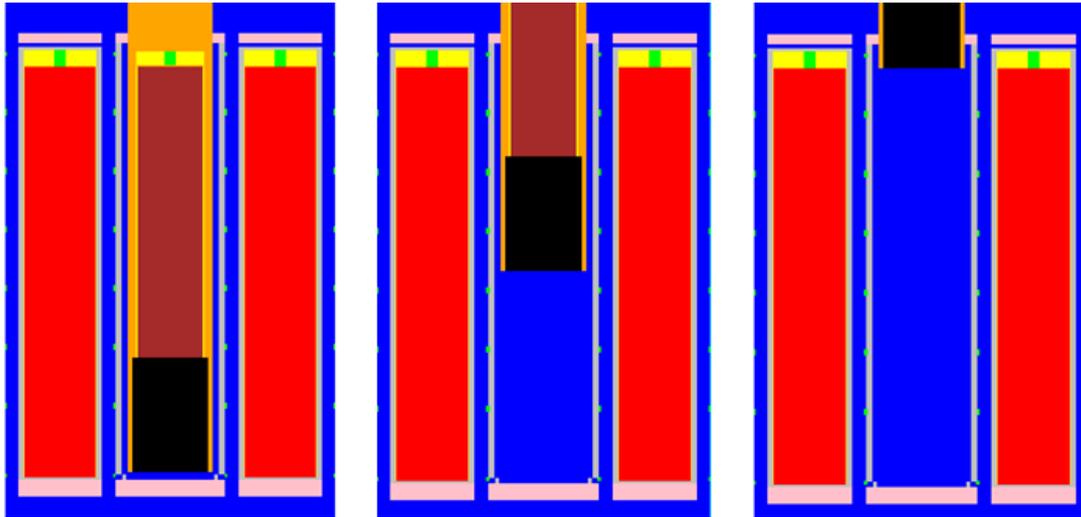


Figure 7.28: CR completely inserted (on the left-hand-side), at half height (in the centre) and completely extracted (on the right-hand-side). Image by TRIPOLI-4®.

7.7.2 . Numerical Results

Tab. 7.4 shows the average number of regions per cell, the total number of directions and the total number of surfaces per cell used in IDT for each CR position. Fig. 7.29 provides a qualitative comparison between IDT and TRIPOLI-4®, in terms of fission reaction rate, for the above-mentioned CR positions. Note that the fission rate has been normalized to 1 for each rod. For the sake of brevity, only one type of fuel rod will be considered, i.e., the one in the bottom-left corner. Analogous results are observed elsewhere.

When the CR is completely inserted, most fissions occur in the bottom region of the reactor, due to the presence of B_4C in the upper region. The curves labeled "idt - cr inserted" and "t4 - cr inserted" tend to overlap, as depicted in Fig. 7.29. The fission reaction rates provided by IDT and TRIPOLI-4® are compared by displaying the absolute error along the z-axis, in Fig. 7.30 ("idt vs t4 - cr inserted"). The largest error is observed in the region occupied by the guide-tube dashpot, due to inappropriate cross sections on the dashpot plane. A dedicated self-shielding calculation would be required to generate accurate nuclear data for IDT usage. The same consideration applies to the fuel nozzles, the spacer grids and the upper plenum springs. Thus, some amount of error is not ascribable to IDT, but to the cross section model itself. This said, IDT shows good agreement with TRIPOLI-4®, as the absolute error barely exceeds 5%. The standard deviation of the fission rate provided by the Monte Carlo simulation tends to grow along the z-axis, but remains lower than 1%.

In the second simulation, the CR tip is located at half height of the reactor. A qualitative comparison between IDT and TRIPOLI-4® is shown in Fig. 7.29 ("idt - cr at half height" vs "t4 - cr at half height"). The absolute error on the fission reaction rate is at most 6% (Fig. 7.30, "idt vs t4 - cr at half height"). Again, the standard deviation of the Monte Carlo simulation tends to grow along the vertical axis, but remains smaller than 1%, if the fission rate is not too close to zero, which is the case for the uppermost planes.

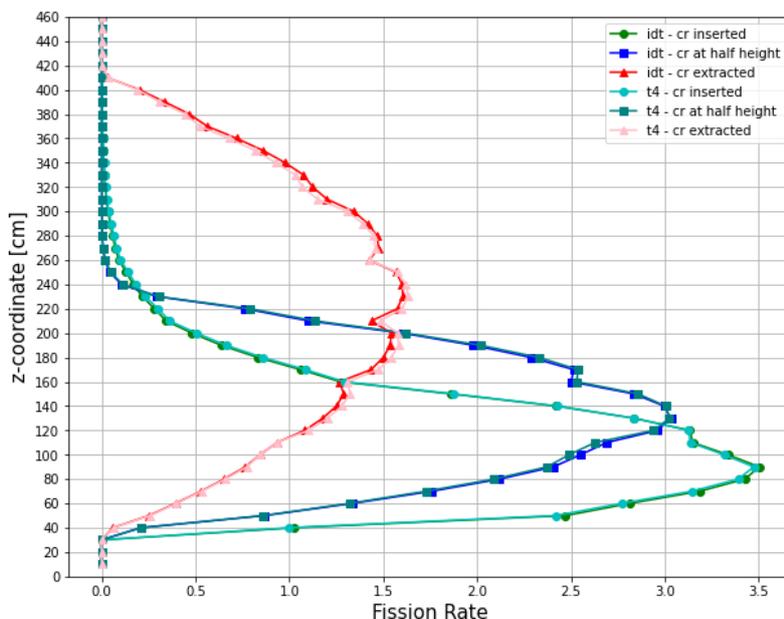


Figure 7.29: Qualitative comparison between IDT and TRIPOLI-4® continuous-energy Monte Carlo, for each CR position.

Finally, the CR is extracted, so that the rod tip lines up with the fuel plenum. The irregular shape of the fission-reaction-rate profile is due to the spacer grids (Fig. 7.29, "idt - cr extracted" vs "t4 - cr extracted"). The maximum absolute error on the fission rate is between 5% and 6% (Fig. 7.30, "idt vs t4 - cr extracted"). The standard deviation of the fission reaction rate predicted by TRIPOLI-4® is always below 0.1%.

To conclude, let us provide some more details about the prior calculations. For each position of the CR, Tab. 7.5 compares the k -eigenvalue provided by IDT and TRIPOLI-4®, in terms of relative error. The largest difference is observed for the first configuration and amounts to 360 pcm. Tab. 7.6 displays the error on the control rod worth, which is, in absolute value, below 0.81%. The CPU time for each configuration ranges from 4 to 6 hours (Tab. 7.7), depending on the spatial discretization (Tab. 7.4) and the computational resources (Tab. 7.7). In this regard, a clarification is required for Tab. 7.7. As shown in the Tab. 7.4, the number of regions for the case with the control rod fully inserted is on average equal to 40 regions per cell. On the contrary, the cases with the control rod completely extracted or with the tip at half height are much more finely discretized, having more than 170 regions per cell. This has a considerable impact both on the simulation runtime necessary for each calculation and on the memory occupation, which will be about 18 – 20 times greater for the angular probability coefficients alone. Therefore, in order to store the angular probability matrices and have a relatively limited calculation time for the cases where the control rod is partially or completely withdrawn, decidedly greater

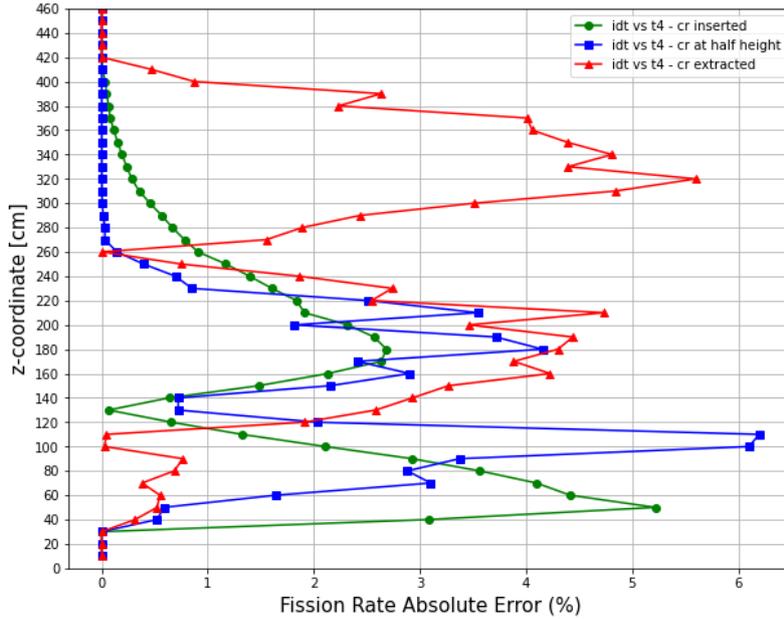


Figure 7.30: Absolute error with respect to TRIPOLI-4® continuous-energy Monte Carlo, for each CR position.

Configuration	IDT	T4	σ (pcm)	Relative error (pcm)
CR all in	0.76958	0.77236	0.8	359
CR at half height	1.30044	1.30116	1.2	55
CR out	1.31234	1.31314	1.2	60

Table 7.5: IDT vs TRIPOLI-4®: k -eigenvalue.

computational resources are required compared to the first calculation, i.e., the one with the control rod completely inserted in the lattice. In this sense, the reader should also notice that the resources indicated in Tab. 7.7 correspond to the maximum amount of nodes and cores allocated for the simulation, but not necessarily used by the code.

Finally, it is important to note that, although the spatial dimensions are evidently very small, particularly in the transversal plane, which measures just $3.78 \text{ cm} \times 3.78 \text{ cm}$, the spatial and angular discretization are particularly refined, with a total number of regions that varies between $2.8 \cdot 10^4$ (control rod fully inserted) and $7.4 \cdot 10^4$ (control-rod tip at half height), and 32 angular directions per octant. The purpose of this test is in fact to test the new modelling capabilities of the code, rather than the agreement with the continuous-energy Monte Carlo reference, which is, to some extent, compromised by the approximations on the cross-section data.

	CRW error (%)
all in/half height	-0.809
half height/out	0.627

Table 7.6: IDT vs TRIPOLI-4®: CR worth error (%).

Configuration	CPU Time	Nodes and Cores/Node
CR all in	5h 51min	1 n – 128 c/n
CR at half height	5h 46min	46 n – 128 c/n
CR out	4h 14min	46 n – 128 c/n

Table 7.7: IDT: CPU time and allocated resources.

7.8 . CONCLUSIONS

The present chapter has illustrated the new geometrical capabilities of IDT. The solver has been updated, in order to improve the numerical accuracy of 2D/3D calculations, by introducing non-conformal and non-extruded Cartesian grids, which can be superimposed on the radial mesh. In Sect. 7.6-7.7, a few numerical study cases have been discussed, in order to test the ensemble of capabilities introduced in the solver, as well as its numerical precision. In the upcoming chapters, the new geometrical model will be applied for a variety of purposes, as the possibility to simulate control-rod withdrawal without homogenization and cusping effect (Chap. 8) or test the accuracy of higher-order short characteristics by comparison with finely-discretized LSC calculations (Chap. 11). In particular, the possibility to refine the inner discretization, while preserving the HCC grid, allows to ensure numerical stability for multiphysics applications involving control-rod movements, with high-fidelity representation of control-rod insertion/extraction by steps of finite length, in real heterogeneous geometry, with no homogenization and cusping effect.

8 - NON-CONFORMING 3D MODEL FOR PWR CONTROL ROD MOVEMENTS WITHOUT HOMOGENIZATION

8.1 . INTRODUCTION

In Chapter 7, the implementation of enhanced geometrical capabilities for high-fidelity representation of fission reactors was discussed. The present chapter is dedicated to 3D control-rod modelling in PWRs. In particular, we propose a transport-based model to avoid the control-rod cusping effect generated by cross-section homogenization.

Due to the geometry of PWR cores, the height of a node generally ranges from 10 to 20 *cm*. On the other hand, the control rods move along the vertical direction with a step of few centimeters. As a result, a control rod cluster may be partially inserted into a set of Cartesian nodes, thus requiring a heterogeneous representation of the cross sections along z .

In traditional 3D transport codes, the cross-sections cannot be modelled by piece-wise distributed functions within a node. Consequently, despite its heterogeneous nature, the node has to be homogenized. Due to the sharp axial heterogeneity, the cross sections can neither be represented by polynomial expansions at the interface between the control rod tip and the water. Currently, the solvers of APOLLO3® do not allow for non-conformal geometries along the z -axis, i.e., all 2D regions have the same number of axial planes.

Generally, high-fidelity transport-based core solvers deal with this issue by flux-weighting homogenization techniques. Nevertheless, the simple homogenization triggers a discontinuity in the first derivative of the curve representing the variation of the fundamental eigenvalue as the controls rods are inserted in the reactor. This non-physical phenomenon is known as the **control-rod cusping effect** (Sect. 5.5). Several numerical remedies have been proposed over the last few decades. For instance, for diffusion, P_N or SP_N -based solvers, Fanning and Palmiotti have proposed a heterogeneous Variational Nodal Method (VNM), [84]. Another heterogeneous VNM was developed by Smith et al., in 2003, [85]. This approach consists in splitting the spatial node into sub-elements with uniform cross sections. The flux is then developed by finite spatial trial functions. The functional of all sub-elements composing the node allows to reconstruct the nodal functional.

In the present work, discrete-ordinate Linear Short Characteristics are applied to 3D Heterogeneous Cartesian Cells. To ensure accurate representation of the control-rod movements, each heterogeneous node, represented by a HCC, is the combination of an inner local tridimensional grid and a set of cylinders representing the fuel rod in its exact shape (Sect. 7.2). The new geometrical model generates local non-conforming geometries. Spatial integrals over the HCC are performed using a modular tridimensional ray-tracing. The ray-tracing technique is based on the combinatorial geometry composed of cylinders and local XYZ planes (Sect. 7.4). In this manner, a single HCC can be equipped with several local steps to follow exactly the axial displacement of the control rod. Neither homogenization nor mesh adjustment is required. Numerical results show that IDT HCC-based transport can eliminate the numerical wiggles in the differential control-rod reactivity-worth curve.

In Sect. 8.2, a brief overview on the control-rod cusping effect is proposed and some notable numerical remedies are recalled. In Sect. 8.3, conformal and non-conformal meshes are applied on the 3D C5G7 benchmark, showing similar numerical accuracy, with respect to the Monte Carlo reference (MCNP). Then, Sect. 8.5 resorts to a non-conformal mesh to compute successive reactor states, corresponding to different control-rod positions, and proves that IDT can effectively remove the control-rod cusping effect. In order to perform such calculation, a new tool is available in IDT, which allows changing the material composition, within a fixed geometry, in an automatic way. This capability is described in Sect. 8.4 and permits to determine several reactor states, one for each control-rod position, by using a single input and launching IDT just once.

8.2 . CONTROL-ROD CUSPING EFFECT

Control-rod modelling in nodal methods represents a major challenge. In fact, the position of the control-rod tip rarely matches the node boundaries. The cross sections of the upper part of the node are modified due to the presence of the control rod, whereas the lower part remains unchanged. Cross-section homogenization by volume-weighting triggers a non-physical behaviour, as the steep gradient of the neutron flux at the interface between the rod tip and the moderator is smeared out. Reaction rates are clearly not preserved and cusps appear in the reactivity-worth curve (Fig. 8.1).

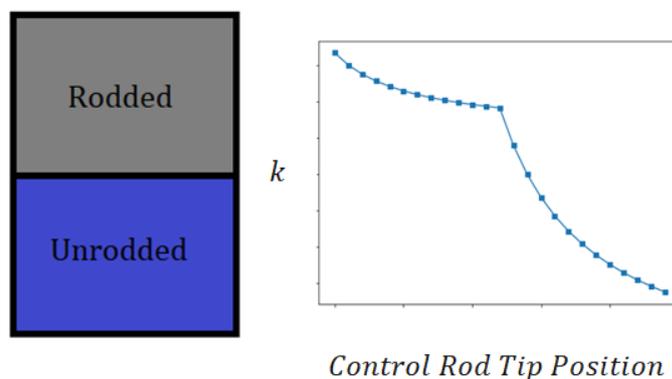


Figure 8.1: Control-rod cusping effect.

Several numerical techniques have been proposed to fix this behaviour. In particular, flux-weighting techniques can reduce the rod cusping effect, [86].

In [64], a model based on a physical equivalence is presented, where the reaction rates of the classical nodal expansion are required to match those obtained by modelling the node as it was divided in two parts, the upper one occupied by the strong absorber and the lower one unrodded, Sect. 5.5.2. Despite generally outperforming flux-weighting methods in terms of numerical precision, the equivalence model still exhibits mild wiggles in the differential worth curve.

In MPACT, control-rods simulation is supported by a set of subplane-based methods, [65]. This approach splits each plane into multiple sub-planes for CMFD and 1D NEM-P3 calcula-

tions, while still using only a single plane for radial MOC (Sect. 5.30). In so doing, the numerical accuracy of CMFD and 1D NEM-P3 is enhanced, without increasing the computational cost of the radial calculations. However, sub-planes based on collision probabilities, as in [65], cannot handle anisotropic scattering sources. Moreover, numerical instabilities may occur, due to very thin cells.

The new version of IDT is able to describe control-rod movements in an exact way, without resorting to any homogenisation and so avoiding rod-cusping effects. In the following, the results obtained for the 3D C5G7 benchmark are presented, [83]. Starting from the Unrodded configuration, the control-rod insertion is simulated by 24 successive steps (Fig. 8.2), passing through the Rodded A configuration (step 12 out of 24) and finally approaching the Rodded-B configuration (Sect. 8.5). Before that, let us compare conforming vs non-conforming mesh

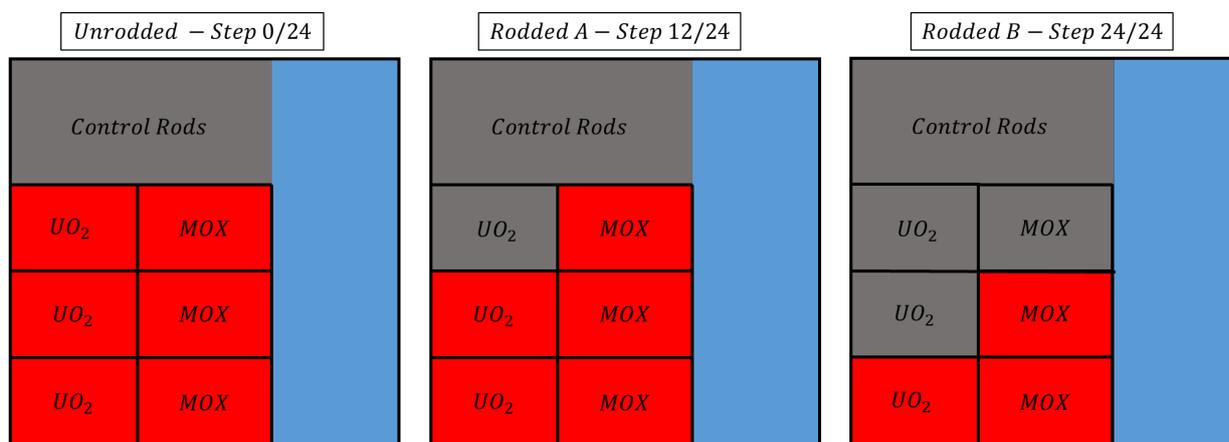


Figure 8.2: Test based on the 3D C5G7 benchmark: the control rods are gradually inserted, by 24 steps of 1.19 cm each.

application, in terms of numerical accuracy and computational time (Sect. 8.3).

8.3 . 3D C5G7 BENCHMARK: CONFORMAL VS NON-CONFORMAL MESH

The 3D C5G7 benchmark consists of a small reactor core, with UOX/MOX fuel assemblies and respective control-rod clusters, arranged in 3 different configurations, [83], namely the Unrodded, Rodded A and Rodded B problems (Fig. 8.2). The control rod clusters are inserted

- in the axial reflector,
- in the axial reflector + $1/3^{rd}$ of the way into the inner UOX assembly (Fig. 7.18),
- in the axial reflector + $2/3^{rd}$ of the way into the inner UOX assembly + $1/3^{rd}$ of the way into both MOX assemblies,

for the Unrodded, Rodded A and Rodded B configurations, respectively.

In this section, the results provided by a conformal mesh are compared to those obtained using a non-conformal mesh. In the former case, the fuel rods, the control rods and the moderator contain 3 horizontal planes per cell (Fig. 8.3a); in the latter, the computational mesh is coarsened, with only the control rods discretized by 3 axial steps per cell (Fig. 8.3b). Each HCC

measures $1.26\text{ cm} \times 1.26\text{ cm} \times 3.14\text{ cm}$.

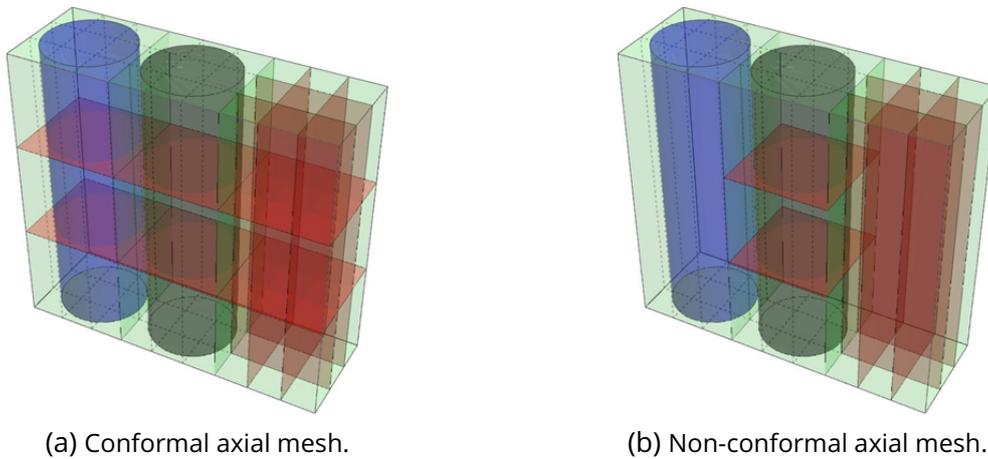
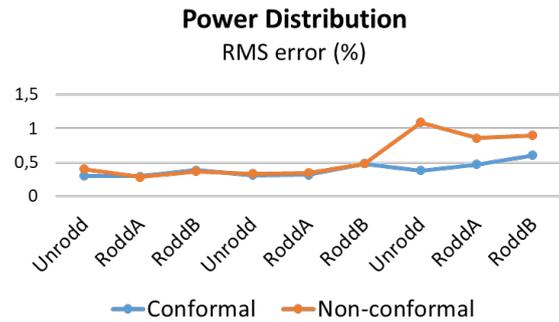
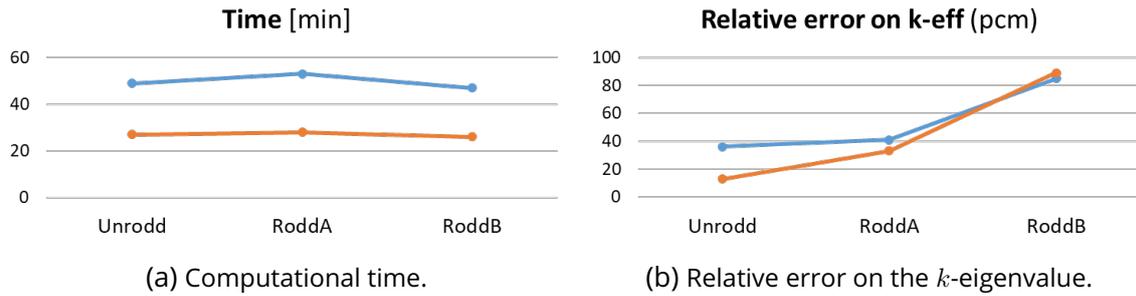


Figure 8.3: Conformal vs non-conformal HCCs along z.

If the second discretization is applied (Fig. 8.3b), a significant saving in the computational time is observed, with a reduction of nearly 50%, as shown in Fig. 8.4a, for the three configurations of the C5G7 benchmark. As depicted in Fig. 8.4b, there is no significant change in the fundamental eigenvalue predicted by IDT. Regarding the power distribution, the largest difference is observed in the top layer, due to the proximity to the axial reflector. Nonetheless, the RMS error is at most 1% for the Unrodded configuration (Fig. 8.4c).

In Sect. 8.5, a fine inner grid is used only in the nodes hosting the control rod, whereas a coarser volume grid is applied elsewhere (e.g., fuel rods and fission chambers). In this manner, the heterogeneous geometry can be represented with no approximation, while retaining a coarser mesh where possible, in order to limit the computational burden.



(c) Root mean square error on the power distribution.

Figure 8.4: C5G7 benchmark: conformal vs non-conformal mesh.

8.4 . AUTOMATIC MEDIUM RELOAD

In order to compute multiple reactor configurations, involving different control-rod positions, IDT relies on a novel tool, which allows to redefine the material composition of a subset of HCCs, automatically compute the new set of angular-dependent probability coefficients and run a new steady-state calculation, for each assigned configuration. Note that the underlying geometry is defined once for all and cannot be modified.

Once computed the first reactor state, namely $s = 1$, corresponding to the initial position of the control rods, IDT can automatically initiate a new simulation, each time a specific input command is detected. The user can indeed reload the material composition of one or more cells, to define a new reactor configuration. First, IDT reads the number of cells that the user intends to modify, and a loop of the kind illustrated in Algo. 1 is performed. In order to calculate the new steady-state point, namely $s + 1$, let us impose $k_{s+1}^{(0)} = k_s, k_{s+1}^{(0)}$ and k_s being respectively the initial guess for the current reactor state, namely $s + 1$, and the converged solution for the prior state, namely s , with $s = 1, \dots, S - 1$. This simple strategy allows to speed up the convergence. In fact, two successive reactor states are typically close, for two subsequent positions of the control rods, as they normally move with a small step size along the vertical direction.

The described tool is applied in Sect. 8.5, in order to compute the reactivity curve resulting from control-rod insertion in the 3D C5G7 benchmark.

Algorithm 1 : HCC-reload algorithm, in order to compute state $s + 1$, given s .

```
1: for  $i = 0 \rightarrow n$ ,  $n$  being the number of HCCs to refill, do
2:   Detect pin-cell name
3:   if pin-cell  $\notin$  pin-cell list then
4:     Error message, code comes to a halt
5:   else
6:     Determine pin-cell identifier,  $u_p$ 
7:     Determine  $u_g = G(u_p)$ ,  $g$  being the pin-cell geometry and  $G : u_p \rightarrow u_g$ 
8:     Read material list,  $\{m_l\}_{l=1,\dots,L}$ 
9:     if  $u_g$  by automatic TRM then
10:      Initialize  $l = 0$ 
11:      for  $k = 1 \rightarrow n_k$  {Loop over planes} do
12:        for  $\rho = 1 \rightarrow n_\rho$  {Loop over inner rings} do
13:           $l \leftarrow l + 1$ 
14:          Refill regions  $\in \rho, k$ , with material  $m_l$ 
15:        end for
16:         $l \leftarrow l + 1$ 
17:        Refill regions  $\in \rho = 0, k$ , with material  $m_l$  {Outer ring}
18:      end for
19:      else if  $u_g$  by non-automatic TRM {Custom option} then
20:        for  $r = 1 \rightarrow n_r$ ,  $n_r$  being the number of comp. regions do
21:          Determine region index,  $\alpha$ 
22:          Refill  $\alpha$  with  $m_r$ 
23:        end for
24:      end if
25:    end if
26:  end for
27:  Recompute coefficients
28:  Determine new reactor state,  $s + 1$ , given  $k_{s+1}^{(0)} = k_s$ 
```

8.5 . CONTROL-ROD MOVEMENT IN C5G7

In Fig. 8.5, three simulations, performed by IDT, are presented:

- A reference calculation, consisting of $51 \times 51 \times 1$ Cartesian nodes, where the heterogeneous geometry is represented in an exact way. Each node covers the whole core height, which amounts to 64.26 cm . In Fig. 8.5, this simulation is referred as "Ref-IDT";
- A second calculation, labeled "Het-IDT-18", where the spatial domain is discretized into $51 \times 51 \times 18$ cells. Once again, the heterogeneous geometry is modelled with no approximation. The height is 3.57 cm for all HCCs;
- A third calculation, referred as "Homo-IDT-18", which shares the same discretization as "Het-IDT-18". Differently from "Ref-IDT" and "Het-IDT-18", the control-rod and guide-tube cross sections are homogenized in the nodes where the control-rod tip does not match the node boundary.

Fig. 8.5 provides the value of k_{eff} as the control rods are gradually inserted, with a step size of 1.19 cm . The curves resulting from "Ref-IDT" and "Het-IDT-18" exhibit no cusping effect. Conversely, the curve labeled "Homo-IDT-18" is affected by numerical wiggles, caused by cross-section homogenization. The figure also compares "Het-IDT-18" and "Homo-IDT-18" to "Ref-IDT", by displaying the relative difference on the fundamental eigenvalue (dashed lines in Fig. 8.5).

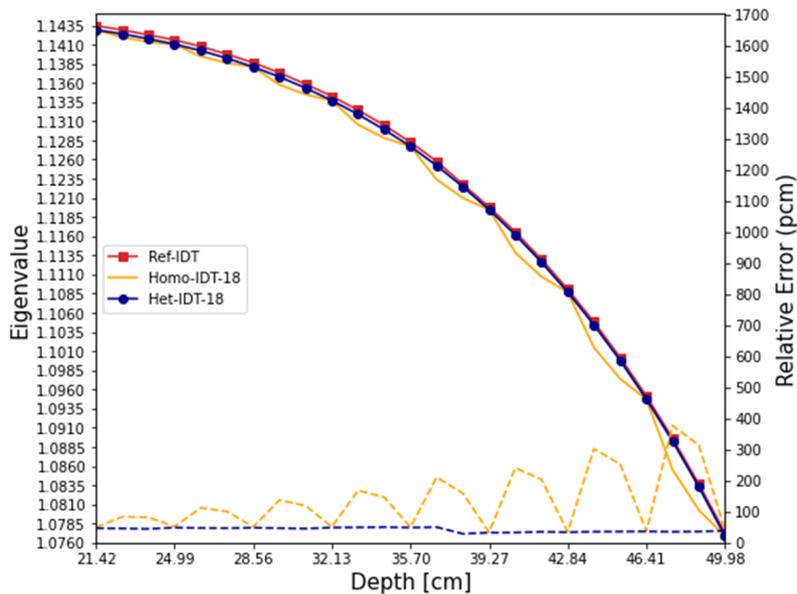


Figure 8.5: Control-rod simulation in the C5G7 benchmark, using nodes with $\Delta h = 3.57 \text{ cm}$: k_{eff} vs control-rod position (solid lines) and relative error with respect to "Ref-IDT" (dashed lines). N.B. The control rods are gradually inserted with a step size of 1.19 cm .

However, generally the axial size of a node is about $10 - 20 \text{ cm}$. On the other hand, the control rods move along the axial direction with a step size of about $1 - 2 \text{ cm}$. If the control rods are homogenized with the underlying moderator in a node of these dimensions, larger wiggles will appear in the numerical solution.

In Fig. 8.6, three simulations, performed by IDT, are compared:

- The aforementioned "Ref-IDT";
- A second calculation, labeled "Het-IDT-4", performing no homogenization, which splits the spatial domain into $51 \times 51 \times 4$ nodes. The axial dimension of one node amounts to 14.28 cm (21.48 cm for the upper reflector);
- A third calculation, referred as "Homo-IDT-4", which shares the same geometry as "Het-IDT-4". Differently from "Ref-IDT" and "Het-IDT-4", the control-rod and the guide-tube cross sections are homogenized in the nodes where the control-rod tip does not match the node boundary. The axial dimension of such nodes is 14.28 cm .

Due to the large axial step, "Homo-IDT-4" shows prominent wiggles and very large differences with respect to "Ref-IDT". On the other hand, no homogenization is actually needed in IDT, thanks to the possibility to represent the heterogeneous geometry in every detail. The numerical results provided by "Ref-IDT" and "Het-IDT-4" in Fig. 8.6 prove that the transport model of IDT can eliminate the rod-cusping effect.

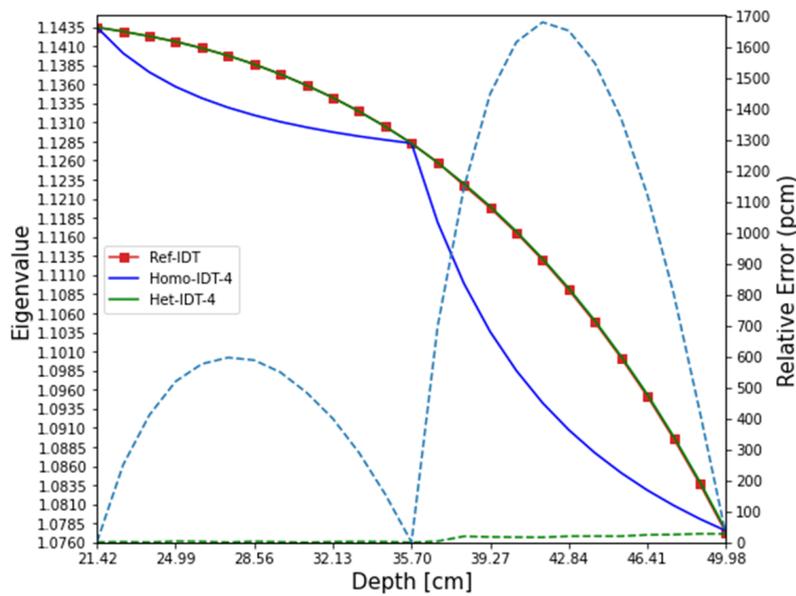


Figure 8.6: Control-rod simulation in the C5G7 benchmark, using nodes with $\Delta h = 14.28 \text{ cm}$: k_{eff} vs control-rod position (solid lines) and relative error with respect to "Ref-IDT" (dashed lines). N.B. The control rods are gradually inserted with a step size of 1.19 cm .

8.6 . CONCLUSION

The new geometric capabilities of IDT enable high-fidelity representation of a fission reactor, at reasonable computational cost. In fact, thanks to the possibility to locally coarsen or refine the volume mesh, one can retain a fine grid only where strictly necessary, thus saving simulation time and memory. The new HCC model can then be effectively applied to control-rod movement simulations, in order to avoid cusping effects. As discussed in Sect. 8.5, the curve representing the control-rod reactivity worth shows no discontinuity in the first derivative, allowing for an accurate calculation of the effective multiplication factor for each position of the control rods.

Part III

NOVELTIES ON THE METHOD OF SHORT CHARACTERISTICS

9 - ASYMPTOTIC EXPANSION OF THE ANGULAR FLUX APPLIED TO DISCRETE-ORDINATE SOURCE ITERATIONS

9.1 . INTRODUCTION

Lattice simulations of nuclear reactors involve a large number of media and thus of effective total macroscopic cross sections, due to flux variations. The parameters affecting the cross sections are numerous and include

- the temperature feedback, for the spatial variation of fuel and moderator cross-sections by Doppler effect and mass density variations;
- the self-shielding, for the spatial variations of the multigroup cross sections, due to the resonances of fuel isotope microscopic cross sections;
- the fuel depletion, for the spatial distribution of the isotopic concentrations of the fuel rods.

In this context, numerical methods based on the integral form of the transport equation, such as Collision Probabilities (CP) or Current-Coupling Collision Probabilities (CCCP) (Sect. 4.6.2), suffer from severe memory burden as the number of media and regions increases [87, 88, 89, 44, 71]. The reason lies in the computation and storage of a high number of collision probability matrices, [90]. These matrices have to be stored for any energy group and, in the framework of IDT, for any direction of the discrete-ordinate angular quadrature (Sect. 4.6.4 and Sect. 6.3.1). In this context, when the number of depleted regions grows up to thousands, the computational cost of the coefficients becomes not negligible and the solver suffers from great memory pressure during runtime [91, 92].

In order to ease such undesirable effect, let us propose an asymptotic technique that can lead to an effective reduction of the memory imprint associated to the discretized CP matrices, at the cost of an increased number of applications of the transport sweeps. The sweeps allow to compute the spatial distribution of the angular flux for a given directional source and incoming boundary flux. The present method is applied to Heterogeneous Cartesian Cells (HCCs) to solve the neutral-particle transport equation by discrete ordinates. In particular, the approach relies on a Neumann series using an "*unperturbed*" transport operator. The latter is constructed on-the-fly and defined by grouping HCCs (i.e., pin cells) having *similar* optical path. For these cells, the solver computes and stores a single set of CP matrices, thus mitigating the memory requirements.

In this chapter, Sect. 9.2 and 9.3 provide the mathematical basis necessary for the new method and details on its practical application and implementation in IDT. In Sect. 9.4, two realistic lattice simulations are presented to evaluate the numerical accuracy and performance of the asymptotic expansion. Preliminary results show reductions of the memory burden by one order of magnitude, without any significant loss in the numerical precision, at the expense of an increased computational cost of the source iterations. A slight optimization allows to limit the latter to +10 – 20% with respect to to the standard calculation.

9.2 . THEORY

Let us consider the one-group discrete-ordinate transport problem

$$\begin{cases} \mathbf{L}\psi = q, & \text{for } \mathbf{r} \in D, \\ \psi_- = \psi_{in}, & \text{for } \mathbf{r}_- \in \partial D_-. \end{cases} \quad (9.1)$$

In Eq. (9.1), the energy group index is omitted for the sake of simplicity and

- $\psi = \psi(\mathbf{r}, \boldsymbol{\Omega}_d)$ is the angular flux for a fixed angular direction $\boldsymbol{\Omega}_d$,
- $\mathbf{L} \equiv \boldsymbol{\Omega}_d \cdot \nabla + \Sigma(\mathbf{r})$ is the one-group leakage-plus-removal operator,
- $q = q(\mathbf{r}, \boldsymbol{\Omega}_d)$ is a fixed source, which is given at each transport iteration by an update of the self-scattering and of the external source contributions (multigroup scattering plus fission source),
- $\psi_{in} = \psi_{in}(\mathbf{r}_-, \boldsymbol{\Omega}_d)$ is a fixed incoming flux on the incoming boundary of the domain, $\partial D_- = \{\mathbf{r} \in \partial D \text{ such that } \mathbf{n}(\mathbf{r}) \cdot \boldsymbol{\Omega}_d < 0\}$.

Equation (9.1) represents a first-collision source problem, which is typically solved through spatial sweeps. The numerical solution of Eq. (9.1) gives the distribution of the angular flux in space for a given volume source, q , and boundary condition, ψ_{in} . Note that the total cross section, $\Sigma(\mathbf{r})$, is the only parameter ruling the operator \mathbf{L} in Eq. (9.1).

Supported by asymptotic theory, [94], the total cross section of Eq. (9.1) can be written as

$$\Sigma(\mathbf{r}) = \Sigma_0(\mathbf{r}) - \delta\Sigma(\mathbf{r}), \quad (9.2)$$

where $\Sigma_0(\mathbf{r})$ is a smooth function representing an artificial "unperturbed" total cross section, while $\delta\Sigma(\mathbf{r})$ is a "perturbation" function. Consequently, the operator \mathbf{L} reads as

$$\mathbf{L} = \mathbf{L}_0 - \delta\Sigma, \quad (9.3)$$

with $\mathbf{L}_0 \equiv \boldsymbol{\Omega}_d \cdot \nabla + \Sigma_0(\mathbf{r})$ as the "unperturbed" operator. Then, Eq. (9.1) becomes

$$\begin{cases} (\mathbf{L}_0 - \delta\Sigma)\psi = q, & \text{for } \mathbf{r} \in D, \\ \psi_- = \psi_{in}, & \text{for } \mathbf{r}_- \in \partial D_-, \end{cases} \quad (9.4)$$

where $\delta\Sigma(\mathbf{r}) = \Sigma_0(\mathbf{r}) - \Sigma(\mathbf{r})$. Note that q and ψ_{in} are unchanged with respect to the original transport problem (Eq. (9.1)). Assuming that \mathbf{L}_0^{-1} exists, Eq. (9.4) can be rearranged as

$$(\mathbf{I} - \mathbf{L}_0^{-1}\delta\Sigma)\psi = \psi_0, \quad (9.5)$$

$\psi_0 = \mathbf{L}_0^{-1}q$ being "unperturbed" flux solving the "unperturbed" problem

$$\mathbf{L}_0\psi_0 = q. \quad (9.6)$$

Since the choice of $\Sigma_0(\mathbf{r})$ is arbitrary, let us set the following criteria

$$\Sigma_0(\mathbf{r}) \geq 0 \text{ for any } \mathbf{r} \in D, \quad (9.7)$$

$$\delta\Sigma(\mathbf{r}) \geq 0 \text{ for any } \mathbf{r} \in D. \quad (9.8)$$

The first is a physical condition, while the second implies that the "unperturbed" cross section is a bounding function for the original cross section, i.e., $\Sigma_0(\mathbf{r}) \geq \Sigma(\mathbf{r})$ over D . If $\Sigma_0(\mathbf{r})$ is such that $\delta\Sigma$ is sufficiently small with respect to $\Sigma_0(\mathbf{r})$, the spectral radius of the linear operator $\mathbf{L}_0^{-1}\delta\Sigma$, here indicated as ρ , is bounded by

$$\rho = \|\mathbf{L}_0^{-1}\delta\Sigma\| < 1. \quad (9.9)$$

In this case, the Neumann series of $\mathbf{L}_0^{-1}\delta\Sigma$ converges in the operator norm to the inverse of $(\mathbf{I} - \mathbf{L}_0^{-1}\delta\Sigma)$. The following identity holds

$$(\mathbf{I} - \mathbf{L}_0^{-1}\delta\Sigma)^{-1} = \sum_{n \geq 0} (\mathbf{L}_0^{-1}\delta\Sigma)^n.$$

The solution of Eq. (9.5) is thus given by Neumann series

$$\psi = \psi_0 + \sum_{n \geq 1} (\psi_n - \psi_{n-1}) = \sum_{n \geq 0} (\mathbf{L}_0^{-1}\delta\Sigma)^n \psi_0, \quad (9.10)$$

where ψ_n , with $n > 0$, are high-order successive approximations of ψ .

9.3 . APPLICATION TO IDT

As mentioned in Chap. 7, the new version of IDT uses Heterogeneous Cartesian Cells (HCCs), containing unstructured combinations of concentric cylinders and inner Cartesian grids, whose intersections define the HCC spatial mesh. Also the boundary surfaces of the HCC box are discretized and a piece-wise polynomial expansion is used to represent the interface angular flux. The numerical solution of Eq. (9.1) within the HCC requires the computation for each energy group of a set of angular-dependent CP matrices (Sect. 4.6.4). Assuming

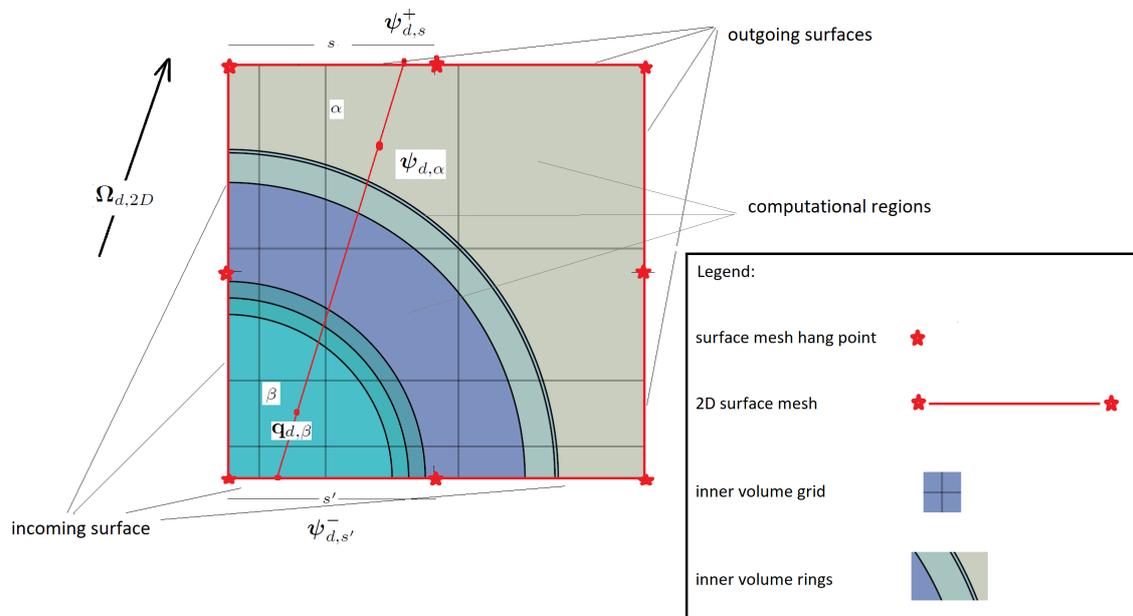


Figure 9.1: Unstructured non-conformal 3D Heterogeneous Cartesian Cell mesh.

- d as the index of direction Ω_d , with weight w_d ,
- $s \in \mathcal{S}_d^+$ and $s' \in \mathcal{S}_d^-$ as the outgoing and incoming surface indexes with respect to d , which belong to the set of outgoing/incoming interface surfaces $\mathcal{S}_d^{+/-}$ on the border of the HCC,
- $\alpha, \beta, \gamma, \dots \in \mathcal{G}$ as the inner volume indexes within the HCC,
- (i) as the source iteration index (Sect. 4.47),
- $\psi_{d,s}^\pm$ as the array containing the outgoing/incoming surface spatial moments of the angular interface flux,
- $\phi_{\alpha,l,m}$ as the array of the spatial moments pertaining the angular moment $\Phi_{l,m}$, in region α ,
- $\mathbf{q}_{d,\alpha}$ as the array of the spatial moments of the angular source, in region α , along direction Ω_d ,
- $\mathbf{s}_{\alpha,l,m}^{ext}$ as the array of the spatial moments of the external source,
- $\Sigma_{s,\alpha,l}$ as the angular moment of order l of the self-scattering cross section,
- $A_{l,m}(\Omega_d)$ as a real-valued spherical harmonic,

the transport sweep of the HCC, along source iterations, is performed applying the CP matrices, namely $\mathbf{T}_{d,s,s'}$, $\mathbf{E}_{d,s,\alpha}$, $\mathbf{I}_{d,\alpha,s'}$, $\mathbf{C}_{d,\alpha,\beta}$, as follows

$$\psi_{d,s}^{+,(i+1)} = \sum_{s' \in \mathcal{S}_d^-} \mathbf{T}_{d,s,s'} \psi_{s',d}^{-,(i+1)} + \sum_{\alpha \in \mathcal{G}} \mathbf{E}_{d,s,\alpha} \mathbf{q}_{d,\alpha}^{(i)}, \text{ for } s \in \mathcal{S}_d^+ \quad (9.11)$$

$$\phi_{\alpha,l,m}^{(i+1)} = \sum_{d \in \mathcal{S}_N} w_d A_{l,m}(\Omega_d) \left[\sum_{s' \in \mathcal{S}_d^-} \mathbf{I}_{d,\alpha,s'} \psi_{s',d}^{-,(i+1)} + \sum_{\beta \in \mathcal{G}} \mathbf{C}_{d,\alpha,\beta} \mathbf{q}_{d,\beta}^{(i)} \right], \text{ for } \alpha \in \mathcal{G} \quad (9.12)$$

$$\mathbf{q}_{d,\alpha}^{(i+1)} = \sum_{k=0}^K \sum_{l=-k}^k A_{l,m}(\Omega_d) \left[\Sigma_{s,\alpha,k} \phi_{\alpha,l,m}^{(i+1)} + \mathbf{s}_{\alpha,l,m}^{ext} \right], \text{ for } \alpha \in \mathcal{G}. \quad (9.13)$$

Equation (9.11) transmits particle contributions through the HCC surfaces, using the transmission matrix, $\mathbf{T}_{d,s,s'}$, and the escape matrix, $\mathbf{E}_{d,s,\alpha}$. Equation (9.12) expresses the particle conservation in $\alpha \in \mathcal{G}$, where the incoming matrix, $\mathbf{I}_{d,\alpha,s'}$, and the collision matrix, $\mathbf{C}_{d,\alpha,\beta}$, are related to the probability for a neutron entering surface $s' \in \mathcal{S}_d^-$ and for a neutron born in region $\beta \in \mathcal{G}$ to contribute to the region of balance. Finally, Eq. 9.13 allows to update the neutron source.

The geometrical module of IDT stores HCC's data for each box $\mathbf{c} = \{i_x, i_y, i_z\}$, with $i_x/y/z$ as the index of the Cartesian mesh hosting the HCCs. This process associates automatically the local geometry of the box, \mathcal{G}_c , to the set of total cross sections, for each region, namely the array

$$\Sigma_{\mathbf{c}} = \{\Sigma_{\alpha}\}_{\alpha \in \mathcal{G}_c}.$$

Provided a perturbation

$$\|\delta\Sigma\| \ll \|\mathbf{L}_0^{-1}\|,$$

let us suppose that the many-media problem, characterized by \mathbf{L} , is a perturbed state of a few-media problem \mathbf{L}_0 , with "*unperturbed*" media, which requires much less of coefficients to compute. The problem is how to construct the operator \mathbf{L}_0 from the original operator, \mathbf{L} .

In the HCC framework, \mathbf{L}_0 can satisfy Eqs. (9.7) and (9.8) by selecting those HCCs that share the same geometrical pattern and by choosing a single set of cross sections for those cells that have similar optical properties. Let us define \mathcal{G}_{c_0} as the cell geometry, which is assumed to be shared by several HCCs, and Σ_{c_0} as the unperturbed set of total cross sections associated to each region of \mathcal{G}_{c_0} , which has to be determined. The couple $(\mathcal{G}_{c_0}, \Sigma_{c_0})$ defines data for a single *unperturbed* HCC, with identifier c_0 . IDT automatically orders physical HCCs that share the same generating geometry in a proper subset of cell indexes. Let us denote this subset by $\mathcal{H}(\mathcal{G}_{c_0})$, whose HCCs share geometry \mathcal{G}_{c_0} .

Once the HCCs are ordered by their modular geometrical pattern, Σ_{c_0} is defined by selecting in $\mathcal{H}(\mathcal{G}_{c_0})$ those cells that satisfy the following *error-control requirement*

$$\begin{aligned} \left| \Sigma_{\mathbf{c},\alpha}^g - \Sigma_{\mathbf{c}',\alpha}^g \right| &< \epsilon_\Sigma \min(\Sigma_{\mathbf{c},\alpha}^g, \Sigma_{\mathbf{c}',\alpha}^g) + \text{EPSILON}(\Sigma), \\ \alpha &\in \mathcal{G}_{c_0} \text{ and } \mathbf{c} \neq \mathbf{c}' \text{ with } \mathbf{c}, \mathbf{c}' \in \mathcal{H}(\mathcal{G}_{c_0}), \\ &\text{and for any energy group } g, \end{aligned} \quad (9.14)$$

that bounds the norm of $\delta\Sigma(\mathbf{r})$ in L^∞ . $\text{EPSILON}(\Sigma)$ is the machine epsilon to evaluate round-off error. To this end, a tolerance parameter on the cross sections, ϵ_Σ , is introduced in Eq. (9.14). The latter is fixed by the user and helps to select cells in $\mathcal{H}(\mathcal{G}_{c_0})$ that have similar optical thickness within a tolerance of ϵ_Σ .

Let us define the set of selected indexes satisfying Eq. (9.14) as $\mathcal{H}(\mathcal{G}_{c_0}, \Sigma_{c_0})$. Of course, this set belongs to $\mathcal{H}(\mathcal{G}_{c_0})$. The unperturbed set of cross sections associated to $\mathcal{H}(\mathcal{G}_{c_0}, \Sigma_{c_0})$ is computed as

$$\begin{aligned} \Sigma_{c_0,\alpha}^g &= \max_{\mathbf{c} \in \mathcal{H}(\mathcal{G}_{c_0}, \Sigma_{c_0})} \{ \Sigma_{\mathbf{c},\alpha}^g \}, \\ &\text{for any } \alpha \in \mathcal{G}_{c_0} \text{ and for any energy group } g. \end{aligned} \quad (9.15)$$

The perturbation $\delta\Sigma_{\mathbf{c},\alpha}^g$ is then automatically defined by

$$\begin{aligned} \delta\Sigma_{\mathbf{c},\alpha}^g &= \Sigma_{\mathbf{c},\alpha}^g - \Sigma_{c_0,\alpha}^g, \text{ for any } \mathbf{c} \in \mathcal{H}(\mathcal{G}_{c_0}, \Sigma_{c_0}), \\ \alpha &\in \mathcal{G}_{c_0}, \text{ and for any energy group } g. \end{aligned} \quad (9.16)$$

Thanks to Eq. (9.15), $\Sigma_{c_0,\alpha}^g$ satisfies Eqs. (9.7) and (9.8), while Eq. (9.14) bounds the norm of the operator $\mathbf{L}_0^{-1}\delta\Sigma$. Considering in first approximation the leakage contribution of second order, the spectral radius is bounded by

$$\rho \sim \max_{\mathbf{c} \in \mathcal{H}(\mathcal{G}_{c_0}, \Sigma_{c_0})} \left\{ \frac{\delta\Sigma_{\mathbf{c},\alpha}^g}{\Sigma_{c_0,\alpha}^g} \right\} < \frac{\epsilon_\Sigma \min(\Sigma_{\mathbf{c},\alpha}^g)}{\Sigma_{c_0,\alpha}^g} < \epsilon_\Sigma. \quad (9.17)$$

More details are provided in App. C. Finally, the couple $(\mathcal{G}_{c_0}, \Sigma_{c_0})$, i.e., the shared geometry and the unperturbed cross sections (Eq. (9.15)), define the single *unperturbed HCC*. The latter is the artificial cell representing the unperturbed state of all HCCs associated to index $\mathbf{c} \in \mathcal{H}(\mathcal{G}_{c_0}, \Sigma_{c_0})$.

As known, CP matrices in Eqs. (9.11) and (9.12) are the discrete form of the operator \mathbf{L}^{-1} restricted to a single HCC. Moreover, as for \mathbf{L} , the parameters of the CPs matrices are the piece-wise total cross sections inside the HCC. Thus, \mathbf{L}_0^{-1} is locally defined by associating to the cells in $\mathcal{H}(\mathcal{G}_{\mathbf{c}_0}, \boldsymbol{\Sigma}_{\mathbf{c}_0})$ a single set of unperturbed matrices,

$$\begin{aligned} \mathbf{L}_0^{-1}(\mathcal{G}_{\mathbf{c}_0}, \boldsymbol{\Sigma}_{\mathbf{c}_0}) &\equiv \{ \mathbf{T}_{d,s,s'}, \mathbf{E}_{d,s,\alpha}, \mathbf{I}_{d,\alpha,s'}, \mathbf{C}_{d,\alpha,\beta} \}(\mathcal{G}_{\mathbf{c}_0}, \boldsymbol{\Sigma}_{\mathbf{c}_0}), \\ &\text{for any } s' \in \mathcal{S}_d^-, s \in \mathcal{S}_d^+, \alpha, \beta \in \mathcal{N}_{\mathbf{c}} \text{ and } d \in \mathcal{S}_N, \\ &\text{and } \mathbf{c} \in \mathcal{H}(\mathcal{G}_{\mathbf{c}_0}, \boldsymbol{\Sigma}_{\mathbf{c}_0}). \end{aligned} \quad (9.18)$$

In this manner, the original set of HCCs, that would require the computation of CP matrices for all the elements in $\mathcal{H}(\mathcal{G}_{\mathbf{c}_0})$, is decomposed in sets $\mathcal{H}(\mathcal{G}_{\mathbf{c}_0}, \boldsymbol{\Sigma}_{\mathbf{c}_0})$, whose elements refer to a single unperturbed operator $\mathbf{L}_0^{-1}(\mathcal{G}_{\mathbf{c}_0}, \boldsymbol{\Sigma}_{\mathbf{c}_0})$.

As mentioned, the construction of sets $\mathcal{H}(\mathcal{G}_{\mathbf{c}_0})$ and $\mathcal{H}(\mathcal{G}_{\mathbf{c}_0}, \boldsymbol{\Sigma}_{\mathbf{c}_0})$ is done automatically in IDT. In lattice applications, geometry $\mathcal{G}_{\mathbf{c}_0}$ often coincides with the one of a pin-cell. Also, the fuel rods and the control rods are often of few kinds. For these reasons, one can expect that elements composing $\mathcal{H}(\mathcal{G}_{\mathbf{c}_0}, \boldsymbol{\Sigma}_{\mathbf{c}_0})$ roughly coincide with pin-cells of the same type. The parameter ϵ_{Σ} permits to calibrate the size of $\mathcal{H}(\mathcal{G}_{\mathbf{c}_0}, \boldsymbol{\Sigma}_{\mathbf{c}_0})$. The allowed range of values is $[0, 1[$. In this way, one can distinguish two possible behaviors of the solver during runtime, more precisely

$$\begin{aligned} \epsilon_{\Sigma} = 0 &\rightarrow \text{to foster memory,} \\ \epsilon_{\Sigma} < 1 &\rightarrow \text{to foster computation.} \end{aligned}$$

The first selects HCCs having exactly the same geometry and cross sections. This is the way *standard IDT* works: there is no asymptotic approximation, but just an optimization for coefficient burden. Thus, identical cells that share exactly the same optical properties generate a single set of coefficients. Because $\epsilon_{\Sigma} = 0$ in Eq. (9.14), the solver performs the transport sweep inverting \mathbf{L} , i.e., the original operator, instead of \mathbf{L}_0 , i.e., the unperturbed operator.

The second setting allows for a variation of the total cross sections up to 100%, i.e., with $\epsilon_{\Sigma} = 1$. In this case, HCCs that belong to $\mathcal{H}(\mathcal{G}_{\mathbf{c}_0}, \boldsymbol{\Sigma}_{\mathbf{c}_0})$ grow in number as the tolerance increases. Therefore, the memory burden as well as the computational cost of CP coefficients is reduced proportionally. Provided $\epsilon_{\Sigma} < 1$, *asymptotic IDT* performs transport sweep to invert \mathbf{L}_0 , by using the set of coefficients in Eq. (9.18). The operator \mathbf{L}_0^{-1} is then applied iteratively (Eq. (9.10)) to build successive approximations of ψ , until convergence. Thus, *asymptotic IDT* lessens the memory pressure while fosters computation. The latter rises because of the increase of the computational cost of source iterations due to Neumann iterations.

9.3.1 . Implementation in IDT

In IDT, Eq. (9.10) is implemented for each energy group and consists in performing an inner loop where successive iterates are progressively higher order approximations of ψ . As mentioned, the transport sweep is performed by using the operator \mathbf{L}_0 . Thus, \mathbf{L}_0^{-1} symbolically represents the inner transport sweep with unperturbed CP matrices. The numerical scheme is summarized in Algo. 2. Note that N_R and N_c are respectively the total number of regions (line 2, Algo. 2) and the number of spatial moments (line 10, Algo. 2), while ϵ_{ψ} is a user-defined tolerance (line 10, Algo. 2). In Algo. 2, Neumann iterations are not performed if the $L_2 - norm$ of the perturbation is less than 1 *pcm* (line 2).

Algorithm 2 : Von Neumann iterations.

```

1: Initialize  $\psi_0 = \mathbf{L}_0^{-1}q$ , for  $\mathbf{r} \in D$ , and  $\psi_{0,-} = \psi_{in}$ , for  $\mathbf{r}_- \in \partial D_-$ 
2: if  $\|\delta\Sigma\|_{L_2} < 10^{-5} + N_R * \text{EPSILON}(\Sigma)$  then
3:   RETURN
4: end if
5: Update  $q_0 \leftarrow q_0 + \delta\Sigma \cdot \psi_0$ 
6: for  $n = 1 \rightarrow N$  {Neumann iterations} do
7:   Determine  $\psi_n = \mathbf{L}_0^{-1}q_{n-1}$ , for  $\mathbf{r} \in D$ 
8:   Determine  $\psi_{n,-} = \psi_{in}$ , for  $\mathbf{r}_- \in \partial D_-$ 
9:   Compute  $\delta\psi = \psi_n - \psi_{n-1}$  {Residual}
10:  if  $\|\delta\psi\|_{L_2} < \epsilon_\psi + N_R * N_C * \text{EPSILON}(\psi)$  then
11:    EXIT{Neumann series expansion truncated at order  $n$ }
12:  end if
13:  Update  $q_n \leftarrow q_{n-1} + \delta\Sigma \cdot \delta\psi$ 
14: end for

```

S_n quadrature	$S_{16,8}$ CL (256 dir.)
Source expansion	Linear
Interface surface flux	Linear
Nb. of groups	281
Anisotropy order	P_3
Nb. of surfaces/HCC-side	3
Inner iteration tol. (L^∞ -norm)	10^{-5}
Reactivity tol. on k -eff.	10^{-5}
Fission source tol. (L^∞ -norm)	10^{-4}
ϵ_Σ	0.1 (= 10%)
ϵ_ψ (L^2 -norm)	10^{-5}

Table 9.1: Lattice problem settings.

Asymptotic IDT can be adjusted to users' scope, by two input parameters. The first is the aforementioned tolerance on total cross sections, i.e., ϵ_Σ , while the second is the tolerance at line 10, namely ϵ_ψ , on the iterative residual norm. While the first one affects CP-matrices memory requirements and the perturbation norm, the second influences the accuracy of ψ and thus the number of Neumann iterations. Therefore, the number of applications of lines 7-13 (Algo. 2) increases with ϵ_Σ , while reduces by increasing ϵ_ψ .

9.4 . NUMERICAL TESTS

In the following tests, the asymptotic model is applied to realistic lattice calculations on 2D PWR assemblies. The problem discretization is shown in Tab. 9.1. Note that all simulations are performed with multigroup CMFD in outer iterations and without inner accelerations, using 4 threads. The workstation is an Intel Xeon Silver 4214 @ 2.20 GHz \approx 16.5 M Cache.

9.4.1 . Application to Self-Shielded Lattice

The present simulation is aimed at testing the asymptotic method in realistic lattice calculations with self-shielded cross-sections. Identical fuel rods may have different nuclear data, due to the spatial variation of the multigroup microscopic cross sections, induced by the presence of resonant isotopes.

The benchmark is a 2D assembly, with UOX fuel and B₄C control rods (Fig. 9.2). We impose reflective boundary conditions. Self-shielding (SSH) is performed using the Fine-Structure (FS) model, between groups 53 and 98, for 12 resonant isotopes. SSH calculation consists of 8 distinct cells (of which 6 UOX pin-cells, 1 cell for the control rod and 1 cell for the instrument tube, respectively) solved by Current-Coupled Collision Probabilities, with 1/8th symmetry.

IDT is compared to TRIPOLI-4® continuous-energy Monte Carlo. The problem is discretized by 1349 spatial regions. The other discretization parameters are listed in Tab. 9.1, which also provides detailed information on the problem settings. Standard IDT produces a reactivity error of 53 pcm. The relative error on the absorption and fission distributions is less than 1%, as depicted in Fig. 9.3.

Now let us assume standard IDT as a reference and provide detailed comparison with the asymptotic solution (Tab. 9.2). Due to self-shielding, 8 sets of HCC coefficients are necessary for standard IDT to run. On the other hand, if one applies the asymptotic expansion, with a cross section tolerance of $\epsilon_{\Sigma} = 10\%$, the number of HCCs is reduced from 8 to 3, as all UOX pin-cells are represented by a single *unperturbed* HCC. Asymptotic IDT allows to reduce the

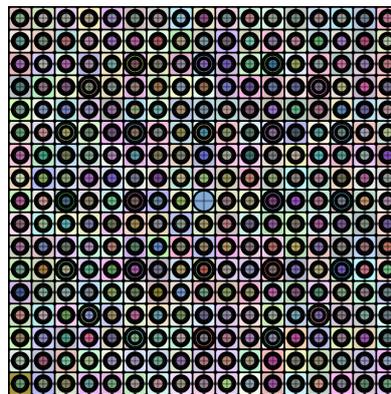


Figure 9.2: 2D UOX/B₄C benchmark.

memory imprint by a factor 8. Also, the percent of time spent in CP computation is reduced from 31% to less than 1%. Nevertheless the time to solution increases by a factor 1.75, because of the repeated application of the unperturbed transport operator. The transport operator is applied almost twice the number of times of the reference solution. This means that a first-order moment, namely

$$\psi_1 = \sum_{n=0}^1 (\mathbf{L}_0^{-1} \delta \Sigma)^n \mathbf{L}_0^{-1} q, \quad (9.19)$$

is necessary, on average, in each energy group.

To test the accuracy of the asymptotic method, let us verify the pin-by-pin relative difference between standard and asymptotic IDT. Fig. 9.4 shows the maximum difference along each row

IDT	Memory	Time	Time for CP	Time to solution	k_{eff} -error
Standard	610 MB	105 s	33 s (31%)	72 s (69%)	-
Asymptotic	72 MB	127 s	1 s (<1%)	126 s (99%)	0.5 pcm

Table 9.2: Standard (reference) vs Asymptotic IDT, performed by LSC and $S_{48,8}$ -Chebyshev-Legendre product quadrature.

of the assembly. The method provides accurate results since errors on fission and absorption distributions remain bounded to less than 0.013% in simple precision. Tab. 9.2 contains the total and partial times of the simulation, as well as the reactivity difference.

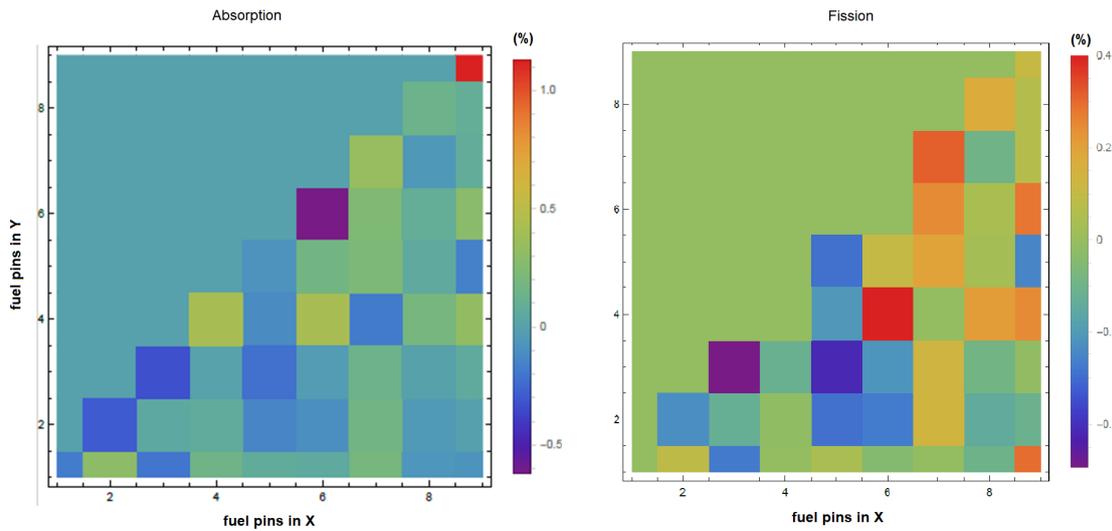


Figure 9.3: TRIPOLI4® vs Asymptotic IDT: relative error on the pin-by-pin absorption and fission distributions.

9.4.2 . Application to depleted lattice

The second benchmark is a 2D assembly, with 3.1%-enriched UOX fuel and fixed Pyrex rods. Fuel depletion is simulated from 0 to 70 $GWd/tonU$, using 50 burn-up steps. The discretization consists of 1473 spatial regions, with $S_{16,8}$ Chebyshev-Legendre angular quadrature formula and 281-group P_3 cross-section library. The UOX and Pyrex rods are depleted preserving a symmetry of $1/8^{th}$ in material distribution. A total of 51 HCCs are necessary to perform the standard calculation. These include 45 depleted fuel cells, 5 Pyrex cells and 1 cell containing the instrumentation. Self-shielding is based on FS model, with 12 self-shielded cells.

In order to set the asymptotic solver, let us investigate the variation of the total cross sections of all cells along depletion. Fig. 9.5 displays the maximum variation per energy group. As this does not exceed 8%, let us set $\epsilon_{\Sigma} = 10\%$. In this way, only two *unperturbed* cells are used for depleted UOX and Pyrex cells, respectively, allowing for an overall reduction of the number of HCCs from 51 to 6. In Fig. 9.6, the number of HCCs and the memory occupation for the angular probability coefficients is depicted along burn-up. A reduction by a factor 10 is observed for the memory burden with respect to standard IDT. This reduction is further visible by looking at the computational time for coefficients, displayed in Fig. 9.7.

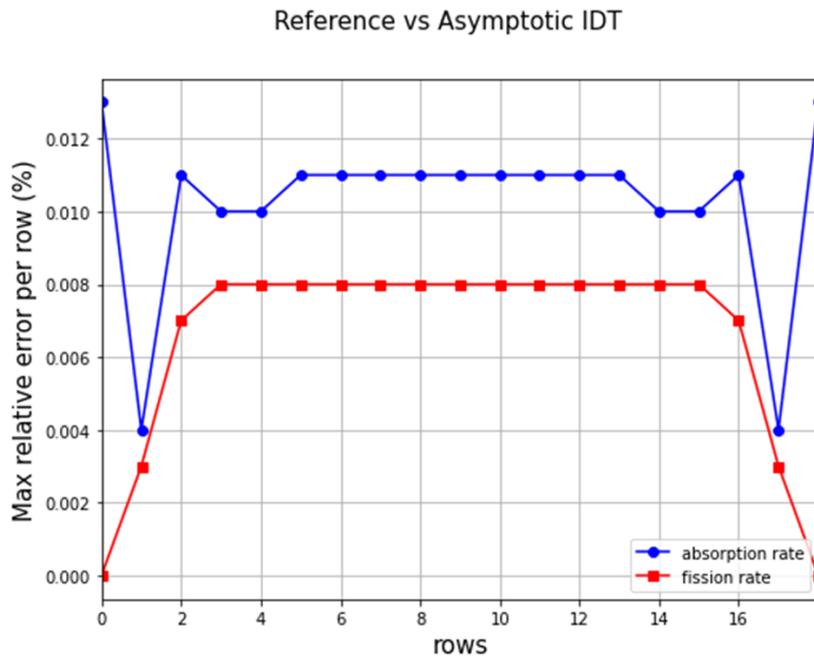


Figure 9.4: Standard vs Asymptotic IDT: maximum relative error per row on the pin-by-pin fission and absorption distributions.

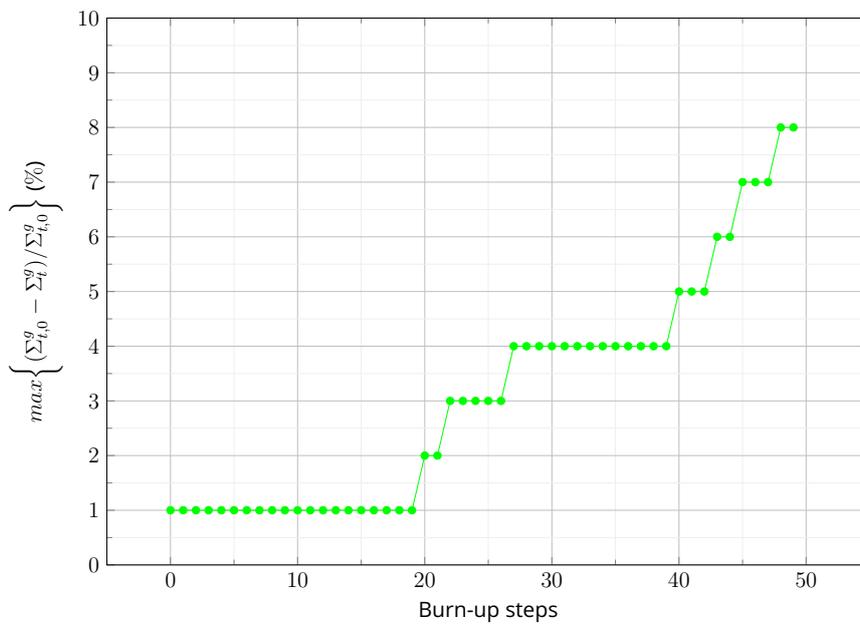


Figure 9.5: Maximum total-cross-section variation (%) vs burn-up.

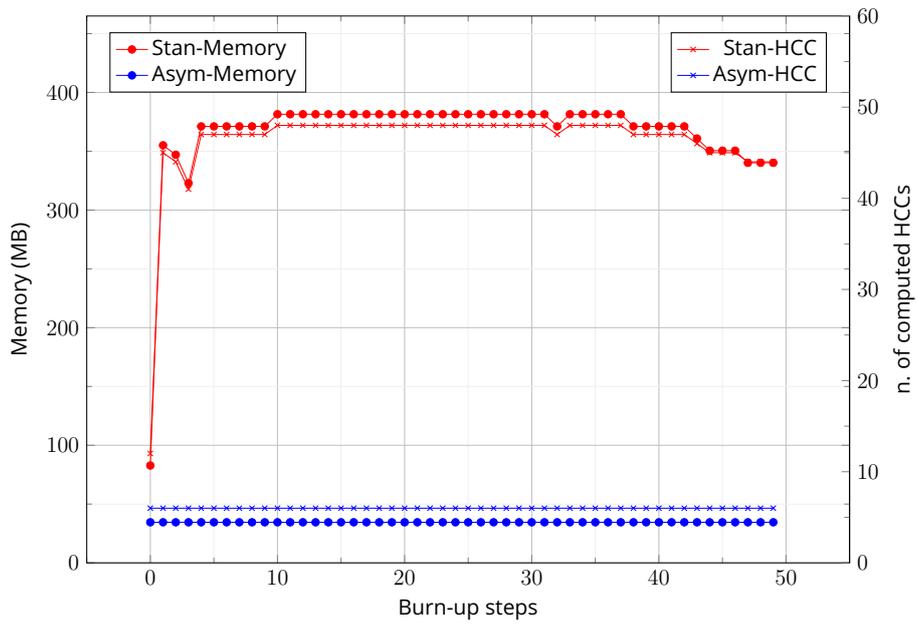


Figure 9.6: Memory for coefficients vs burn-up.

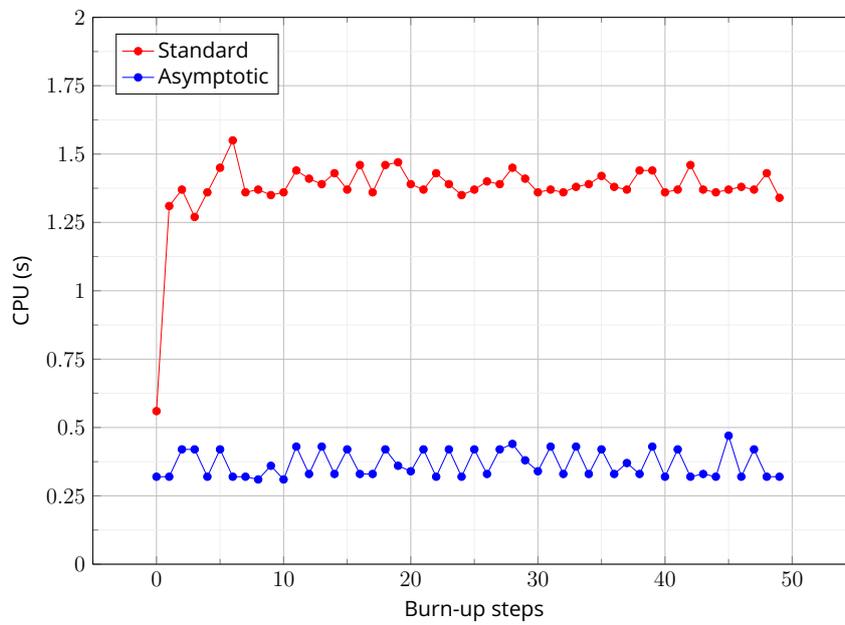


Figure 9.7: Time for HCC coefficients vs burn-up.

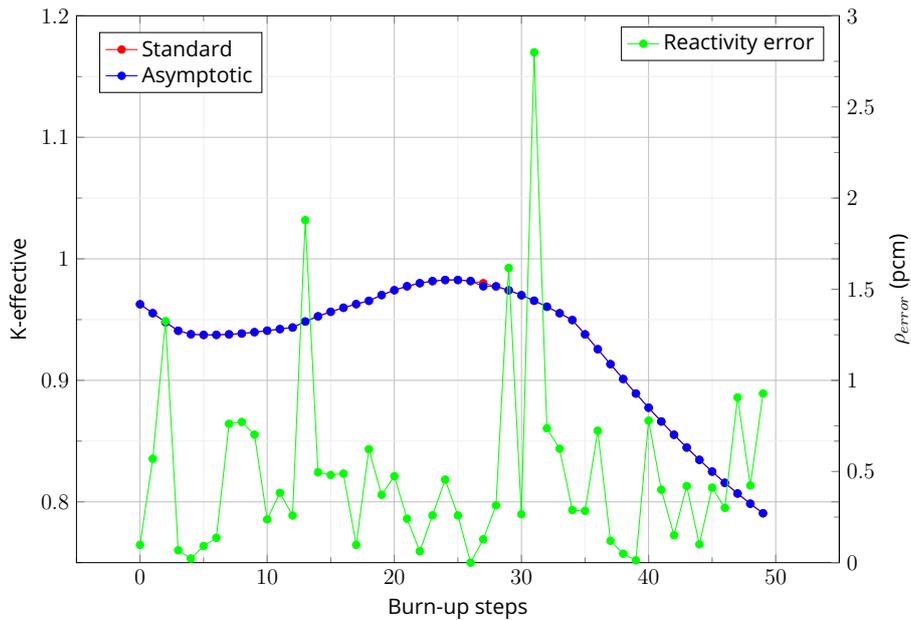


Figure 9.8: k_{eff} (blue and red curves) and reactivity error (green curve) vs burn-up.

Figure 9.8 shows the value of k_{eff} and the reactivity error along depletion. The numerical accuracy is acceptable, as the latter remains bounded to few pcm for the whole cycle. As expected, the asymptotic method pays an extra cost in terms of iterative solution, due to the overhead of the inner Neumann loop that is added at each source iteration. This is visible in Fig. 9.9, where the total computational time of outer iterations and inner transport sweeps is displayed along burn-up. At BOC, the time of the inner transport sweeps of the asymptotic method doubles the time of the standard calculation and, as soon as fuel depletes and concentrations change, it triples the time of standard IDT. This behavior is not surprising, since materials differ because of self-shielding at BOC and because of both self-shielding and concentrations along the cycle. As in Sect. 9.4.1, the first-order moment is sufficient to represent self-shielded cells at BOC. When fuel and Pyrex compositions change due to depletion, the second-order moment is also necessary to reconstruct the flux. To reduce the computational cost, a simple optimization is introduced, where Neumann iterations are activated only once k_{eff} has reached convergence. In IDT, three stopping criteria have to be satisfied to exit iterations, involving

- the L_{∞} -error on the angular moments of the flux;
- the L_{∞} -error on the fission production rate (per fissile isotope);
- The reactivity error on k_{eff} .

As the latter is the first to stabilize, the flux is approximated to the "unperturbed" moment, until convergence of the eigenvalue. Once reached this condition, Neumann iterations are activated to converge the flux moments and the fission sources. Fig. 9.10 shows the computational time of the asymptotic method with the new optimization. The overhead due to Neumann iterations is reduced to 10 – 20% along the whole cycle. More investigation has to be done to tune the CMFD acceleration consistently with the new iterative strategy.

9.5 . CONCLUSION

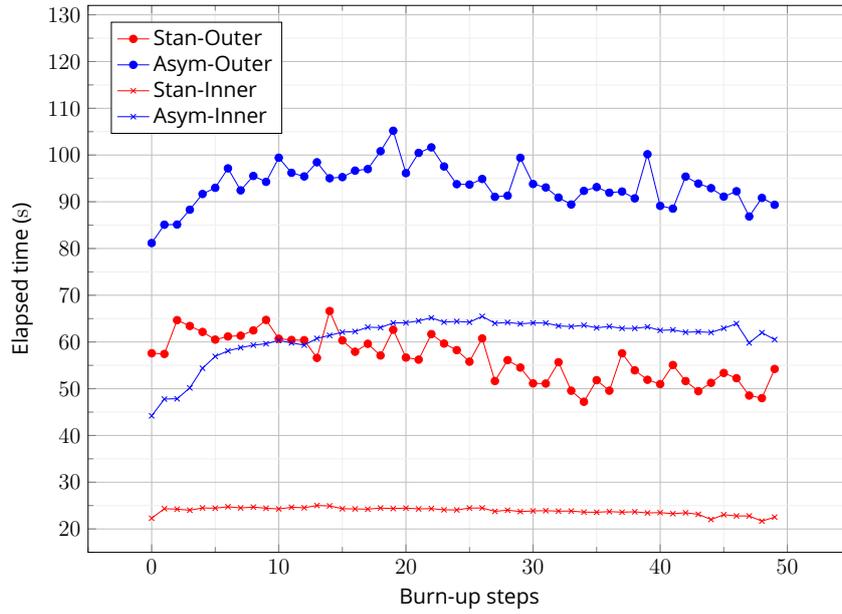


Figure 9.9: Time for inner and outer iterations vs burn-up.

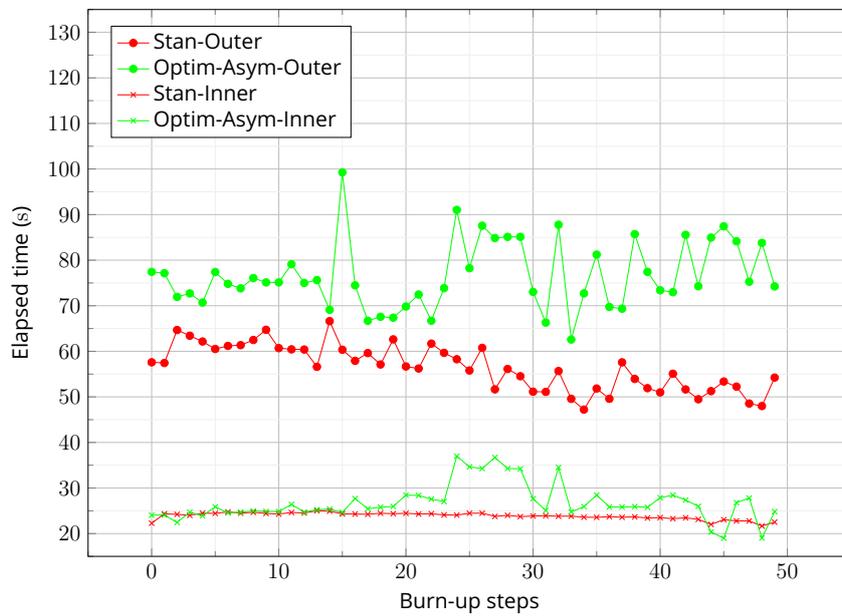


Figure 9.10: Time for inner and outer iterations vs burn-up (asymptotic IDT is optimized by including high-order moments only once k_{eff} has reached convergence).

In this chapter, an asymptotic flux expansion has been presented, to effectively reduce the memory burden of CP matrices, in IDT. This novel technique transfers the computational cost of the simulation from coefficients to inner transport iterations that are performed more than once for each angle. These iterations compute the '*perturbed*' moments of the flux that allow to reconstruct the solution up to adaptive order of expansion. Despite an increasing cost for inner iterations of about the 10 – 20% in fuel-assembly simulations, the memory imprint reduces by an order of magnitude. Moreover, the increased computational cost has to be evaluated for each specific problem. The application of the asymptotic expansion to full-core simulations, involving billions of computational regions with thousand of depleted media, is expected to be of extraordinary importance to control the memory needs, already huge in this type of calculations. The use of an effective acceleration (e.g., CMFD or BPA) will reduce the time to solution, thus increasing the global computational efficiency of the method.

10 - CP COEFFICIENTS - PROPERTIES ANALYSIS AND APPLICATIONS

10.1 . INTRODUCTION

As mentioned in Sect. 4.6.2 and 4.6.4, CP matrices are arrays of considerable size, which are responsible for massive memory pressure and non-negligible computational time. Generally, the array with the largest memory imprint and time requirements is the collision matrix. The overall number of collision coefficients depends linearly on the number of energy groups and angular directions and grows quadratically with the mean number of regions per cell and the number of spatial components. The analysis of CP properties has already been addressed by several works, as [101] [102]. Further references may be found in [5], [31], [103].

The present study is intended to emphasize the properties of CP matrices. We propose three simple developments:

- An approach approximating exponential-like functions by piece-wise Taylor polynomials, for optical paths up to few tens, and by their asymptotic behaviour, for larger values of τ (Sect. 10.2). The combined use of Eqs. (10.31)-(10.32) allows reducing the computational time for CP coefficients determination;
- A set of balance equations to be satisfied by CP matrices, for all incoming and outgoing degrees of freedom, enabling coefficients verification, up to double precision (Sect. 10.3);
- The use of symmetry and reciprocity properties, to reduce the number of stored directions and the computational cost of CP coefficients (Sect. 10.4).

10.2 . COEFFICIENTS TABLES AND ASYMPTOTIC APPROXIMATION

10.2.1 . Background

As illustrated in Sect. 4.6.4 (Chap. 4), the Method of Short Characteristics requires the calculation of angular-dependent probability matrices, whose expression is provided in Eq. (4.98)-(4.102) and depends on functions (4.103)-(4.105). The latter may be rewritten as [42]

$$\mathbf{F}(x^+, x^-, \mathbf{r}^-, \boldsymbol{\Omega}, \Sigma) = \int_{x^-}^{x^+} \mathbf{P}(\mathbf{r}^- + x\boldsymbol{\Omega}) e^{-\Sigma(x^+ - x)} dx, \quad (10.1)$$

$$\mathbf{G}(x^+, x^-, \mathbf{r}^-, \boldsymbol{\Omega}, \Sigma) = \int_{x^-}^{x^+} \mathbf{P}(\mathbf{r}^- + x\boldsymbol{\Omega}) e^{-\Sigma(x - x^-)} dx, \quad (10.2)$$

$$\underline{\mathbf{H}}(x^+, x^-, \mathbf{r}^-, \boldsymbol{\Omega}, \Sigma) = \int_{x^-}^{x^+} \mathbf{P}(\mathbf{r}^- + x\boldsymbol{\Omega}) \times \mathbf{F}(x, x^-, \mathbf{r}^-, \boldsymbol{\Omega}) dx, \quad (10.3)$$

and decomposed into

$$\mathbf{F}(x^+, x^-, \mathbf{r}^-, \boldsymbol{\Omega}, \Sigma) = \underline{\mathbf{B}}(\mathbf{r}^-, \boldsymbol{\Omega}) \underline{\boldsymbol{\Delta}}(l) \cdot \widehat{\mathbf{F}}(\tau^+, \tau^-), \quad (10.4)$$

$$\mathbf{G}(x^+, x^-, \mathbf{r}^-, \boldsymbol{\Omega}, \Sigma) = \underline{\mathbf{B}}(\mathbf{r}^-, \boldsymbol{\Omega}) \underline{\boldsymbol{\Delta}}(l) \cdot \widehat{\mathbf{G}}(\tau^+, \tau^-), \quad (10.5)$$

$$\underline{\mathbf{H}}(x^+, x^-, \mathbf{r}^-, \boldsymbol{\Omega}, \Sigma) = \underline{\mathbf{B}}(\mathbf{r}^-, \boldsymbol{\Omega}) \underline{\boldsymbol{\Delta}}(l) \widehat{\mathbf{H}}(\tau^+, \tau^-) \underline{\boldsymbol{\Delta}}(l) \underline{\mathbf{B}}^T(\mathbf{r}^-, \boldsymbol{\Omega}), \quad (10.6)$$

where, for LSC,

$$\hat{\mathbf{F}}(\tau^+, \tau^-) = \begin{bmatrix} \hat{F}_0(\tau^+, \tau^-) \\ \hat{F}_1(\tau^+, \tau^-) \end{bmatrix}, \quad (10.7)$$

$$\hat{\mathbf{G}}(\tau^+, \tau^-) = \begin{bmatrix} \hat{G}_0(\tau^+, \tau^-) \\ \hat{G}_1(\tau^+, \tau^-) \end{bmatrix}, \quad (10.8)$$

$$\hat{\mathbf{H}}(\tau^+, \tau^-) = \begin{bmatrix} \hat{H}_{0,0}(\tau^+, \tau^-) & \hat{H}_{0,1}(\tau^+, \tau^-) \\ \hat{H}_{1,0}(\tau^+, \tau^-) & \hat{H}_{1,1}(\tau^+, \tau^-) \end{bmatrix}, \quad (10.9)$$

$$\underline{\mathbf{B}}(\mathbf{r}^-, \boldsymbol{\Omega}) = \begin{bmatrix} 1 & 0 \\ \mathbf{r}^- - \mathbf{r}_0 & \boldsymbol{\Omega} \end{bmatrix}, \quad (10.10)$$

$$\underline{\boldsymbol{\Delta}}(l) = \begin{bmatrix} l & 0 \\ 0 & l^2 \end{bmatrix}. \quad (10.11)$$

Provided $\tau = l\Sigma = \tau^+ - \tau^-$, Eq. (10.7)-(10.9) read as

$$\hat{F}_n(\tau^+, \tau^-) = \frac{1}{\tau^{n+1}} \int_{\tau^-}^{\tau^+} dx x^n e^{-(\tau^+ - x)}, \quad (10.12)$$

$$\hat{G}_n(\tau^+, \tau^-) = \frac{1}{\tau^{n+1}} \int_{\tau^-}^{\tau^+} dx x^n e^{-(x - \tau^-)}, \quad (10.13)$$

$$\hat{H}_{n,m}(\tau^+, \tau^-) = \frac{1}{\tau^{n+m+2}} \int_{\tau^-}^{\tau^+} dx x^n \int_{\tau^-}^x dy y^m e^{-(x-y)}, \quad (10.14)$$

If the following transformation is introduced, [42],

$$x = \tau u + \tau^-, \quad u \in [0, 1], \quad (10.15)$$

Eq. (10.7)-(10.9) may be re-written as binomial expansions,

$$\hat{F}_n(\tau^+, \tau^-) = \sum_{i=0}^n \binom{n}{i} (\tau^-)^{n-i} \tau^i F_i(\tau), \quad (10.16)$$

$$\hat{G}_n(\tau^+, \tau^-) = \sum_{i=0}^n \binom{n}{i} (\tau^-)^{n-i} \tau^i G_i(\tau), \quad (10.17)$$

$$\hat{H}_{n,m}(\tau^+, \tau^-) = \sum_{i=0}^n \sum_{j=0}^m \binom{n}{i} \binom{m}{j} (\tau^-)^{m+n-(i+j)} \tau^{i+j} H_{i,j}(\tau), \quad (10.18)$$

with

$$G_n(\tau) = \int_0^1 du u^n e^{-\tau u}, \quad (10.19)$$

$$F_n(\tau) = \int_0^1 du u^n e^{-\tau(1-u)} = \sum_i^n \binom{n}{i} (-1)^i G_i(\tau), \quad (10.20)$$

$$H_{n,m}(\tau) = \int_0^1 du u^n \int_0^u dv v^m e^{-\tau(u-v)}, \quad (10.21)$$

depending on the optical path and the expansion order. Note that one has to compute only Eq. (10.19) and (10.21), due to Eq. (10.20).

10.2.2 . Analytic Remarks

Let us consider $\{G_n(\tau)\}_{n \in \mathbb{N}}$ (Eq. (10.19)). One can easily prove that an explicit expression for G_n is provided by

$$G_n(\tau) = \frac{n!}{\tau^{n+1}} \left(1 - S_n(\tau) e^{-\tau} \right), \quad (10.22)$$

$S_n(\tau)$ being the sequence of functions defined by

$$S_n(\tau) = \sum_{i=0}^n \frac{1}{(n-i)!} \tau^{n-i}. \quad (10.23)$$

By deriving $G_n(\tau)$, one can prove that

$$\frac{dG_n}{d\tau}(\tau) + G_{n+1}(\tau) = 0, \quad (10.24)$$

which recalls the hyperbolic nature of the transport equation. Similarly, an explicit expression may be obtained for $\{F_n(\tau)\}_{n \in \mathbb{N}}$ (Eq. (10.20)), namely

$$F_n(\tau) = \frac{n!}{\tau^{n+1}} \left(Q_n(\tau) + (-1)^{n+1} e^{-\tau} \right), \quad (10.25)$$

$Q_n(\tau)$ being the sequence of functions defined by

$$Q_n(\tau) = \sum_{i=0}^n \frac{(-1)^i}{(n-i)!} \tau^{n-i}. \quad (10.26)$$

By deriving $F_n(\tau)$, one can prove that

$$\frac{dF_n}{d\tau}(\tau) = F_{n+1}(\tau) - F_n(\tau). \quad (10.27)$$

Finally, let us consider $\{H_{n,m}(\tau)\}_{n,m \in \mathbb{N}}$. This can be written as

$$H_{n,m}(\tau) = \int_0^1 du u^n F_m(\tau, u), \quad (10.28)$$

where

$$F_m(\tau, u) = \int_0^u dv v^m e^{-\tau(u-v)}. \quad (10.29)$$

In other terms, the functions $H_{n,m}(\tau)$, with $n, m = 0, 1, 2, \dots$, represent the moments of $F_m(\tau, u)$, with respect to the monomial basis defined by $\mathbf{B}(u) = \{u^n\}_{n \in \mathbb{N}} = [1, u, u^2, \dots]^T$. As F_n depends on G_n , in the following we focus only on G_n and $H_{n,m}$, of which a representation is given in Figs. 10.1-10.2.

10.2.3 . Taylor Development and Asymptotic Behaviour

For the sake of simplicity, let us consider Eq. (10.19). In void regions, $\{G_n(\tau)\}_{n \in \mathbb{N}}$ is defined as

$$\lim_{\tau \rightarrow 0} G_n(\tau) = \frac{1}{n+1}, \quad n \in \mathbb{N}. \quad (10.30)$$

Equation (10.30) ensures the correct behaviour of short characteristics in void regions. Due to Eq. (10.24), the functions G_n satisfy the following:

$$G_n(\tau) = \sum_{i=0}^{\infty} (-1)^i \cdot \frac{(\tau - \bar{\tau})^i}{i!} G_{i+n}(\bar{\tau}), \quad (10.31)$$

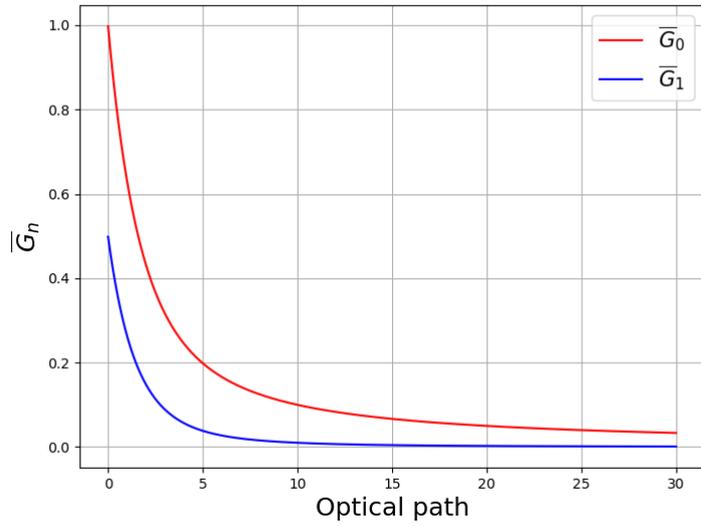


Figure 10.1: G_n vs τ , for $n = 0, 1$ (linear short characteristics) and $\tau \in [0, 30]$.

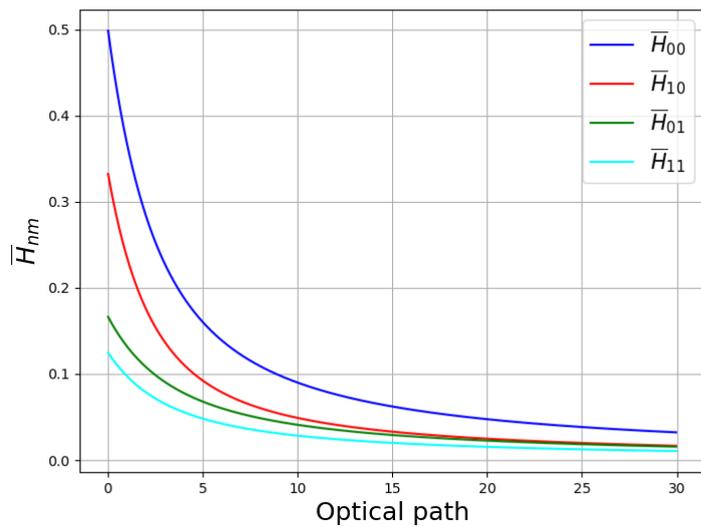


Figure 10.2: H_{nm} vs τ , for $n = 0, 1, m = 0, 1$ (linear short characteristics) and $\tau \in [0, 30]$.

where $\bar{\tau}$ is the centre of a Taylor series. Moreover, because of $e^{-\tau}$ in Eq. (10.22), the asymptotic behaviour of $G_n(\tau)$ is ruled by

$$G_n(\tau) \sim G_n^\infty(\tau) = \frac{n!}{\tau^{n+1}}, \forall \tau \gg 1, \quad (10.32)$$

as shown in Figs. 10.3-10.4. Functions (10.19)-(10.21) are computationally expensive, as they in-

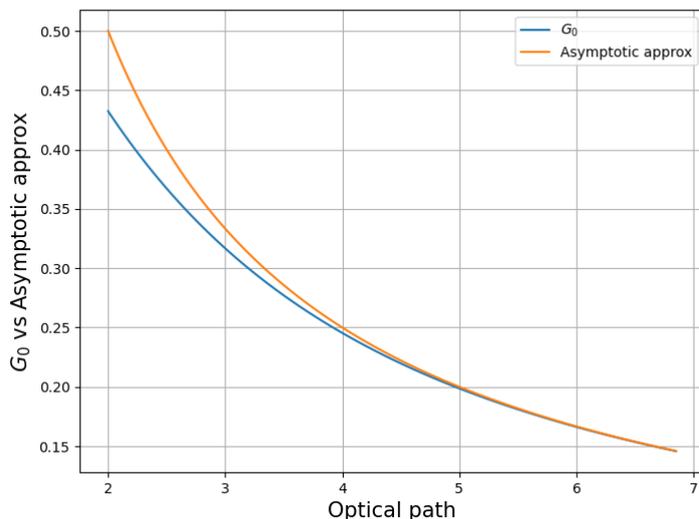


Figure 10.3: G_0 vs G_0^∞ , as a function of the optical path, for $\tau \gtrsim 1$.

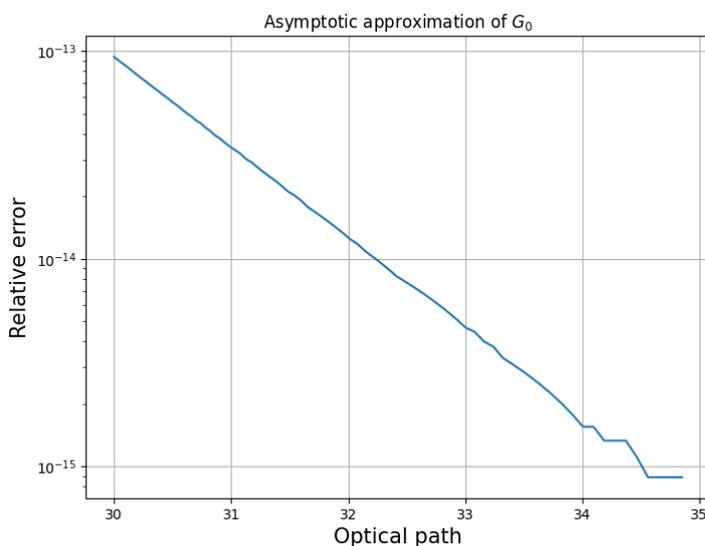


Figure 10.4: Relative difference between G_0 and G_0^∞ , as a function of the optical path, for $\tau > 30$.

volve the calculation of exponential functions, [104]. To lower the computational time of the angular probability matrices, IDT takes advantage of Eqs. (10.31)-(10.32). In practice, two values, τ_{inf} and τ_{sup} , have been selected. The latter is such that Eq. (10.32) is verified up to double precision, $\forall \tau \geq \tau_{sup}$. On the other hand, τ_{inf} is a relatively small negative value, of the order of few

units, that is able to account for negative optical lengths, which may appear in B_n -eigenvalue problems, in nearly vacuum regions as the pellet-clad gap. For all optical paths between τ_{inf} and τ_{sup} , the functions G_n are computed by piece-wise Taylor developments. In practice, the interval $[\tau_{inf}, \tau_{sup}]$ is divided into J sub-intervals of equal size, $\Delta\tau_j = \Delta\tau, j = 1, \dots, J$. For each sub-interval j , the following polynomial is defined

$$P_{I(j)}(\tau) = \sum_{i=0}^{I(j)} (-1)^i \cdot \frac{(\tau - \tau_j)^i}{i!} G_{i+n}(\tau_j), \tau \in \left[\tau_j - \frac{\Delta\tau}{2}, \tau_j + \frac{\Delta\tau}{2} \right], \quad (10.33)$$

where τ_j is the sub-interval mid-point and $I(j)$ is such that $P_{I(j)}$ is equal to $G_n(\tau)$ up to double precision. The coefficients of the polynomials are precomputed parameters, which are stored in memory at the beginning of each simulation and define the *coefficients tables*. In order to recover their values for a given sub-interval, IDT also stores the address of the zero-order coefficients. This is done by means of an array u , which in the case of G_0 , reads as

$$u = \left\{ ad(G_0^{(1)}), \dots, ad(G_0^{(J)}) \right\}. \quad (10.34)$$

ad being the address in memory for the specific coefficient. Another array, namely v , contains the Taylor coefficients. In the case of G_0 , this reads as

$$v = \left\{ G_0^{(1)}, \dots, G_{I(1)}^{(1)}, G_0^{(2)}, \dots, G_{I(2)}^{(2)}, \dots, G_0^{(J)}, \dots, G_{I(J)}^{(J)} \right\} = \left\{ G_0^{(j)}, \dots, G_{I(j)}^{(j)} \right\}_{j=1, \dots, J}. \quad (10.35)$$

where $G_i^{(j)} = G_i(\tau_j), \forall i = 0, \dots, I(j), \forall j = 1, \dots, J$.

Algo. 3 provides details about the practical evaluation of G_0 at a generic point $\tau \in [\tau_{inf}, +\infty[$.

Algorithm 3 : Evaluation of $G_0(\tau)$.

- 1: Given $\tau \in [\tau_{inf}, +\infty[$
 - 2: $G_0(\tau) = \frac{1}{\tau}$ {Asymptotic behaviour, Eq. (10.32)}
 - 3: **if** $\tau < \tau_{sup}$ **then**
 - 4: $j = \lceil \frac{\tau - \tau_{inf}}{\Delta\tau} \rceil$ {Sub-interval order number}
 - 5: $\tau_j = \tau_{inf} + \Delta\tau * j - \frac{\Delta\tau}{2}$ {Sub-interval mid-point}
 - 6: $f = u(j)$ {address of the first coefficient}
 - 7: $l = u(j + 1) - 1$ {address of the last coefficient}
 - 8: Compute $G_0(\tau)$ by Eq. (10.33), using $v(f), \dots, v(l)$ and τ_j
 - 9: **end if**
-

10.2.4 . Numerical Test

In order to prove the effectiveness of the presented approach, let us propose a simple 3x3 2D-benchmark, with UOX fuel and AIC control rod (Sect. 7.6.2), using a 281-group cross section library. The solid angle has been discretized by a rectangular $S_{48,24}$ Chebyshev-Legendre angular quadrature formula, while the spatial refinement comprises a total of 240 regions and 8 surfaces per cell side, in order to boost the computational time of the angular probability matrices. The analysis is limited to LSC. Two approaches have been compared, i.e., the standard strategy, which evaluates functions (10.19) and (10.21) by means of their own analytical expressions (except for $\tau = 0$, where a Taylor series development is necessary), and the strategy based

on the coefficients tables, for $\tau \in [\tau_{inf}, \tau_{sup}]$, and the asymptotic approximation, for $\tau > \tau_{sup}$. For estimating the CPU time, the simulations have been run repeatedly. On average, the proposed method allows reducing the computational time of the angular probability matrices by $\sim 15\%$, with negligible relative difference on the fundamental eigenvalue and on the scalar flux distribution per energy group and material, after 1 outer iteration.

10.3 . BALANCE EQUATIONS

In IDT, the numerical accuracy of the angular probability matrices may be tested up to double precision, using a debug option. This is made possible by a set of balance equations, of which the present analysis provides a mathematical proof. For the sake of brevity, let us consider linear short characteristics; the application to higher expansion order is straightforward.

Let us consider the boundary value problem defined by

$$(\boldsymbol{\Omega} \cdot \nabla + \Sigma(\mathbf{r}))\psi(\mathbf{r}) = q(\mathbf{r}), \mathbf{r} \in V \quad (10.36)$$

$$\psi(\mathbf{r}) = \psi_{in}(\mathbf{r}), \mathbf{r} \in \partial V^-. \quad (10.37)$$

Eq. (10.36) is projected onto a monomial basis of the kind $\mathbf{B}(\mathbf{r}) = [1, \mathbf{r}]$, with $\mathbf{r} = [x, y, z]^T$. By applying the identity $\nabla \cdot (\boldsymbol{\Omega}\psi) = \boldsymbol{\Omega} \cdot \nabla\psi$ and the divergence theorem, one obtains

$$\langle \mathbf{B}, \psi \rangle^+ - \langle \mathbf{B}, \psi \rangle^- + (\mathbf{L}^\dagger \mathbf{B}, \psi) = (\mathbf{B}, q), \quad (10.38)$$

where $\langle \cdot, \cdot \rangle^\pm$ and (\cdot, \cdot) denote a surface and volume scalar product, on ∂V^\pm and V , respectively, while \mathbf{L}^\dagger indicates the adjoint to the leakage-plus-removal operator. The flux, the neutron source and the interface angular flux are expanded onto \mathbf{P}_α and \mathbf{P}_s , defined as

$$P_\alpha(\mathbf{r}) = [1, \mathbf{r} - \mathbf{c}_\alpha]^T, \quad P_s(\mathbf{r}) = [1, \mathbf{r} - \mathbf{c}_s]^T, \quad (10.39)$$

with \mathbf{c}_α and \mathbf{c}_s being the region and surface centroid of α and s , respectively. Hence, we obtain

$$\sum_s \langle \mathbf{B}, \mathbf{P}_s \rangle^+ \psi_s^+ - \sum_{s'} \langle \mathbf{B}, \mathbf{P}_{s'} \rangle^- \psi_{s'}^- + \sum_\alpha (\mathbf{L}^\dagger \mathbf{B}, \mathbf{P}_\alpha) \psi_\alpha = \sum_\alpha (\mathbf{B}, \mathbf{P}_\alpha) \mathbf{q}_\alpha. \quad (10.40)$$

The scalar products in (10.40) may be interpreted as the surface mass matrices of s and s' , i.e. $\underline{\mathbf{M}}_{s/s'} = \langle \mathbf{B}, \mathbf{P}_{s/s'} \rangle^\pm$, the volume mass matrix of region α , $\underline{\mathbf{M}}_\alpha = (\mathbf{B}, \mathbf{P}_\alpha)$, and the removal + leakage matrix of region α , $\underline{\mathbf{S}}_\alpha = (\mathbf{L}^\dagger \mathbf{B}, \mathbf{P}_\alpha)$. The solution provided by MOSC reads as

$$\psi_{d,\alpha} = \sum_{s' \subset \partial V^-} \underline{\mathbf{I}}_{d,\alpha,s'} \psi_{d,s'}^- + \sum_\beta \underline{\mathbf{C}}_{d,\alpha,\beta} \mathbf{q}_{d,\beta} + \underline{\mathbf{C}}_{d,\alpha,\alpha} \mathbf{q}_{d,\alpha}, \quad (10.41)$$

$$\psi_{d,s}^+ = \sum_{s' \subset \partial V^-} \underline{\mathbf{T}}_{d,s,s'} \psi_{d,s'}^- + \sum_\alpha \underline{\mathbf{E}}_{d,s,\alpha} \mathbf{q}_{d,\alpha}, \quad (10.42)$$

where s' belongs to the incoming boundary ∂V^- illuminated by direction $\boldsymbol{\Omega}_d$ and $\psi_{d,s}^+$ and $\psi_{d,\alpha}$ represent respectively the spatial moments of the outgoing flux on surface s and of the angular flux in region α (Sect. 4.6.4). By introducing Eq. (10.41) and (10.42) into Eq. (10.40), the neutron balance becomes

$$\sum_{s'} \left[\sum_{s(s')} \underline{\mathbf{M}}_s \underline{\mathbf{T}}_{s,s'} + \sum_{\alpha(s')} \underline{\mathbf{S}}_\alpha \underline{\mathbf{I}}_{\alpha,s'} - \underline{\mathbf{M}}_{s'} \right] \psi_{s'}^- + \sum_\beta \left[\sum_{s(\beta)} \underline{\mathbf{M}}_s \underline{\mathbf{E}}_{s,\beta} + \sum_{\alpha(\beta)} \underline{\mathbf{S}}_\alpha \underline{\mathbf{C}}_{\alpha,\beta} - \underline{\mathbf{M}}_\beta \right] \mathbf{q}_\beta = \mathbf{0}. \quad (10.43)$$

Equation (10.43) has to be satisfied for all moments $\psi_{s'}$ and \mathbf{q}_β , leading to

$$\sum_{s(s')} \underline{\mathbf{M}}_s \underline{\mathbf{T}}_{s,s'} + \sum_{\alpha(s')} \underline{\mathbf{S}}_\alpha \underline{\mathbf{I}}_{\alpha,s'} = \underline{\mathbf{M}}_{s'}, \quad \forall s' \in \partial V^-, \quad (10.44)$$

$$\sum_{s(\beta)} \underline{\mathbf{M}}_s \underline{\mathbf{E}}_{s,\beta} + \sum_{\alpha(\beta)} \underline{\mathbf{S}}_\alpha \underline{\mathbf{C}}_{\alpha,\beta} = \underline{\mathbf{M}}_\beta, \quad \forall \beta \in V. \quad (10.45)$$

Equation (10.44) provides a set of conservation relations, relating all incoming and outgoing degrees of freedom. In IDT, Eq. (10.44) and (10.45) are used to verify the angular probability matrices. Nevertheless, they may also be suited to other numerical approaches, e.g., the method of collision probabilities, as a set normalization relations.

10.4 . MEMORY FOOTPRINT REDUCTION, BY SYMMETRY AND RECIPROCITY RELATIONS

IDT relies on the symmetry properties of the unit cell to ease the memory pressure and computational time of the angular probability matrices, by reducing the number of stored directions. For 2D cells with $1/8^{th}$ symmetry, only directions Ω_{ref} with XY-projection between 0 and 45° are directly computed and stored, the others are retrieved by applying simple transformations, i.e., a permutation to obtain the angular matrices along Ω_{sym} and a change of sign to attain the coefficients along direction $\Omega_d \perp \Omega_{sym}$ (Fig. 10.5), as indicated in Eq. (10.46)-(10.51). In particular, given the permutation matrix $\underline{\mathbf{P}}_{23} = [\mathbf{e}_1, \mathbf{e}_3, \mathbf{e}_2]^T$, where \mathbf{e}_i is the i -th unit vector of the canonical basis in \mathbb{R}^3 , the following relations hold, for 2D LSC:

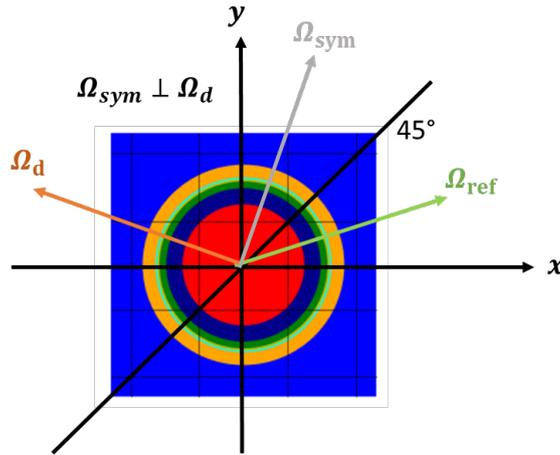


Figure 10.5: Symmetry relations in IDT: 2D cell with $1/8^{th}$ symmetry. N.B. Ω_{sym} is obtained by reflection of Ω_{ref} (reference direction) about the direction at 45° . Ω_d is perpendicular to Ω_{sym} .

$$\underline{\mathbf{C}}_{(\alpha,\beta)_{sym}}(\Omega_{sym}) = \underline{\mathbf{P}}_{23} \underline{\mathbf{C}}_{(\alpha,\beta)}(\Omega_{ref}) \underline{\mathbf{P}}_{23}, \quad \underline{\mathbf{I}}_{(\alpha,s')_{sym}}(\Omega_{sym}) = \underline{\mathbf{P}}_{23} \underline{\mathbf{I}}_{(\alpha,s')}(\Omega_{ref}), \quad (10.46)$$

$$\underline{\mathbf{E}}_{(s,\beta)_{sym}}(\Omega_{sym}) = \underline{\mathbf{E}}_{(s,\beta)}(\Omega_{ref}) \underline{\mathbf{P}}_{23}, \quad \underline{\mathbf{T}}_{(s,s')_{sym}}(\Omega_{sym}) = \underline{\mathbf{T}}_{(s,s')}(\Omega_{ref}), \quad (10.47)$$

$$\underline{\mathbf{C}}_{(\alpha,\beta)_\perp}(\Omega_\perp) = \text{sgn}(\mathbf{B}_{\alpha_\perp} \mathbf{B}_{\beta_\perp}^T) * \underline{\mathbf{C}}_{(\alpha,\beta)_{sym}}(\Omega_{sym}), \quad (10.48)$$

$$\underline{\mathbf{E}}_{(s,\beta)_\perp}(\Omega_\perp) = \text{sgn}(\mathbf{B}_{s_\perp} \mathbf{B}_{\beta_\perp}^T) * \underline{\mathbf{E}}_{(s,\beta)_{sym}}(\Omega_{sym}), \quad (10.49)$$

$$\underline{\mathbf{I}}_{(\alpha,s')_\perp}(\Omega_\perp) = \text{sgn}(\mathbf{B}_{\alpha_\perp} \mathbf{B}_{s'_\perp}^T) * \underline{\mathbf{I}}_{(\alpha,s')_{sym}}(\Omega_{sym}), \quad (10.50)$$

$$\underline{\mathbf{T}}_{(s,s')_\perp}(\Omega_\perp) = \text{sgn}(\mathbf{B}_{s_\perp} \mathbf{B}_{s'_\perp}^T) * \underline{\mathbf{T}}_{(s,s')_{sym}}(\Omega_{sym}), \quad (10.51)$$

where \mathbf{B} denotes the monomial basis, sgn is the sign function and $*$ indicates the Hadamard product of two matrices.

Similar arguments apply to 3D cells, with $1/16^{th}$ symmetry. Equation (10.46) still holds for the collision matrix, provided that $\mathbf{P}_{23} = [\mathbf{e}_1, \mathbf{e}_3, \mathbf{e}_2, \mathbf{e}_4]^T$. Nevertheless, a distinction has to be made for the incoming, escape and transmission matrices, as the symmetry relations depend on the orientation of the incoming and outgoing surfaces. In particular,

$$\underline{\mathbf{I}}_{(\alpha,s')_{sym}}(\Omega_{sym}) = \mathbf{P}_{23} \underline{\mathbf{I}}_{(\alpha,s')}(\Omega_{ref}), \text{ if } \mathbf{n}_{s'} \parallel \mathbf{e}_{1/2}, \quad (10.52)$$

$$\underline{\mathbf{I}}_{(\alpha,s')_{sym}}(\Omega_{sym}) = \mathbf{P}_{23} \underline{\mathbf{I}}_{(\alpha,s')}(\Omega_{ref}) \mathbf{P}_{23}, \text{ if } \mathbf{n}_{s'} \parallel \mathbf{e}_3, \quad (10.53)$$

for the incoming matrix,

$$\underline{\mathbf{E}}_{(s,\beta)_{sym}}(\Omega_{sym}) = \underline{\mathbf{E}}_{(s,\beta)}(\Omega_{ref}) \mathbf{P}_{23}, \text{ if } \mathbf{n}_s \parallel \mathbf{e}_1 \text{ or } \mathbf{n}_s \parallel \mathbf{e}_2, \quad (10.54)$$

$$\underline{\mathbf{E}}_{(s,\beta)_{sym}}(\Omega_{sym}) = \mathbf{P}_{23} \underline{\mathbf{E}}_{(s,\beta)}(\Omega_{ref}) \mathbf{P}_{23}, \text{ if } \mathbf{n}_s \parallel \mathbf{e}_3, \quad (10.55)$$

for the escape matrix, and

$$\underline{\mathbf{T}}_{(s,s')_{sym}}(\Omega_{sym}) = \mathbf{P}_{23} \underline{\mathbf{T}}_{(s,s')}(\Omega_{ref}), \text{ if } \mathbf{n}_s \parallel \mathbf{e}_3 \text{ and } \mathbf{n}_{s'} \parallel \mathbf{e}_{1/2}, \quad (10.56)$$

$$\underline{\mathbf{T}}_{(s,s')_{sym}}(\Omega_{sym}) = \underline{\mathbf{T}}_{(s,s')}(\Omega_{ref}) \mathbf{P}_{23}, \text{ if } \mathbf{n}_{s'} \parallel \mathbf{e}_3 \text{ and } \mathbf{n}_s \parallel \mathbf{e}_{1/2}, \quad (10.57)$$

$$\underline{\mathbf{T}}_{(s,s')_{sym}}(\Omega_{sym}) = \mathbf{P}_{23} \underline{\mathbf{T}}_{(s,s')}(\Omega_{ref}) \mathbf{P}_{23}, \text{ if } \mathbf{n}_{s'} \parallel \mathbf{e}_3 \text{ and } \mathbf{n}_s \parallel \mathbf{e}_3, \quad (10.58)$$

$$\underline{\mathbf{T}}_{(s,s')_{sym}}(\Omega_{sym}) = \underline{\mathbf{T}}_{(s,s')}(\Omega_{ref}), \text{ otherwise,} \quad (10.59)$$

for the transmission matrix. Equations (10.48)-(10.51) still hold.

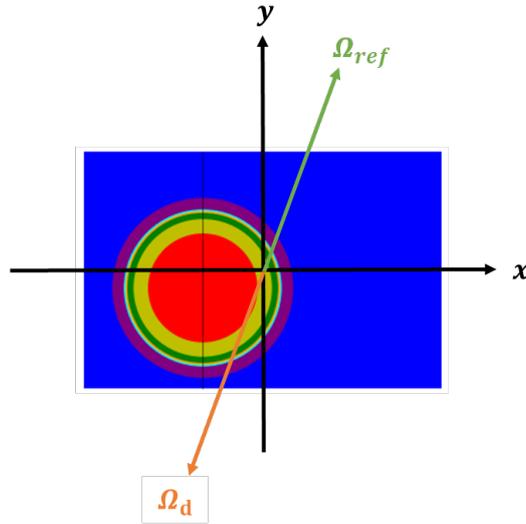


Figure 10.6: Symmetry relations in IDT: asymmetric HCC.

A cell may also be completely asymmetric, as in Fig. 10.6. In this event, only directions on the half sphere are directly computed and stored, the others may be derived by simple reciprocity relations:

$$\underline{\mathbf{M}}_{\alpha} \underline{\mathbf{C}}_{\alpha,\beta}(-\Omega_{\mathbf{d}}) = [\underline{\mathbf{M}}_{\beta} \underline{\mathbf{C}}_{\beta,\alpha}(\Omega_{\mathbf{d}})]^T, \quad (10.60)$$

$$\underline{\mathbf{M}}_{\mathbf{s}} \underline{\mathbf{E}}_{\mathbf{s},\alpha}(-\Omega_{\mathbf{d}}) = [\underline{\mathbf{M}}_{\alpha} \underline{\mathbf{I}}_{\alpha,\mathbf{s}}(\Omega_{\mathbf{d}})]^T, \quad (10.61)$$

$$\underline{\mathbf{M}}_{\alpha} \underline{\mathbf{I}}_{\alpha,\mathbf{s}}(-\Omega_{\mathbf{d}}) = [\underline{\mathbf{M}}_{\mathbf{s}} \underline{\mathbf{E}}_{\mathbf{s},\alpha}(\Omega_{\mathbf{d}})]^T, \quad (10.62)$$

$$\underline{\mathbf{M}}_{\mathbf{s}} \underline{\mathbf{T}}_{\mathbf{s},\mathbf{s}'}(-\Omega_{\mathbf{d}}) = [\underline{\mathbf{M}}_{\mathbf{s}'} \underline{\mathbf{T}}_{\mathbf{s}',\mathbf{s}}(\Omega_{\mathbf{d}})]^T, \quad (10.63)$$

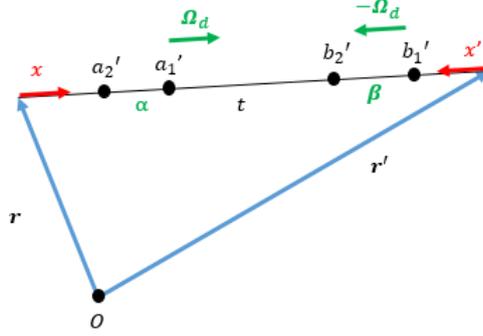


Figure 10.7: Reciprocity relations.

Let us provide a proof of Eq. (10.60). Fig. 10.7 depicts a trajectory $t \parallel -\Omega_d$, entering the domain at \mathbf{r}' and exiting at \mathbf{r} . x' is the coordinate along t , with origin at \mathbf{r}' and direction $-\Omega_d$. The points delimiting the regions α and β along t have coordinates a_1', a_2', b_1' and b_2' , respectively. Due to Eq. (4.99), the integrand of the term on the left-hand side of Eq. (10.60) reads as

$$\int_{a_1'}^{a_2'} e^{-\Sigma_\alpha(x'-a_1)} \mathbf{P}_\alpha(\mathbf{r}' - x' \Omega_d) dx' \times \int_{b_1'}^{b_2'} e^{-\Sigma_\beta(b_2'-x')} \mathbf{P}_\beta(\mathbf{r}' - x' \Omega_d) dx'. \quad (10.64)$$

The reader will notice that the exponential of the sum of the optical path lengths between b_2' and a_1' (Eq. (4.99)) has been neglected, as this does not change if the trajectory is swept along $-\Omega_d$ or along Ω_d . Let us now consider the integrand of the term on the right hand-side and refer to the coordinate system with origin at \mathbf{r} and direction Ω_d , namely x , in Fig. 10.7. If $l = \|\mathbf{r}' - \mathbf{r}\|$ is the trajectory length, the integrand of $\underline{\mathbf{M}}_\beta \underline{\mathbf{C}}_{\beta,\alpha}(\Omega_d)$ reads as

$$\int_{l-b_2'}^{l-b_1'} e^{-\Sigma_\beta(x-(l-b_2))} \mathbf{P}_\beta(\mathbf{r} + x \Omega_d) dx \times \int_{l-a_2'}^{l-a_1'} e^{-\Sigma_\alpha((l-a_1)-x)} \mathbf{P}_\alpha(\mathbf{r} + x \Omega_d) dx. \quad (10.65)$$

By replacing x with x' , where $x' = l - x$, one obtains

$$\int_{b_1'}^{b_2'} e^{-\Sigma_\beta(b_2'-x')} \mathbf{P}_\beta(\mathbf{r}' - x' \Omega_d) dx' \times \int_{a_1'}^{a_2'} e^{-\Sigma_\alpha(x'-a_1')} \mathbf{P}_\alpha(\mathbf{r}' - x' \Omega_d) dx', \quad (10.66)$$

which is the transposed of Eq. (10.64). Equations (10.61)-(10.63) are derived in a similar way and are not analysed, for brevity's sake.

Finally, for 3D transport, a cell may be $1/8^{\text{th}}$ symmetric about the XY plane, but have no symmetry along Z. An example is provided in Fig. 10.8, where the discretization steps above and below $z = 0$ are not specular. In this case, only the coefficients pertaining to the directions on the grey section of Fig. (10.8) are stored in memory. All the others may be derived, applying XY symmetry relations.

Symmetry and reciprocity relations allow for a significant saving of CPU time and memory occupation, which can be reduced by a factor 16, 8 or 2, depending on the geometry dimensionality and symmetry properties. Note that, while the use of the prior reciprocity relations may be generalised to any order of expansion, Eqs. (10.46)-(10.59) are specific to LSC and their extension to parabolic short characteristics is not possible, as illustrated in the upcoming chapter.

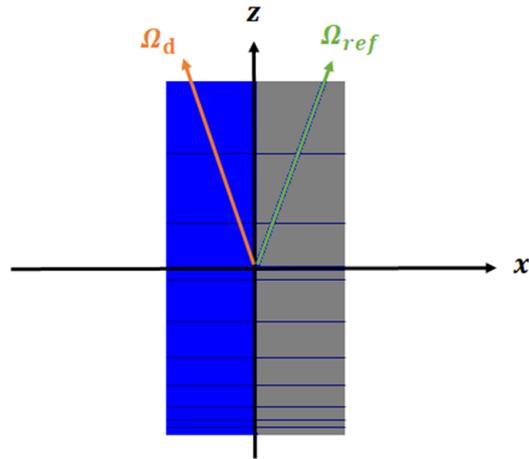


Figure 10.8: Symmetry relations in IDT: asymmetric 3D cell along the z-axis. Symmetry on the XY plane is still verified.

10.5 . CONCLUSION

In this chapter, some mathematical properties of CP coefficients have been illustrated and applied to optimize the implementation of constant and linear short characteristics in IDT. In particular, the application of symmetry and reciprocity relations (Sect. 10.4) allows for a major reduction of memory and CPU time for CP matrices storage and calculation, by a factor 16, 8 or 2, depending on the symmetry properties of the cell. The computational time can be furtherly mitigated by the combined use of Taylor series development and asymptotic approximation of exponential-like functions, allowing for the determination of the angular probability matrices, with unmodified numerical accuracy (Sect. 10.2). Finally, a new set of balance equations has been derived and applied to CP coefficients V&V (Sect. 10.3), up to double precision. This tool has also been extended to verify short characteristics up to incomplete parabolic expansion, as illustrated in Chap. 11.

11 - INCOMPLETE PARABOLIC SHORT CHARACTERISTICS

11.1 . INTRODUCTION

Methods based on collision probabilities (Sect. 4.6.2), e.g., the method of short characteristics (Sect. 4.6.4), are able to model unstructured geometries and remain finite, when the total cross section approaches zero. As a result, on the one hand, they are capable of describing systems with high geometrical complexity and control-rod movements, with no homogenization, and, on the other hand, to model empty media, in an exact way. On the downside, CP-matrices are responsible for high memory pressure, particularly the collision matrix, whose size is proportional to the square of the number of regions per cell.

The present chapter focuses on the method of short characteristics and, more specifically, on its extension up to parabolic order, without including the bilinear terms. The main objective is to show that such a choice allows for a major reduction of the number of regions, without degrading the numerical accuracy. For non-symmetric cells, the increase in the memory footprint, due to the growth of spatial components, is generally outmatched by the reduction of computational regions.

Nevertheless, standard incomplete parabolic short characteristics do not verify the symmetry relations discussed in Chap. 10, Sect. 10.4. As parabolic-order radial coefficients are not permutable, a mixed-order approach, employing piece-wise linear distributions for the x and y -coordinate and incomplete parabolic development along the z -axis, may be preferred. In this way, one can employ the aforementioned transformations, i.e., Eqs. 10.54-10.59 (Chap. 10), used for linear short characteristics, to decrease the number of stored directions, and, at the same time benefit from piece-wise parabolic expansion in z , to coarsen the spatial discretization along the vertical axis.

The chapter is structured as follows. First, Sects. 11.2-11.3 provide a theoretical introduction to incomplete parabolic short characteristic. Then, two numerical tests are proposed, i.e., the set of 2D 3×3 UOX/MOX patterns illustrated in Sect. 7.6.2 and the 3D Watts Bar benchmark, to verify the numerical accuracy of PSC, by comparisons with TRIPOLI-4®. After this, a heuristic proof shows that the symmetry relations described in Sect. 10.4 do not hold for incomplete parabolic short characteristics (Sect. 11.4.3). Hence, in Sect. 11.5, a hybrid numerical pathway, using linear piece-wise polynomials on the radial plane and parabolic development in z is proposed for short characteristics, in order to recover the symmetry relations on the radial plane and, at the same time, reduce the number of regions.

11.2 . THEORY

MOSC is a method based on angular current-coupled collision probabilities (Sect. 4.6.2), which is able to model anisotropic scattering sources (Sects. 4.3). Short Characteristics generally assume piece-wise polynomial distributions of the neutron source and interface angular flux (Sect. 4.6.4). In the present discussion, the polynomial series development is generalized to parabolic order, without including the bilinear terms. In fact, based on previous studies in

IDT [69], bilinear short characteristics (BSC) provide little numerical benefit with respect to LSC, while the memory occupation increases dramatically, i.e., by a factor 4 for 3D BSC.

In order to introduce the method of incomplete parabolic short characteristics (PSC), let us consider the one-speed integral form of the Boltzmann equation, along a trajectory $t \parallel \Omega_d$, at a point $\mathbf{r} = \mathbf{r}_t^- + \xi \Omega_d$, namely

$$\psi_d(\mathbf{r}_t^- + \xi \Omega_d) = \psi_d(\mathbf{r}_t^-) e^{-\tau_t(\xi, 0)} + \int_0^\xi q_d(\mathbf{r}^- + \xi' \Omega_d) e^{-\tau_t(\xi, \xi')} d\xi'. \quad (11.1)$$

In Eq. (11.1), $\tau_t(\xi, \xi') = \int_{\xi'}^\xi d\xi'' \Sigma(\mathbf{r}_t^- + \xi'' \Omega_d)$ is the optical path length, ξ is the coordinate along the trajectory and \mathbf{r}_t^- is the hang-point of the trajectory on the incoming surface of the domain. Given a region $\alpha \in V$ and a surface $s \in \partial V$, V being a cell and ∂V its boundary, the neutron source and interface angular flux are developed as follows,

$$q_{d,\alpha}(\mathbf{r}) = \sum_{c=1}^C P_{\alpha,c}(\mathbf{r}) q_{d,\alpha,c} = \mathbf{P}_\alpha(\mathbf{r}) \cdot \mathbf{q}_{d,\alpha}, \quad (11.2)$$

$$\psi_{d,s}(\mathbf{r}) = \sum_{b=1}^B P_{s,b}(\mathbf{r}) \psi_{d,s,b} = \mathbf{P}_s(\mathbf{r}) \cdot \psi_{d,s}, \quad (11.3)$$

in which $C = 4$ and $B = 3$ for LSC, whereas $C = 7$ and $B = 5$ for PSC. More precisely, provided that $\mathbf{r} = [x, y, z]^T$ and $\mathbf{r}^2 = [x^2, y^2, z^2]^T$ denote the first and second order monomials, the volume and boundary grid basis functions read as

$$\mathbf{P}_\alpha(\mathbf{r}) = \begin{bmatrix} 1 \\ \mathbf{r} - \mathbf{c}_\alpha \\ \mathbf{r}^2 - \mathbf{d}_\alpha^2 \end{bmatrix}, \quad \mathbf{P}_s(\mathbf{r}) = \begin{bmatrix} 1 \\ \mathbf{r} - \mathbf{c}_s \\ \mathbf{r}^2 - \mathbf{d}_s^2 \end{bmatrix}, \quad (11.4)$$

where the volume and surface centroids, \mathbf{c}_α and \mathbf{c}_s , and the second order vectors, \mathbf{d}_α^2 and \mathbf{d}_s^2 , are such that the zero order moments are orthogonal to the first and to the second order moments, respectively, i.e.,

$$\int_{D_\alpha/\Gamma_s} d^3r (\mathbf{r} - \mathbf{c}_{\alpha/s}) = 0, \quad \int_{D_\alpha/\Gamma_s} d^3r (\mathbf{r}^2 - \mathbf{d}_{\alpha/s}^2) = 0. \quad (11.5)$$

As a result, the volume and surface mass matrices (Eq. (4.89)) take the form

$$\underline{\mathbf{M}}_\alpha = (\mathbf{P}_\alpha, \mathbf{P}_\alpha) = \begin{bmatrix} 1 & \mathbf{0}^T & \mathbf{0}^T \\ \mathbf{0} & \langle \mathbf{r} - \mathbf{c}_\alpha, \mathbf{r} - \mathbf{c}_\alpha \rangle & \langle \mathbf{r} - \mathbf{c}_\alpha, \mathbf{r}^2 - \mathbf{d}_\alpha^2 \rangle \\ \mathbf{0} & \langle \mathbf{r}^2 - \mathbf{d}_\alpha^2, \mathbf{r} - \mathbf{c}_\alpha \rangle & \langle \mathbf{r}^2 - \mathbf{d}_\alpha^2, \mathbf{r}^2 - \mathbf{d}_\alpha^2 \rangle \end{bmatrix}, \quad (11.6)$$

$$\underline{\mathbf{M}}_s = (\mathbf{P}_s, \mathbf{P}_s) = \begin{bmatrix} 1 & \mathbf{0}^T & \mathbf{0}^T \\ \mathbf{0} & \langle \mathbf{r} - \mathbf{c}_s, \mathbf{r} - \mathbf{c}_s \rangle & \langle \mathbf{r} - \mathbf{c}_s, \mathbf{r}^2 - \mathbf{d}_s^2 \rangle \\ \mathbf{0} & \langle \mathbf{r}^2 - \mathbf{d}_s^2, \mathbf{r} - \mathbf{c}_s \rangle & \langle \mathbf{r}^2 - \mathbf{d}_s^2, \mathbf{r}^2 - \mathbf{d}_s^2 \rangle \end{bmatrix}, \quad (11.7)$$

with

$$\langle \mathbf{v}, \mathbf{w}_\alpha \rangle = \int_{D_\alpha} \mathbf{v} \mathbf{w}_\alpha^T d^3r, \quad (11.8)$$

$$\langle \mathbf{v}, \mathbf{w}_s \rangle = \oint_{\Gamma_s} \mathbf{v} \mathbf{w}_s^T d^3r, \quad (11.9)$$

$$\mathbf{0} = [0, 0, 0]^T. \quad (11.10)$$

This implies that the zero order moment coincides with the average value. The polynomial basis is decomposed as

$$\mathbf{P}_\alpha(\mathbf{r}) = \underline{\mathbf{B}}_\alpha(\mathbf{r}^-, \boldsymbol{\Omega})\mathbf{L}(l), \quad (11.11)$$

where

$$\underline{\mathbf{B}}_\alpha(\mathbf{r}^-, \boldsymbol{\Omega}) = \begin{bmatrix} 1 & 0 & 0 \\ \mathbf{r}^- - \mathbf{c}_\alpha & \boldsymbol{\Omega} & 0 \\ (\mathbf{r}^-)^2 - \mathbf{d}_\alpha^2 & 2\mathbf{r}^- * \boldsymbol{\Omega} & \boldsymbol{\Omega}^2 \end{bmatrix}, \quad \mathbf{L}(l) = \begin{bmatrix} 1 \\ l \\ l^2 \end{bmatrix}. \quad (11.12)$$

with * being the Hadamard product and $\boldsymbol{\Omega}^2 = [\Omega_1^2, \Omega_2^2, \Omega_3^2]^T$. In order to determine the angular probability matrices, we resort to Eqs. (10.4)-(10.6), where

$$\widehat{\mathbf{F}}(\tau^+, \tau^-) = \begin{bmatrix} \widehat{F}_0(\tau^+, \tau^-) \\ \widehat{F}_1(\tau^+, \tau^-) \\ \widehat{F}_2(\tau^+, \tau^-) \end{bmatrix}, \quad (11.13)$$

$$\widehat{\mathbf{G}}(\tau^+, \tau^-) = \begin{bmatrix} \widehat{G}_0(\tau^+, \tau^-) \\ \widehat{G}_1(\tau^+, \tau^-) \\ \widehat{G}_2(\tau^+, \tau^-) \end{bmatrix}, \quad (11.14)$$

$$\widehat{\mathbf{H}}(\tau^+, \tau^-) = \begin{bmatrix} \widehat{H}_{0,0}(\tau^+, \tau^-) & \widehat{H}_{0,1}(\tau^+, \tau^-) & \widehat{H}_{0,2}(\tau^+, \tau^-) \\ \widehat{H}_{1,0}(\tau^+, \tau^-) & \widehat{H}_{1,1}(\tau^+, \tau^-) & \widehat{H}_{1,2}(\tau^+, \tau^-) \\ \widehat{H}_{2,0}(\tau^+, \tau^-) & \widehat{H}_{2,1}(\tau^+, \tau^-) & \widehat{H}_{2,2}(\tau^+, \tau^-) \end{bmatrix}. \quad (11.15)$$

As for CSC and LSC, the CP matrices can be verified up to double precision, for all degrees of freedom, by extending Eqs. (10.44)-(10.45) to PSC. With respect to Sect. 10.3, it is enough to redefine the monomial basis as $\mathbf{B} = [1, \mathbf{r}, \mathbf{r}^2]^T$, and the polynomial bases as indicated in Eq. (11.4). The generalisation to PSC is straightforward.

11.3 . REMARKS

The motivations for PSC are multiple, and comprise the possibility to lower the number of computational regions, while retaining high numerical accuracy, and also to facilitate users' work, as the HCC inner discretization can be greatly coarsened and simplified.

As mentioned in Chaps. 9-10, short characteristics are affected by large memory pressure, due to the size of the angular probability matrices. More specifically, the dimensions of the CP matrices, for one HCC, energy group and angular direction, depend on the spatial discretization and the expansion order, as indicated in Tab. 11.1, where N_r and N_s represent the number of regions within a cell, V , and the number of surfaces on its boundary, ∂V , respectively. In real problems, the array that is responsible for the largest memory occupation is the collision matrix. As the number of volume moments raises from 4 to 7, from LSC to PSC, the idea is to lower the number of computational regions by half. This choice is also supported by the mathematical shape of the neutron flux in the fuel pins and along the z-axis and is extensively tested in Sect. 11.4.

11.4 . NUMERICAL TESTS

In the following, the numerical accuracy of PSC is tested, by comparisons with TRIPOLI-4®, [105]. The presented analysis also proposes a comparison between PSC and LSC, showing that

Matrix	Discretization	Components
Collision	$\propto N_r^2$	$\propto C^2$
Incoming	$\propto N_r \cdot N_s$	$\propto C \cdot B$
Escape	$\propto N_s \cdot N_r$	$\propto B \cdot C$
Transmission	$\propto N_s^2$	$\propto B^2$

Table 11.1: Size of the CP matrices, given 1 HCC, 1 energy group and 1 discrete direction: N_r and N_s denote the number of regions and the number of surfaces for the considered cell, while C and B indicate the number of volume and surface components of the polynomial series expansion.

the former attains a precision of the same order as the latter, despite the significant reduction in the number of regions.

11.4.1. Numerical Accuracy of PSC - 2D 3x3 UOX/MOX patterns

The first benchmark is the set of 2D 3x3 UOX/MOX patterns illustrated in Sect. 7.6.2. In order for PSC to be of any practical interest, the spatial mesh has to be reduced to the strict minimum. Hence, the number of regions in a cell is set equal to the number of cylinders + 1. The solid angle is discretized into 32 directions per octant, by $S_{16,8}$ Chebyshev-Legendre product quadrature. The cell boundary is subdivided into $N_{sub,x} = N_{sub,y} = N_{sub}$ equal-sized surfaces per HCC side. The cross section library contains 281 energy groups and P_3 scattering matrices. PSC is compared to TRIPOLI-4@ continuous-energy Monte Carlo, for $N_{sub} = 1, 2, 3, 4$, in terms of multigroup scalar flux per SSH material. This choice entails relatively large errors, as there are no compensations due to the fine energy grid (Sect. 7.6.2). Fig. 11.1 displays the RMS relative error (%) per SSH material and the relative error (pcm) on k_{eff} . For the sake of conciseness, the former pertains the sole UOX-AIC case, but similar results are observed for the whole set of UOX/MOX patterns. One can observe that the lines corresponding to $N_{sub} = 3$ and $N_{sub} = 4$ tend to overlap.

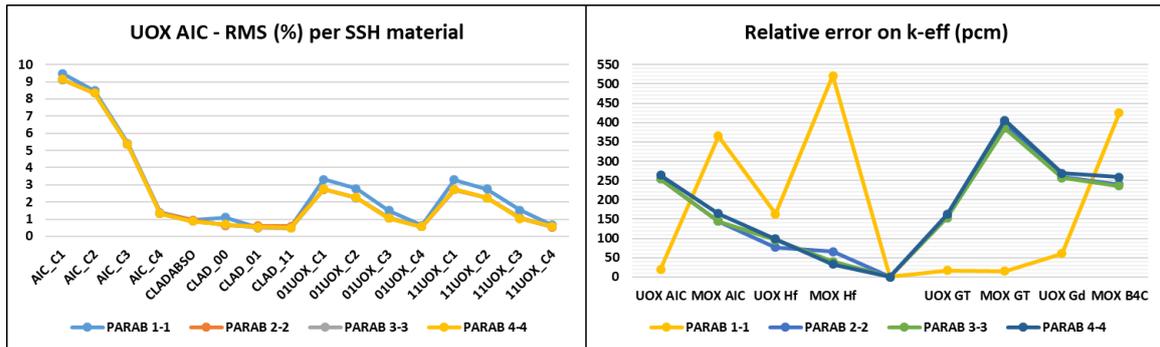


Figure 11.1: PSC IDT, with $N_{sub} = 1, 2, 3, 4$, vs TRIPOLI-4@ continuous-energy Monte Carlo: on the left-hand side, the RMS relative error, computed over 281 energy groups, per SSH material, for the UOX-AIC benchmark; on the right-hand side, the relative error on the fundamental eigenvalue, for the ensemble of test cases. The curves pertaining to $N_{sub} = 3$ and $N_{sub} = 4$ tend to superimpose.

Thus, in the following, the number of surfaces per side is set to $N_{sub} = 3$, for PSC. Tab. 11.2 compares the total number of regions applied to PSC and LSC, where LSC is discretized as in Sect. 7.6.2, for the whole set of UOX/MOX patterns. Let us compare PSC and LSC to TRIPOLI-

4®, in terms of average RMS relative error (%) per material type (Figs. 11.2-11.3). For the sake of completeness, CSC and step-MOC (APOLLO3®-TDT) are also displayed. All simulations show

Model	AIC	Hf	GT	Gd	B ₄ C
LSC	128/161	128	164	109	128
PSC	65	65	59	70	65

Table 11.2: LSC vs PSC: total number of regions for each UOX/MOX pattern. For LSC, UOX-AIC and MOX-AIC do not share the same mesh. In this case, the number of regions is indicated separately as UOX/MOX.

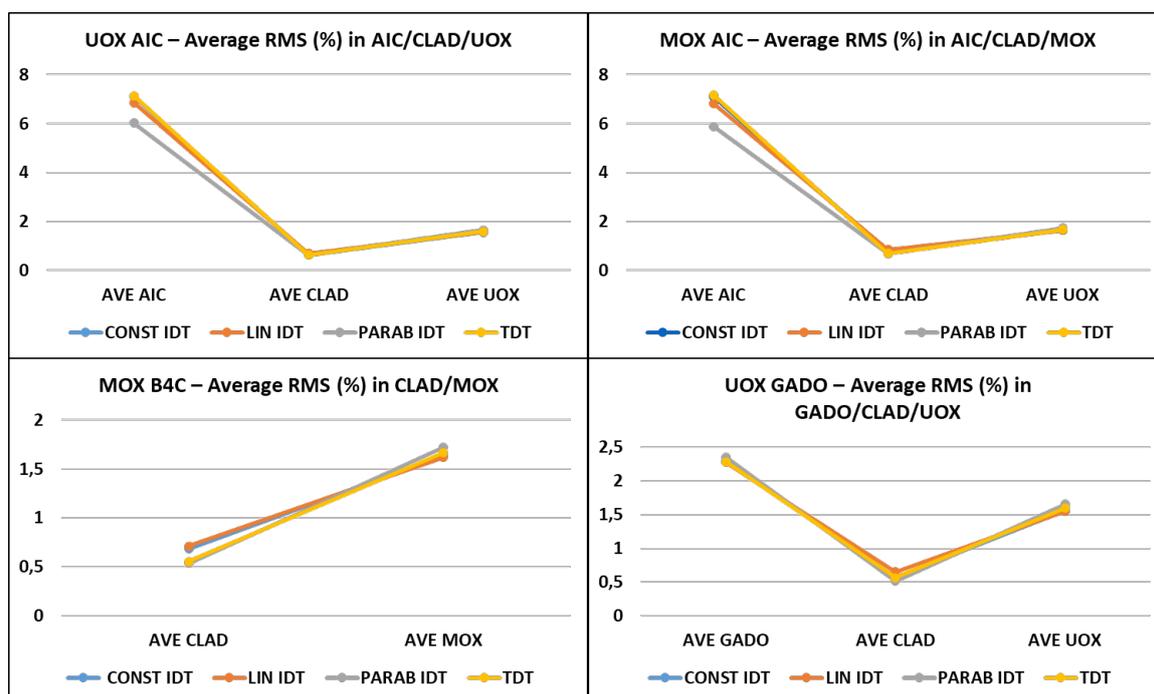


Figure 11.2: Comparison with TRIPOLI-4® continuous-energy Monte Carlo: average RMS relative error on the scalar flux, for each material type, computed over 281 energy groups.

similar numerical accuracy. As in Sect. 7.6.2, large errors are observed in the energy spectrum (Fig. 11.4), due to the Self-Shielding model.

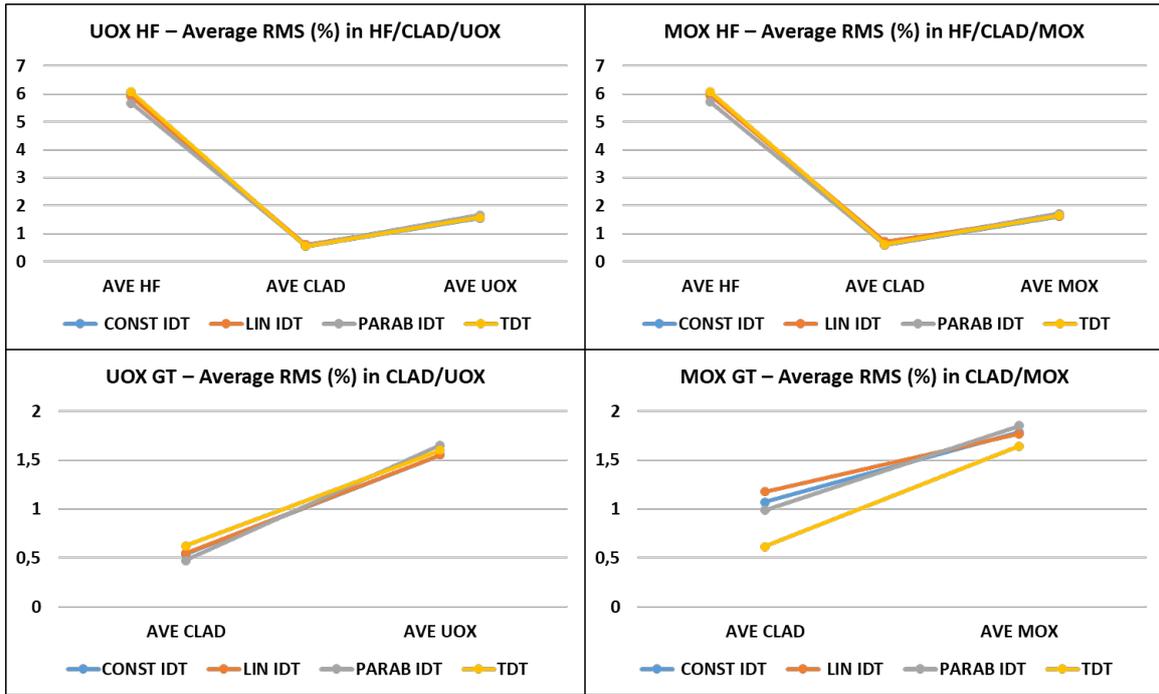


Figure 11.3: Comparison with TRIPOLI-4® continuous-energy Monte Carlo: average RMS relative error on the scalar flux, for each material type, computed over 281 energy groups.

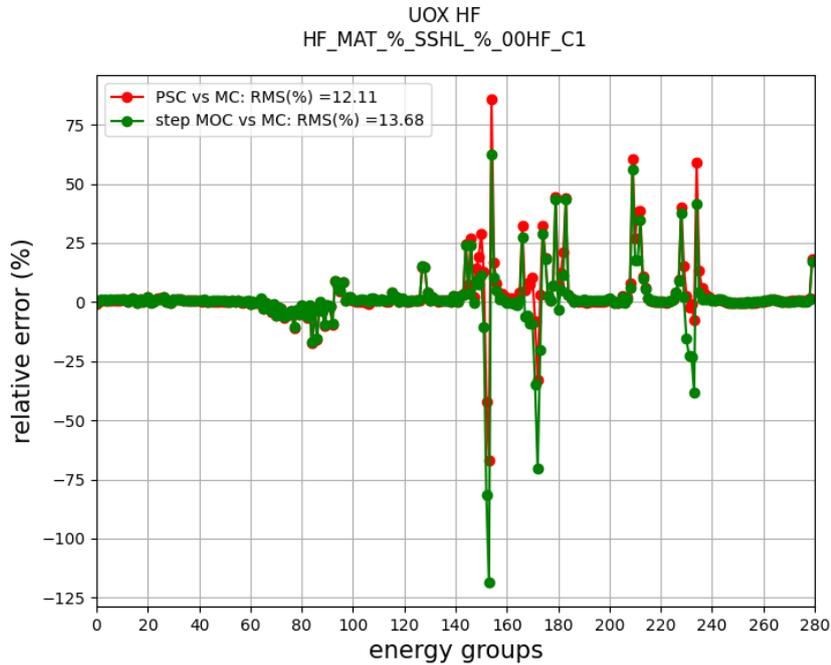


Figure 11.4: UOX-Hf benchmark - PSC vs TRIPOLI-4® continuous-energy Monte Carlo: relative error on the multigroup scalar flux, for the Hf-rod innermost ring.

Such a result, which may appear disappointing at first glance, requires a more careful reading. First of all, it is worth noting again that comparisons with continuous-energy Monte Carlo are usually conducted on few-group reaction rates, rather than on the multigroup scalar

flux.⁹ Then, as mentioned, in all cases, IDT shows numerical accuracy of the same order as APOLLO3®-TDT Step MOC. It is worth reiterating that both IDT and APOLLO3®-TDT Step MOC use the same self-shielding model, number of groups ($G = 281$) and P_N development of the scattering cross section ($N = 3$). Therefore, in order to conduct a more fair analysis, a direct comparison with TDT Step MOC, which represents the golden standard for French industry lattice calculations, is proposed. For the sake of brevity, let us focus our attention on a single case study, that is the one with UOX fuel and AIC control rod. Figure 11.5 compares the prior CSC, LSC and PSC calculations to APOLLO3®-TDT, in terms of RMS relative error (%), on the multigroup scalar flux, per self-shielding region. According to Fig. 11.5, CSC, LSC and PSC show excellent

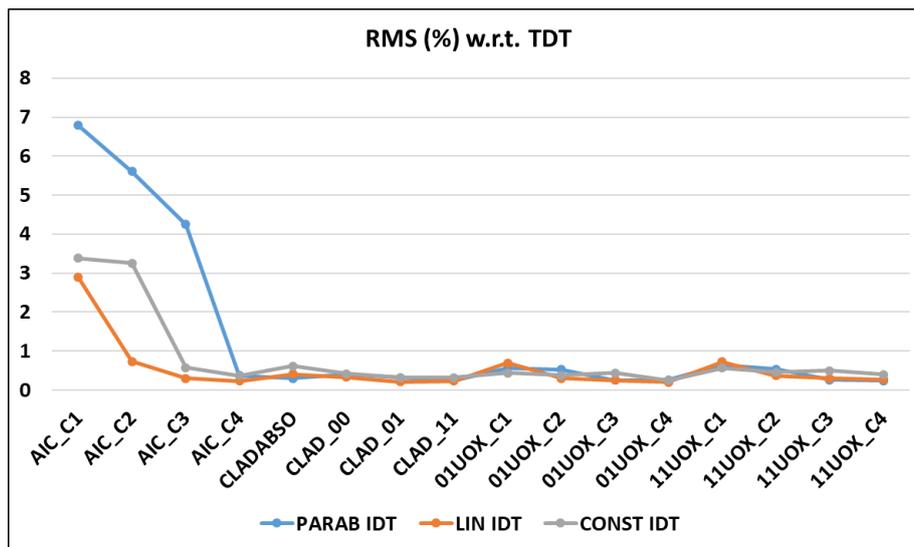


Figure 11.5: RMS relative error on the multigroup scalar flux per self-shielding region, with respect to APOLLO3®-TDT Step MOC.

agreement with TDT, with $RMS < 1\%$, everywhere, except for the three innermost rings of the AIC control rod. Nevertheless, one should notice that CSC, LSC and especially PSC outperform TDT Step MOC, in these specific regions, as depicted in Fig. 11.6, comparing IDT CSC, LSC and PSC and TDT Step MOC to TRIPOLI-4® continuous-energy Monte Carlo.

However, regardless of the particular calculation, it is worth observing that the total multigroup cross section of the first AIC ring can reach values up to approximately 460 cm^{-1} ($g = 190$) and 175 cm^{-1} ($g = 222$), which would require a mesh of the order of 0.01 mm (Fig. 11.7). It is clear that such a level of detail is beyond the modeling capabilities of any code. For the sake of completeness, let us also show the relative error with respect to TRIPOLI-4® of both IDT PSC and APOLLO3®-TDT Step MOC, for the innermost AIC ring, as a function of energy. The largest errors are observed at the cross-section resonances, as it can be seen by cross-checking Figs. 11.7-11.8.

Figures 11.9-11.10 provide the relative error (pcm) on the fundamental eigenvalue, for the entire set of deterministic simulations. Finally, LSC and PSC are compared, in terms of total memory imprint of the angular probability matrices and total CPU time on one thread. It is im-

⁹The choice to draw comparisons in terms of multigroup scalar flux, which may rightly appear unorthodox, is due to the availability of data at the time the study was performed.

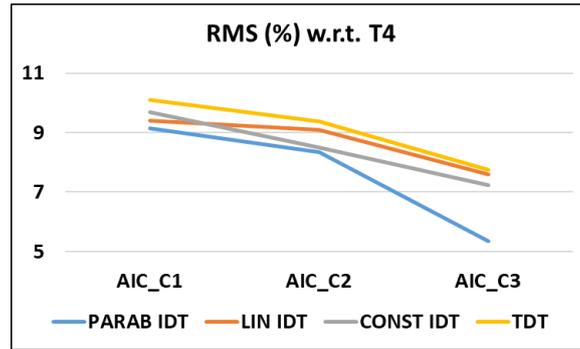


Figure 11.6: RMS relative error on the multigroup scalar flux within the three innermost rings of the control rod, for the UOX-AIC case, with respect to TRIPOLI-4® continuous-energy Monte Carlo.

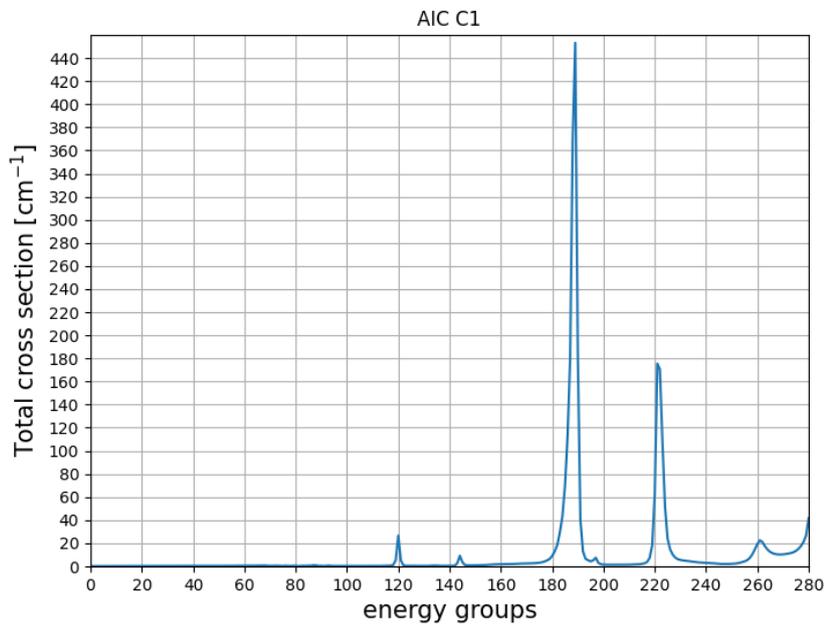


Figure 11.7: Self-shielded total macroscopic cross section of AIC (Ag-In-Cd), within the innermost ring of the control rod, for the 2D 3x3 UOX-AIC benchmark.

portant to mention that the displayed values refer to calculations applying only reciprocity relations to soften the memory occupation and computational time of the CP matrices (Sect.10.4). Hence, the only parameters affecting their size are the spatial mesh and the expansion order (Sect. 11.3). If HCC symmetry properties are not taken into account for probability coefficients evaluation, PSC tends to outperform LSC, due to the coarser spatial grid. In Sect. 11.4.3, we prove that IDT symmetry transformations cannot be applied to PSC, as radial x^2/y^2 -coefficients cannot be swapped.

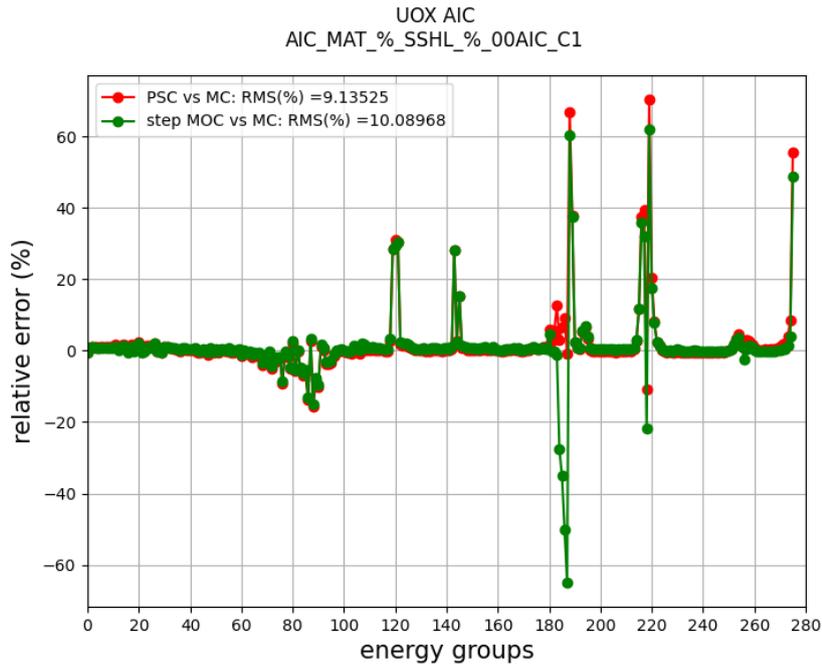


Figure 11.8: UOX-AIC benchmark - PSC and Step MOC vs TRIPOLI-4® continuous-energy Monte Carlo: relative error on the multigroup scalar flux, for the AIC-rod innermost ring.

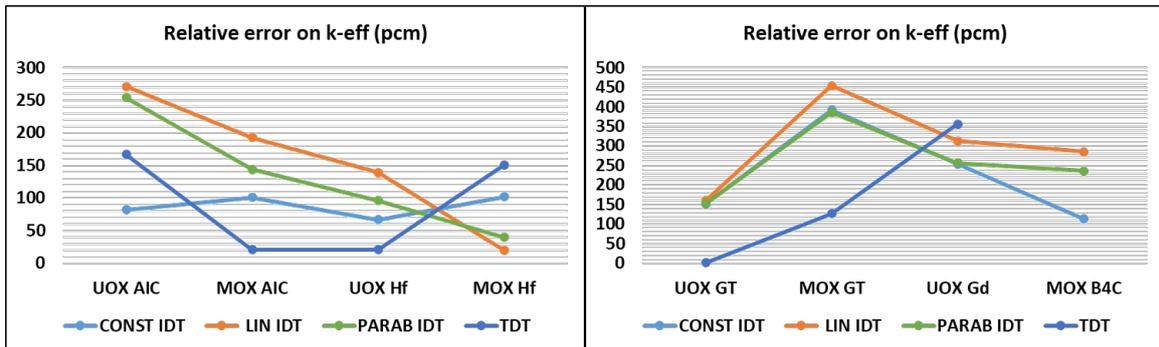


Figure 11.9: Relative error on the fundamental eigenvalue, with respect to TRIPOLI-4® continuous-energy Monte Carlo.

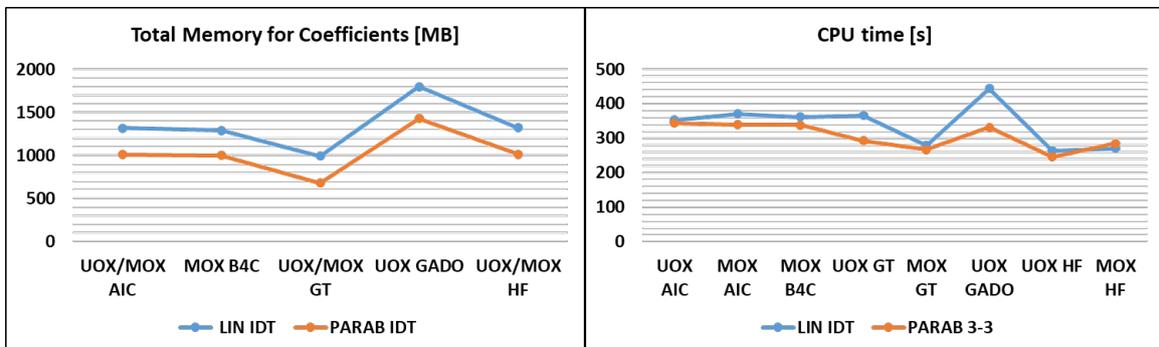
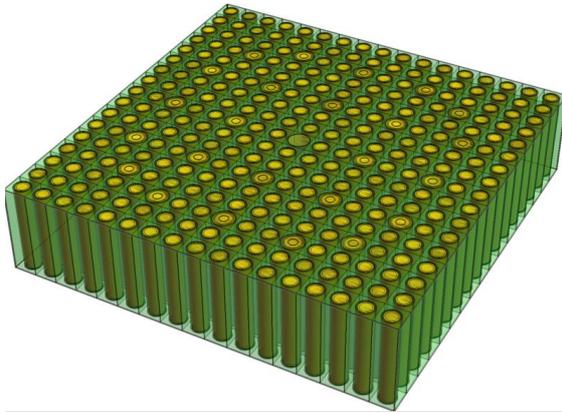


Figure 11.10: PSC vs LSC: total memory footprint of the angular probability matrices and simulation runtime on one thread, provided that only the reciprocity relations are applied. The memory imprint is reduced by 20 – 30%, from LSC to PSC.

11.4.2 . Numerical Accuracy of PSC - 3D Watts Bar Benchmark

The present study is based on a 17x17 3D lattice + water blades, which takes inspiration from one of the problems in the suite of the VERA benchmark, with Pyrex control rods and UOX fuel, [106]. Some structural components, including the bottom and top nozzles, the lower and upper core plates, and the fuel plenum are neglected. The total height is adjusted to 399.662 cm, which comprise lower and upper axial reflectors, measuring 16.951 cm each, and homogenized spacer grids in inconel ($N_{NiFeCr} = 1$, from 16.951 to 20.817 cm) and zircaloy ($N_{Zr} = 6$, of equal size, the first one from 78.295 to 82.105 cm, spaced one from the other by 48.39 cm). The radial geometry is illustrated in Fig 11.11b. Fig. 11.11a depicts an axial slice of the active part of the core. The figure is obtained by Wolfram Mathematica, by means of a script automatically generated by IDT.



(a) 17x17 UOX/Pyrex lattice:
Axial slice between 100 and 105 cm.
Figure generated by IDT.

```

lc
c5 lv
c3 c4 lv
p2 c3 c4 lv
py p1 c3 c4 lv
p2 p1 p1 p2 c4 lv
py p1 p1 py p1 c4 lv
p2 p1 p2 p2 p1 p2 c4 lv
c2 p2 p1 p2 p2 p1 p2 c4 lv
tg c1 p1 py p1 p1 py p1 c4 lv

```

(b) UOX/Pyrex radial geometry: $1/8^{th}$ symmetry.
Legend: c/p = fuel, tg = guide tube, py = pyrex, lv/lc
= water blade.

Figure 11.11: 3D Watts Bar benchmark.

IDT uses fine-structure self-shielded cross sections, with 281 energy groups and P_1 scattering. Two simulations, based on PSC and LSC, respectively, are compared to the Monte Carlo reference, in terms of fission integral, on a $17 \times 17 \times 40$ output grid ($\sigma_{MC} < 1\%$). Both PSC and LSC use $19 \times 19 \times 80 = 28880$ HCCs, with $3 \times 3 \times 2$ surfaces per cell. The solid angle is discretized by S_8 Level-Symmetric angular quadrature.

Figure 11.12 compares LSC (on the left-hand side) and PSC (on the right-hand side) with the Monte Carlo reference, displaying the axial shape of the fission integral distribution and its relative error (%), for a single fuel rod (more precisely, the one located at position $x = 10, y = 9$). In order to compare IDT and TRIPOLI-4®, the fission integral of each fuel rod is normalized to 1. Table 11.3 compares LSC and PSC, in terms of number of regions, relative error on the fundamental eigenvalue, maximum local integral fission error (%) and average RMS relative error (%).

In order to reduce the computational time, IDT resorts to DDM, [75], with hybrid MPI/OpenMP parallelism, over $2 \times 2 \times 80$ subdomains. With 5 MPI processes of 64 threads each, for LSC, the CPU amounts to 6 h, whereas, for PSC, this is 4 times longer, since symmetry relations available in IDT do not hold for the parabolic order, as detailed in Sect. 11.4.3.

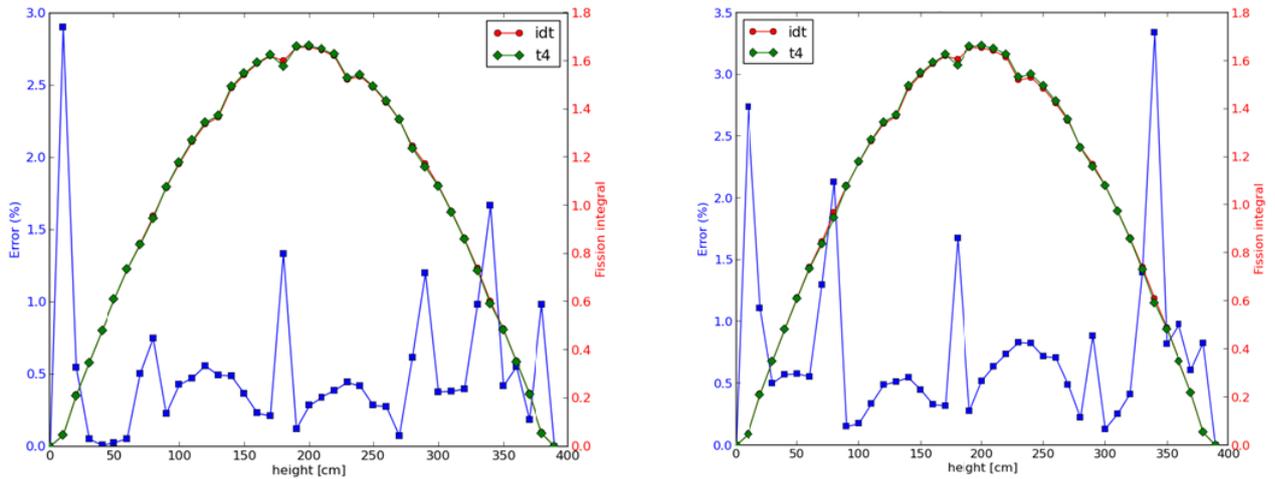


Figure 11.12: LSC (on the left hand side) vs PSC (on the right hand side): fission integral distribution along the z-axis (IDT in red, TRIPOLI-4® in blue) and relative error with respect to TRIPOLI-4®, for the fuel rod located in position $x = 10, y = 9$.

Spatial Approximation	Nb. of Regions	Mean nb. of Regions per cell	Relative error on k_{eff} (pcm)	Max Error (%)	Average RMS (%)
LSC	571073	19.78	83	3.51	0.73
PSC	298699	10.35	94	3.36	1.

Table 11.3: LSC vs PSC: number of computational regions and numerical accuracy. The reference is provided by TRIPOLI-4® continuous-energy Monte Carlo.

11.4.3 . Use of Symmetry Relations

IDT relies on HCC symmetry properties to ease the memory pressure and computational time of the angular probability matrices, by reducing the number of stored directions, as discussed in Sect. 10.4. Let us show that the symmetry relations described by Eqs. (10.46)-(10.59)

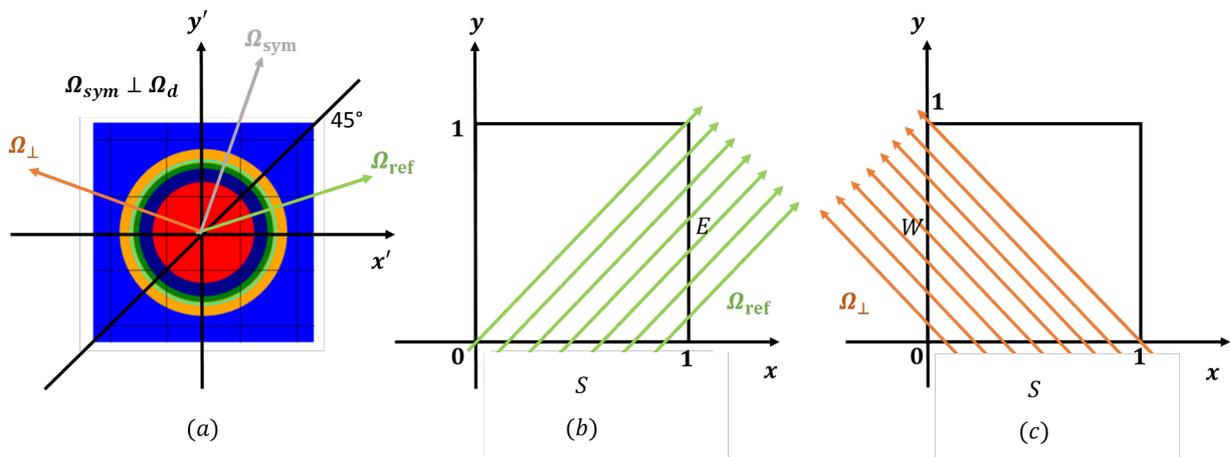


Figure 11.13: Symmetry relations in IDT: on the left hand side, a heterogeneous cell with $1/8^{th}$ symmetry, while, in the middle and on the right hand side, a homogeneous square cell, with unitary side length, illuminated by direction Ω_{ref} oriented along the diagonal and by direction Ω_{\perp} , respectively.

do not hold for PSC, by means of a heuristic proof. For the sake of simplicity, let us consider a homogeneous square cell, with unit cross section and side length, illuminated by direction $\boldsymbol{\Omega}_{ref} \in S_2$ in the first octant, S_2 being the angular quadrature formula, and with 1 surface per cell side (Fig. 10.5.b). The transmission matrix from surface S to E , namely $\underline{\mathbf{T}}_{E,S}(\boldsymbol{\Omega}_{ref})$, where S and E indicate the southern and eastern side of the square, reads as

$$\underline{\mathbf{T}}_{E,S}(\boldsymbol{\Omega}_{ref}) = \underline{\mathbf{M}}^{-1} \int_0^1 dx e^{-y\sqrt{3}} \begin{bmatrix} 1 \\ y - \frac{1}{2} \\ y^2 - \frac{1}{3} \end{bmatrix} \begin{bmatrix} 1 & x - \frac{1}{2} & x^2 - \frac{1}{3} \end{bmatrix}, \quad (11.16)$$

where $\underline{\mathbf{M}}$ is the surface mass matrix of side E . As $x = 1 - y$ (Fig. 11.13.b), Eq. (11.16) becomes

$$\underline{\mathbf{T}}_{E,S}(\boldsymbol{\Omega}_{ref}) = \underline{\mathbf{M}}^{-1} \int_0^1 dy e^{-y\sqrt{3}} \begin{bmatrix} 1 & \frac{1}{2} - y & y^2 - 2y + \frac{2}{3} \\ y - \frac{1}{2} & -(y - \frac{1}{2})^2 & (y - \frac{1}{2})(y^2 - 2y + \frac{2}{3}) \\ y^2 - \frac{1}{3} & (y^2 - \frac{1}{3})(\frac{1}{2} - y) & (y^2 - \frac{1}{3})(y^2 - 2y + \frac{2}{3}) \end{bmatrix}. \quad (11.17)$$

Let us now consider the direction $\boldsymbol{\Omega}_\perp \perp \boldsymbol{\Omega}_{ref}$ and the transmission matrix from surface S to W , namely $\underline{\mathbf{T}}_{W,S}(\boldsymbol{\Omega}_\perp)$, where W indicates the western side (Fig. 11.13.c). One can easily prove that the surface mass matrix has not changed. Hence, the transmission matrix is given by

$$\underline{\mathbf{T}}_{W,S}(\boldsymbol{\Omega}_\perp) = \underline{\mathbf{M}}^{-1} \int_0^1 dx e^{-y\sqrt{3}} \begin{bmatrix} 1 \\ y - \frac{1}{2} \\ y^2 - \frac{1}{3} \end{bmatrix} \begin{bmatrix} 1 & x - \frac{1}{2} & x^2 - \frac{1}{3} \end{bmatrix}. \quad (11.18)$$

As $x = y$ (Fig. 10.5.c), one obtains

$$\underline{\mathbf{T}}_{W,S}(\boldsymbol{\Omega}_\perp) = \underline{\mathbf{M}}^{-1} \int_0^1 dy e^{-y\sqrt{3}} \begin{bmatrix} 1 & y - \frac{1}{2} & y^2 - \frac{1}{3} \\ y - \frac{1}{2} & (y - \frac{1}{2})^2 & (y - \frac{1}{2})(y^2 - \frac{1}{3}) \\ y^2 - \frac{1}{3} & (y^2 - \frac{1}{3})(y - \frac{1}{2}) & (y^2 - \frac{1}{3})(y^2 - \frac{1}{3}) \end{bmatrix}. \quad (11.19)$$

One can notice that Eqs. (11.17)-(11.19) satisfy Eqs. (10.47)-(10.51) only up to linear order. In fact, the terms introduced by PSC are sensitive to the change of variable, making the generalization of the prior symmetry relations not possible for parabolic short characteristics. Similar considerations apply to the other angular probability matrices. This problem has not been resolved yet and will be addressed in the near future.

11.5 . MIXED-ORDER SHORT CHARACTERISTICS

Although PSC allows for a major reduction of the number of regions (Sects. 11.4.1-11.4.2), IDT is not equipped, at the time being, with a set of symmetry relations that permit decreasing the memory requirements of the second-order probability coefficients. Nevertheless, starting from PSC, a hybrid approach may be tested, based on linear-order expansions on the radial plane and parabolic development along the z -axis (HSC). More precisely, the polynomial bases may be set equal to

$$P_\alpha(\mathbf{r}) = [1, \mathbf{r} - \mathbf{c}_\alpha, z^2 - d_{\alpha,z}^2]^T, \quad (11.20)$$

$$P_s(\mathbf{r}) = [1, \mathbf{r} - \mathbf{c}_s]^T, \mathbf{n}_s \cdot \mathbf{e}_z = 0, \quad (11.21)$$

$$P_s(\mathbf{r}) = [1, \mathbf{r} - \mathbf{c}_s, z^2 - d_{s,z}^2]^T, \mathbf{n}_s \cdot \mathbf{e}_z = \pm 1. \quad (11.22)$$

The reason is twofold: on the one hand, the possibility to recover transformations (10.46)-(10.59) on the radial plane, as linear components may be swapped, and, on the other, the possibility to coarsen the axial mesh, and thus the memory imprint of the angular coefficients, thanks to the additional parabolic term along the z -axis. This approach has been investigated within the frame of PSC, by setting the second-order x/y coefficients equal to zero. Note that the memory imprint is lowered by a factor 8, as only directions with $\varphi \in [0, \frac{\pi}{4}]$ and $\theta \in [-\frac{\pi}{2}, \frac{\pi}{2}]$ are stored, φ and θ being the azimuthal and polar angle, respectively (Fig. 10.8). In the following, a comparison between HSC and LSC is proposed, in terms of number of regions and numerical accuracy, for the C5G7 benchmark, [29].

11.5.1. Numerical Accuracy of HSC - 3D C5G7 Benchmark

The present analysis compares HSC and LSC for the Rodded A, Rodded B and Unrodded configurations of the C5G7 benchmark (Sect. 7.6.1 and Sect. 8.3). The outer grid comprises $51 \times 51 \times 9$ HCCs, of 7.14 cm each along the z -axis. Each HCC is equipped with a number of inner z -planes, depending on the expansion order and the physical discontinuities of the benchmark. The total number of regions is about $6.3 \cdot 10^4$, for LSC, and $3.9 \cdot 10^4$, for HSC. Each cell boundary is subdivided into $3 \times 3 \times 4$ surfaces. The angle is discretized by a S_8 Level-Symmetric angular quadrature formula.

Tab. 11.4 provides the minimum, maximum and RMS relative errors on the power distribution of LSC and HSC, with respect to the MCNP code, [83], for each configuration of the C5G7 benchmark. The maximum reactivity errors amount to 31 pcm (Unrodded case) for HSC and 25 pcm (Rodded B case) for LSC. By using DDM, over $10 \times 10 \times 9$ subdomains, with 15 processes of 64 threads each, the CPU time is reduced to about 7 minutes for LSC. For HSC, a reliable estimate of the simulation time is not available yet, due to the calculation of unnecessary moments, as mentioned in Sect. 11.5.

Error	HSC (LSC) / A - Layer 1	HSC (LSC) / A - Layer 2	HSC (LSC) / A - Layer 3
Min (%)	-1, 283 (-1, 227)	-1, 217 (-1, 203)	-1, 137 (-1, 326)
Max (%)	0, 615 (0, 662)	0, 629 (0, 669)	1, 316 (1, 213)
RMS (%)	0, 313 (0, 296)	0, 322 (0, 322)	0, 474 (0, 449)
Error	HSC (LSC) / B - Layer 1	HSC (LSC) / B - Layer 2	HSC (LSC) / B - Layer 3
Min (%)	-1, 408 (-1, 365)	-1, 295 (-1, 305)	-1, 642 (-1, 883)
Max (%)	0, 643 (0, 688)	1, 211 (1, 158)	1, 641 (1, 571)
RMS (%)	0, 404 (0, 387)	0, 446 (0, 433)	0, 596 (0, 608)
Error	HSC (LSC) / U - Layer 1	HSC (LSC) / U - Layer 2	HSC (LSC) / U - Layer 3
Min (%)	-1, 011 (-0, 983)	-1, 244 (-1, 217)	-1, 538 (-1, 725)
Max (%)	0, 61 (0, 644)	0, 75 (0, 759)	1, 033 (0, 904)
RMS (%)	0, 292 (0, 287)	0, 3 (0, 298)	0, 35 (0, 397)

Table 11.4: Rodded A (= A), Rodded B (= B) and Unrodded (= U) configurations.

11.6 . CONCLUSIONS

This chapter has introduced the method of incomplete parabolic short characteristics, showing encouraging preliminary results. Despite the significant decrease in the number of computational regions, the total memory imprint of the angular probability matrices is still larger than linear characteristics. New symmetry relations have to be constructed, in order for PSC to be competitive with LSC. In the meantime, a hybrid linear/parabolic approach has been tested, showing a numerical accuracy of the same order as LSC and lowering the memory imprint by a factor 4, with respect to PSC. In the near future, this mixed-order method will be implemented in IDT, in order to optimize the CPU time, numerical accuracy and memory occupation.

Part IV

MULTIPHYSICS ITERATIONS IN IDT

12 - NEUTRONICS/THERMAL-HYDRAULICS COUPLING

12.1 . LIAISON AND OBJECTIVES

In Chaps. 7-11, the major limits of the past version of IDT have been addressed, providing a new short-characteristics solver, which is able to treat a larger set of geometries, with increased numerical accuracy and lower memory requirements. Applications of IDT both to two-dimensional and three-dimensional patterns or assemblies have been discussed in Sects. 7.6.1, 7.6.2, 7.7.2, 8.5, 9.4.1, 9.4.2, 11.4.1, 11.4.2, where comparisons with TRIPOLI-4® and APOLLO3® have been drawn for code validation. In this chapter, the use of IDT for detailed neutron transport, on 'quasi-industrial' real-sized 3D calculations, is proposed, including, for the time being, the sole thermal feedback. More specifically, the final scope is to outline a novel numerical pathway, using IDT, together with THEDI (App. A), for coupled neutronics/thermal-hydraulics simulations, in exact 3D geometry, with no cross-section homogenization.

12.2 . INTRODUCTION

A fission reactor is a multiphysics system, involving several coupled problems, including neutron transport, thermal-hydraulics, fuel-to-coolant heat transfer, thermal-mechanics and corrosion, to name a few. The detailed modelling of every physical process, by a dedicated high-fidelity solver, may require large computational resources. Often, only a subset of physical problems is taken into account, to analyse their interdependencies, thus neglecting the effect of the other processes. In this respect, fission reactor physicists usually focus on neutronics, thermal-hydraulics and fuel-to-coolant heat transfer, due to their strong mutual exchanges, in terms of generated nuclear power and subsequent temperature feedback on the neutron cross-sections. From this standpoint, priority may be given to neutronics (later abbreviated as N), which is solved by fine multigroup neutron transport, whereas thermal-hydraulics and heat transfer (hereinafter referred to as thermal-hydraulics or TH, for brevity's sake) are approached by a simplified model, based on a set of assumptions, to reduce the overall computational cost.¹⁰ Such an objective may be pursued by two opposite strategies, namely the serial integration and the parallel processing coupling [107]. The former consists in implementing (ex novo) a TH module into the neutron transport code, and, as such, may be less convenient. Conversely, the other approach typically requires minor programming efforts, as merely performing data exchange from one code to another.

State-of-the-art reactor core calculation schemes generally implement N/TH parallel processing coupling. Despite the increased availability of computational resources, two-step techniques are still preferred to 3D direct transport (Sect. 5.2-5.3), as less computationally expensive. As mentioned in Chap. 5, the standard two-step approach uses deterministic transport to generate lattice-homogenized few-group cross-section libraries, to be employed in 3D nodal diffusion or simplified transport. At the time being, the two-step strategy is the most widespread technique for reactor calculations.

¹⁰The opposite may also apply, depending on the application domain. For instance, THEDI [77] and CATHARE [108] are equipped with a simplified neutron transport model, based on point kinetics.

Notwithstanding the unquestioned merits of conventional two-step schemes and their role in current industrial simulations, some limitations may be pointed out:

- The lattice calculation is generally conducted under the assumption of axially infinite assemblies, which may be a poor approximation, as in the case of sodium-cooled fast reactors, since they exhibit strong heterogeneities along the z-axis [109];
- Along the lattice calculation, assemblies are assumed to be periodically infinite arrays, thus requiring artificial critical-leakage models to force the eigenvalue to converge to 1 (Sect. 5.2.1);
- The two-step approach is based on dimensionality reduction. The idea is to generate, by means of lattice calculations, homogenized cross sections, to feed into the core simulation. This, in turn, provides the neutron flux and reaction rate distributions over the whole reactor. Generally, flux-weighted cross sections from the first step are not able to reproduce accurate reaction rate distributions, showing poor agreement with the reference solution. For this reason, an intermediate step, based on equivalence theory, with superhomogénéisation (SPH) or flux discontinuity ratios, may be necessary to improve the results [110];
- Homogenized cross sections depend not only on the whole set of parameters (Sect. 5.2), but also on the neutron flux, and, as a consequence, on the whole behaviour of the reactor (Sect. 1).

In this chapter, a new simulation pathway is proposed, for coupled N/TH calculations, implementing no dimensionality reduction, in exact 3D heterogeneous core geometry. As discussed in Sect. 12.3-12.4, cross-sections do not undergo flux-weighting homogenization. As a result, they show milder dependence on the physical parameters (Sect. 12.4.2-12.4.3) and may be easily interpolated, for each reactor state. In order to perform multiphysics simulations, the thermal feedback is included within the outer iteration loop, where the TH field is provided by an external code (THEDI, App. A). Priority is given to the neutronic calculation, which is solved by multigroup discrete-ordinates neutron transport, where MOSC is applied onto heterogeneous Cartesian nodes (IDT). This generates a pin-by-pin power-map (Sect. 12.5) that is provided to the TH solver, which, in turn, calculates the temperature and density distribution, under simplified modelling assumptions. These include independent stationary TH sub-channels, exchanging neither heat nor mass, under monophasic flow conditions (App. A). A dedicated python script allows for the exchange of data (12.7.14), between IDT and THEDI, and updates the input file, for microscopic cross-section interpolation. The latter is performed by a capability of XSTOOL, which generates ASCII files, containing the nuclear data for IDT usage. Multigroup iterations allow to update the transport spectrum, where Neumann iterations are used to treat cross-section variations (Sect. 9.2), due to Doppler broadening and volume dilation. The transport solution is used to construct a synthetic operator, based on 3D CMFD diffusion, with fewer energy groups and homogenized pin-cells (Sect. 4.7.2). The resulting eigenpair permits to update the eigenvalue and renormalize the angular flux distribution. In this framework, differently from standard two-step calculations, no critical-leakage model is needed to account for fuel-assemblies finite size.

The chapter is structured as follows. First, a brief introduction is proposed on effective macroscopic cross section generation (Sect. 12.3). In this context, an approach purely based on

microscopic cross-section interpolation, with no prior homogenization, is discussed (Sect. 12.3.4). The presented strategy is supported by a comprehensive study, showing the dependence of the parametric cross-section library on burn-up and fuel temperature (Sect. 12.4). In Sect. 12.7, the practical algorithm allowing for coupled N/TH simulations is analysed in detail. Finally, results are presented in Sect. 12.8.

12.3 . EFFECTIVE MACROSCOPIC CROSS-SECTION GENERATION

12.3.1 . Multigroup approximation and self-shielding

As mentioned in Sect. 4.4, all deterministic methods integrate the Boltzmann equation over energy intervals of finite length (i.e., the energy groups). The transport problem transforms as in Eq. (4.29), where

$$\Sigma^g(\mathbf{r}, \boldsymbol{\Omega}) = \frac{\int_g \Sigma(\mathbf{r}, E) \psi(\mathbf{r}, \boldsymbol{\Omega}, E) dE}{\int_g \psi(\mathbf{r}, \boldsymbol{\Omega}, E) dE}, \quad g = 1, \dots, G \quad (12.1)$$

defines the multigroup cross section, depending on all phase-space variables, including the neutron direction, which is later neglected. Note that this artificial quantity depends on the angular flux, i.e., the solution of the Boltzmann equation, which is unknown. Equation (12.1) is approached by subdividing the multigroup microscopic cross sections in two categories:

- The non-resonant (or infinite-dilution) cross sections, namely

$$\sigma_j^{g,\infty} = \frac{\int_g \sigma_j(u) w(u) du}{\int_g w(u) du}, \quad (12.2)$$

where j indicates a nuclear isotope and w is an optimal weight, generally defined as a representative spectrum of the whole reactor. $u = \ln(\frac{E_0}{E})$ denotes the neutron lethargy, with E and E_0 being the neutron energy and the maximum energy of the neutron spectrum, respectively. This formula applies to all light isotopes, as their resonances appear at energies $E > E_0$. Note that Eq. (12.2) is independent of the position.

- The resonant cross section, namely

$$\sigma_j^{g,e} = \frac{\int_g \int_{D_e} \sigma_j(\mathbf{r}, u) \hat{\psi}(\mathbf{r}, u) d^3r du}{\int_g \int_{D_e} \hat{\psi}(\mathbf{r}, u) d^3r du}, \quad (12.3)$$

where $\hat{\psi}(\mathbf{r}, u)$ is an approximate solution of Eq. (4.29). The spatial variable is somehow retained, as $\sigma_j^{g,e}$ is region-dependent. The evaluation of $\hat{\psi}(\mathbf{r}, u)$ is addressed by approximate transport models, defining the so-called Self-Shielding (SSH) calculation, where the geometry is generally collapsed into one or two dimensions and neutron physics is simplified. Cross sections defined by Eq. (12.3) are therefore also referred to as effective or self-shielded cross sections, whose determination requires quadrature formulas, as detailed in [111]. In the following, the effective microscopic cross sections are expressed as

$$\sigma_j^{g,e} = \mu_j^{e,g} \sigma_j^{g,\infty}, \quad (12.4)$$

where $\mu_j^{e,g}$, $g = 1, \dots, G$, represent the self-shielding factors.

The ensemble of approximations introduced by the Self-Shielding models represents an inherent limitation common to all deterministic simulations, affecting the accuracy of the final numerical output.

12.3.2 . Thermal feedback

Macroscopic cross sections are temperature-dependent quantities through

- Mass density variation, induced by a change of temperature, which, in turn, affects the average number of nuclides per unit volume;
- Thermal agitation, which causes an apparent change in the resonant cross section of a neutron, due to the relative motion between the projectile (i.e., the neutron itself) and the target nucleus. This phenomenon goes under the name of Doppler effect and is responsible for a change in the shape of a resonance, which tends to broaden if the temperature increases, while lowering its maximum value.

As mentioned in Sect. 12.3.1, light isotopes, such as the hydrogen and oxygen, show resonances only at energies of the order of few MeV. Hence, from a practical standpoint, the moderator/coolant of LWRs, i.e., water, is not impacted by the Doppler effect. On the contrary, the change in density caused by a temperature variation generally has a significant effect on moderator macroscopic cross sections and, as a consequence, on the reactivity of the multiplying system.

Conversely, nuclear fuel is generally less sensitive to density variations, the change in the macroscopic cross sections being mostly due to Doppler broadening. This introduces a negative feedback, in case of temperature increase, which lowers the reactivity of the reactor.

12.3.3 . Standard two-step reactor calculations

As illustrated in Sect. 5.2, industrial reactor core calculations are generally conducted in two steps. First, detailed neutron transport is computed on a set of fuel-assembly types, under reflective boundary conditions. As steady-state reactors, with negligible neutron sources, are critical, every fuel-assembly is furtherly processed, by applying a critical leakage model (Sect. 5.2.1), until convergence (i.e., $k_{eff} = 1$). Note that the flux calculation requires Eq. (12.2) and Eq. (12.3), which are respectively provided by an infinite-dilution nuclear database and SSH cross-section libraries. The calculation is repeated for a variety of state-points, \mathbf{p}_n , $n = 1, \dots, N_{states}$, each corresponding to a set of physical parameters, namely

$$\mathbf{p} = \{T_f, T_w, \rho_w, F/A_{conf}, C_B, BU, t\}, \quad (12.5)$$

where

- T_f is the fuel temperature;
- T_w is the moderator temperature;
- ρ_w is the moderator mass density;
- F/A_{conf} indicates the fuel-assembly configuration (e.g., rodged/unrodged);
- C_B denotes the boron concentration;
- BU represents the burn-up;
- t is the time.

The lattice calculation generates a library of homogenized and condensed cross sections, depending on the aforementioned parameters, which synthesize the behaviour of the reactor under several operating conditions. In the second step (Sect. 5.2.2), the reactor core is calculated by few-group 3D diffusion or simplified transport, using the parametric cross-section library, which is interpolated, in order to meet the actual state of the core.

Standard two-step simulations are affected by a number limitations. In particular,

- Neutron currents between fuel-assemblies are neglected. In order to account for the criticality condition, while preserving specular reflection on the boundary, an approximate model is introduced, with simplified neutron flux distribution (Eq. 5.1);
- The output of the lattice calculation is a homogenized parametric library, where the cross-sections encompass the whole reactor behaviour, under a wide variety of operating conditions. Cross sections feeding the core calculation are thus affected by an intrinsic error, of difficult estimation.

12.3.4 . Advanced two-step reactor calculations

An alternative to the standard two-step approach consists in interpolating the infinite-dilution and the self-shielded cross sections, without resorting to spatial homogenization beforehand. By neglecting the energy group index, one has

$$\sigma_j^\infty(T) = \sum_k f(T; T_k) \sigma_j^\infty(T_k), \quad (12.6)$$

$$\sigma_j^e(T, \mathbf{p}) = \sum_{l,n} g(T, \mathbf{p}; T_l, \mathbf{p}_n) \sigma_j^e(T_l, \mathbf{p}_n), \quad (12.7)$$

where

- T_k is a temperature point in the external reference library;
- T_l and \mathbf{p}_n are respectively a temperature and a state point in the parametric library of self-shielding factors, $\mu_j^e(T_l, \mathbf{p}_n)$;
- $f(T; T_k)$ is a linear-tent interpolating function for the temperature point T_k , defined as

$$f(T; T_k) = \begin{cases} \frac{T - T_{k-1}}{T_k - T_{k-1}}, & T \in [T_{k-1}, T_k], \\ \frac{T_{k+1} - T}{T_{k+1} - T_k}, & T \in [T_k, T_{k+1}], \\ 0, & \text{otherwise}; \end{cases} \quad (12.8)$$

- $g(T, \mathbf{p}; T_l, \mathbf{p}_n)$ is the interpolating function for the temperature T_l and the state point \mathbf{p}_n , generally defined as

$$g(T, \mathbf{p}; T_l, \mathbf{p}_n) = \begin{cases} 1, & T = T_l \text{ and } \mathbf{p} = \mathbf{p}_n, \\ 0, & T = T_{l' \neq l} \text{ or } \mathbf{p} = \mathbf{p}_{n' \neq n}; \end{cases} \quad (12.9)$$

whose expression is indicated in Sect. 12.3.5.

- $\sigma_j^e(T_l, \mathbf{p}_n) = \mu_j^e(T_l, \mathbf{p}_n) \sigma_j^\infty(T_l)$, where $\sigma_j^\infty(T_l)$ is interpolated as in Eq. (12.6).

Note that the self-shielding factors are not directly interpolated. The macroscopic cross-section library for the medium m is obtained as follows:

$$\Sigma_m = \Sigma_m^\infty + \Sigma_m^e = \sum_{j \in m^\infty} c_j \sigma_j^\infty(T) + \sum_{j \in m^e} c_j \sigma_j^e(T, \mathbf{p}), \quad (12.10)$$

where the first and second term account for the infinite-dilution and the resonant isotopes, respectively.

In the present work, the interpolation of the cross section and the generation of the parametric library are performed by XSTOOL, of which a brief introduction is proposed in Sect. 12.3.5.

12.3.5 . A brief introduction to XSTOOL

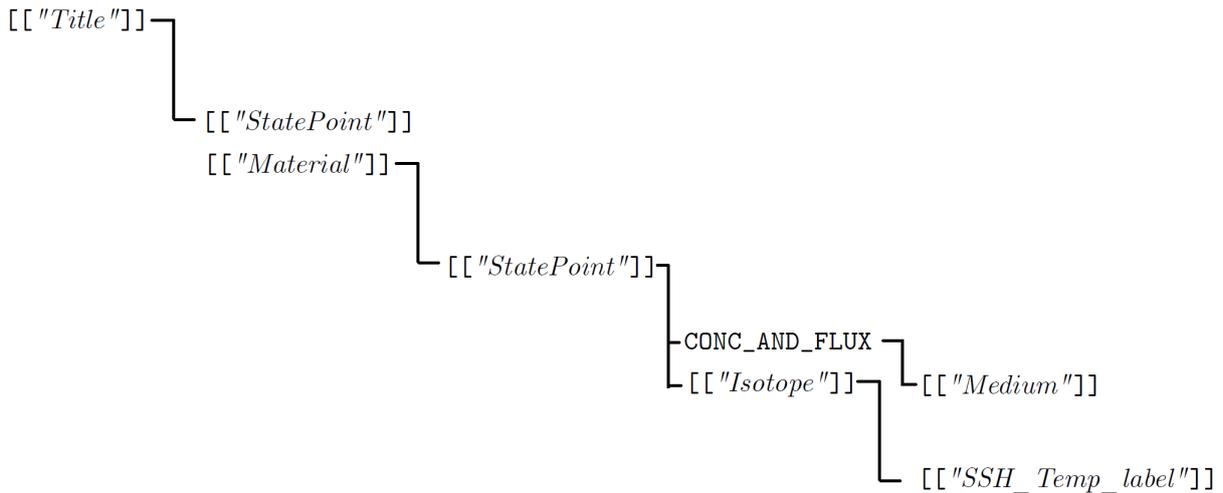
XSTOOL (*Cross-Section Tool*) is a mixed C++/FORTRAN software performing cross-section interpolation (and condensation, if required) and providing multigroup effective cross-section libraries, which can be used in production codes, as IDT. It resorts to a capability of APOLLO3® to generate, at each burn-up step of a lattice calculation, a snapshot of the self-shielded cross sections, material flux distributions and concentrations, for all resonant isotopes of the simulation. These data are written and stored in a compressed form over a hierarchical HDF file, at the reference energy mesh, composed by 281 or 363 energy groups.

This file, namely the **External Parametric Library of Self-Shielding Factors** (EPL-SSF), is provided by APOLLO3®, using a command, namely `WriteExternalLibrary`, developed in this work. The hierarchical structure of the EPL-SSF file is represented schematically in Fig. 12.1. This organization in a single data-base is designed in a way that allows easy access to the self-shielding factors μ , stored at each state-point calculation.

The foreseen applications of the EPL-SSF file are as follows:

- **Multi-physics simulations**, with the neutron/photon transport equation receiving the thermal-hydraulic feedback on cross sections, for critical-boron or time-dependent calculations, performed by a deterministic solver;
- **Radiation shielding simulations**,
 - as built-in neutron multigroup cross-section library for IDT, in the framework of the TRIPOLI-4® CADIS variance-reduction technique;
 - as built-in neutron multigroup cross-section library for forward and adjoint stand-alone deterministic simulations;
- **Storage of concentrations, flux and self-shielding factors**, for reference reaction rates reconstruction, at any state-point within APOLLO3®;
- **Restart of an APOLLO3® simulation**, with already-stored self-shielded cross sections, isotopic concentrations and flux per medium;
- **Re-homogenization** of perturbed configurations, with already-stored nominal flux.

The objectives of the EPL-SSF file are as follows:



- ["Title"] ➡ UDM title
- ["Material"] ➡ UDM material
- ["StatePoint"] ➡ name of the state-point (internal build)
- CONC_AND_FLUX ➡ isotopic concentration and flux container per medium
- ["Medium"] ➡ name of medium (CDM name function)
- ["Isotope"] ➡ UDM isotope
- ["SSH_Temp_label"] ➡ self-shielding label plus temperature label

Figure 12.1: Hierarchical structure of the External Parametric Library of Self-Shielding Factors (EPL-SSF) (in the figure, UDM is an acronym for the User Data Model of APOLLO3®).

- The construction of a **parameterized data-base of physical self-shielding factors**, for a variety of resonant isotopes, for different physical configurations per state point;
- The storage of the **isotopic concentrations** per medium and state point;
- The storage of the **average multigroup scalar flux** per medium of the reference calculation, for each state point;
- The storage of the **self-shielding data**, in particular, the effective temperature of the isotope, the energy range of the self-shielding model and the relative back-ground multigroup cross section.

The library architecture is organized according to the following guidelines:

- The structure of the EPF-SSF file is based on Hierarchical Data Format (HDF);
- The library stores the reference self-shielding factors, for any resonant isotope, for five reactions of interest. These reactions are: **absorption, capture, P_0 diffusion, fission and fission production**;
- The library contains a complete set of data to allow for the reconstruction of reference isotopic reaction rates, at any state point present in the library;
- The library has a physical-like organization: the isotopes are stored within material directories, for each state point and self-shielding label;

- All the key-words used as data-base directory names are constructed by retrieving the names for materials, isotopes and geometrical-zone labels, assigned by the user, in the APOLLO3® input file;
- The library has no knowledge of the geometry;
- Several lattice dataset can share the same file, by using different titles.

In order to save memory and access time, the EPL-SSF file complies with the following specifications:

- The library contains only resonant isotopes of the materials, i.e., the isotopes that are not self-shielded are not stored;
- The self-shielding factors are stored only in the energy range of the self-shielding model.

XSTOOL can handle Cartesian grids as well as sparse grids of state-points. In case of Cartesian grids, the interpolating functions, $g(T, \mathbf{p}; T_l, \mathbf{p}_n)$, are multilinear tent-functions. Otherwise, in case of unstructured grids, $g(T, \mathbf{p}; T_l, \mathbf{p}_n)$ are radial basis kernel functions, namely

$$g(\mathbf{x}; \mathbf{x}_k) = \exp\left(-\frac{\|\mathbf{x} - \mathbf{x}_k\|^2}{2\sigma^2}\right), \quad (12.11)$$

with $\mathbf{x} = (T, \mathbf{p})$ and $\mathbf{x}_k = (T_l, \mathbf{p}_n)$, while σ is defined as the standard deviation of the dataset. Alternatively, for sparse grids, the user can use a quadratic-polynomial kernel function of the kind

$$g(\mathbf{x}; \mathbf{x}_k) = \lambda_k(\mathbf{x} \cdot \mathbf{x}_k + d)^2, \quad (12.12)$$

with d as a regularization parameter, set as the average cosine across all points of the dataset, and λ_k as the normalization factor, i.e., $\lambda_k = \frac{1}{(\mathbf{x}_k \cdot \mathbf{x}_k + d)^2}$.

XSTOOL requires three inputs, namely the external reference library (GALILEE, [112]), the parametric SSH library (EPL-SSF) generated by APOLLO3® and an input file of the kind illustrated in Algo. 4. In particular,

Algorithm 4 : XSTOOL input.

- 1: Read macroscopic XS library output path
 - 2: Read anisotropy order, N
 - 3: Read number of energy groups, H
 - 4: **for** medium = 1 \rightarrow ∞ **do**
 - 5: **if** End command **then**
 - 6: Return
 - 7: **end if**
 - 8: Read [$mat + e + c$; T ; T_f ; T_w ; C_B ; BU ; $\%CR$; ρ_w ; t]
 - 9: **end for**
-

- mat is a label identifying the material;
- e is the SSH label;

- c represents the HCC order number;
- T is the local temperature;
- $T_f, T_w, \rho_w, C_B, BU$ and t are indicated in Sect. 12.3.3;
- $\%CR$ quantifies the control-rod insertion fraction (%) in c .

The output is the interpolated macroscopic cross-section library, which contains H energy groups and P_N -scattering matrices, for every $mat + e + c$ string detected in Algo. 4.

12.4 . PARAMETRIC EFFECTIVE MICROSCOPIC CROSS-SECTION ANALYSIS

In the following, a preliminary study on the dependence of parametric effective microscopic cross-sections on burn-up and fuel temperature is proposed. In Sect. 12.4.1, the benchmark is introduced. Section 12.4.2 shows that microscopic cross sections exhibit mild variations with respect to burn-up, thus suggesting the possibility to reduce the number of burn-up steps and, as consequence, the memory occupation. In Sect. 12.4.3, a factorization is considered for the dependence on burn-up and fuel temperature and the numerical error is evaluated, by comparison with the reference microscopic cross sections.

12.4.1 . Test Case Specifications

Let us consider a 17x17 2D UOX/Gd assembly, with 3.2% enrichment and 32 Gadolinium poisoned pin cells. The external reference library is provided by GALILEE and contains 281 energy groups. The parametric library of SSH-factors is generated by APOLLO3®, by Multi-Cell Fine-Structure Self-Shielding. For the sake of brevity, only the results obtained for the UOX-cell inner ring and the UOX/Gd-cell outer ring are presented.

12.4.2 . Dependence on burn-up

Let us define the average microscopic cross section of isotope j and group g , namely

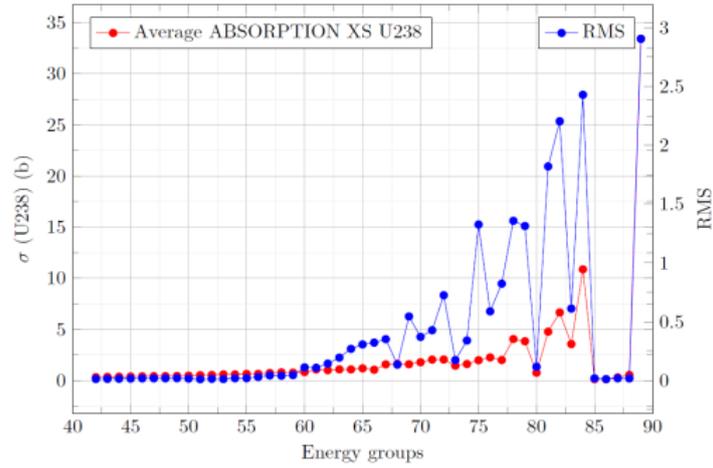
$$\bar{\sigma}_j^g = \frac{1}{N_e N_{BU}} \sum_n \sum_e \sigma_j^{g,e}(BU_n), \quad (12.13)$$

and the associated RMS,

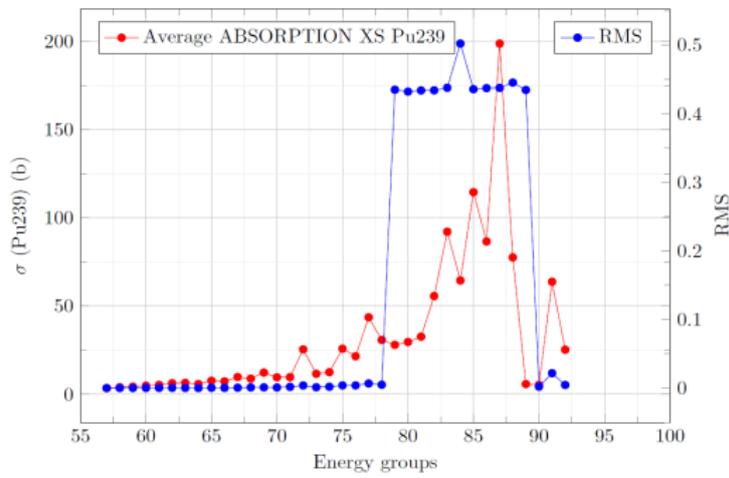
$$RMS_j^g = \frac{1}{N_e N_{BU}} \sqrt{\sum_n \sum_e (\sigma_j^{g,e}(BU_n) - \bar{\sigma}_j^g)^2}, \quad (12.14)$$

where N_{BU} is the number of burn-up steps and N_e is the number of rings containing the isotope. Note that all the other parameters are fixed and are not explicitly indicated, for the sake of conciseness. Figures 12.2a-12.2b compare the first and second-order moments (Eqs. (12.13)-(12.14), respectively), of ^{238}U and ^{239}Pu , for all self-shielded energy groups, i.e., all groups within $g_2 = 42$ and $g_1 = 89$, for ^{238}U , and all groups within $g_2 = 57$ and $g_1 = 94$, for ^{239}Pu . Generally, $\bar{\sigma}_j^g \gg RMS_j^g$, hence $\sigma_j^{g,e}(BU)$ varies slightly around its mean value.

Figures 12.3-12.4 display the effective microscopic absorption cross section of ^{238}U and ^{239}Pu , for the innermost and outermost rings of UOX and UOX/Gd, as a function of burn-up, for all energy groups within g_2 and g_1 (defined as above). In both cases, the change in the effective microscopic cross section is a weak function of BU , thus these trends can be easily interpolated



(a) Average microscopic absorption cross section of ^{238}U vs RMS, for groups between $g_2 = 42$ and $g_1 = 89$.



(b) Average microscopic absorption cross section of ^{239}Pu vs RMS, for groups between $g_2 = 57$ and $g_1 = 92$.

Figure 12.2: Average microscopic absorption cross section vs RMS for two fuel isotopes.

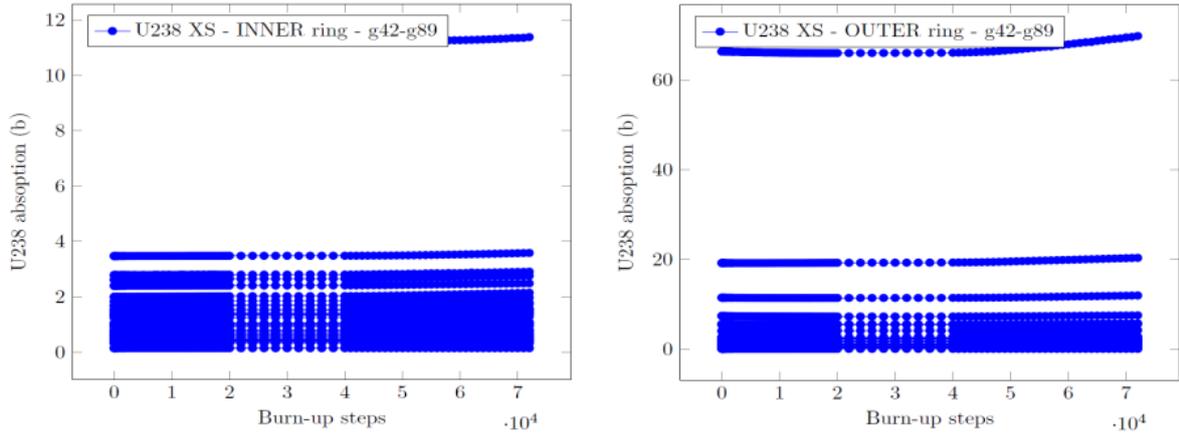


Figure 12.3: $\sigma_{U-238}^{g,in}$ and $\sigma_{U-238}^{g,out}$ vs burn-up, for $g \in [42, 89]$.

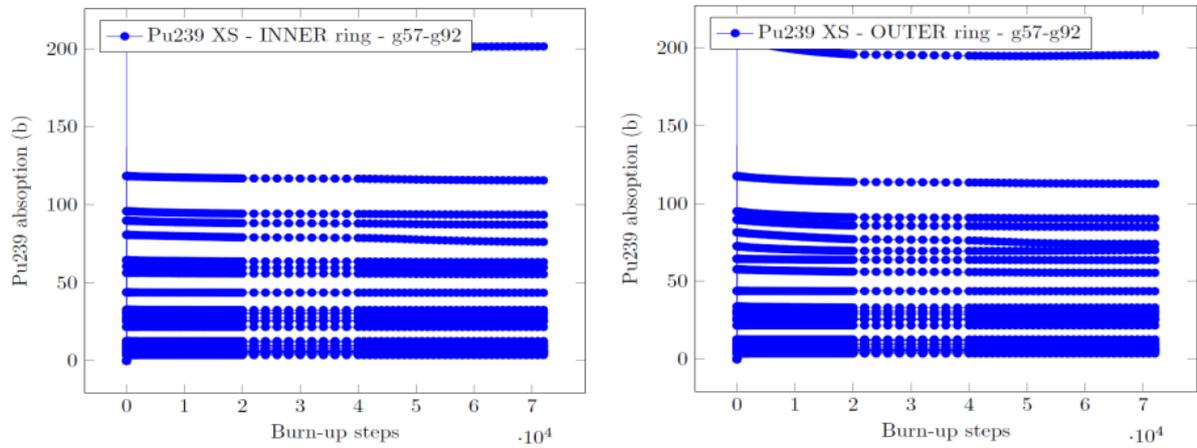


Figure 12.4: $\sigma_{Pu-239}^{g,in}$ and $\sigma_{Pu-239}^{g,out}$ vs burn-up, for $g \in [54, 92]$.

with respect to burn-up. One can also notice that, for ^{238}U , the absorption cross section in the inner ring of UOX is about 5 times smaller than in the outer ring of UOX/Gd, due to the skin effect.

The same conclusion may be drawn, by analysing the cross section variation, defined as

$$\Delta_j^{g,e}(BU) = \left(1 - \frac{\sigma_j^{g,e}(BU)}{\sigma_j^{g,e}(0)}\right) \cdot 100\%, \quad (12.15)$$

as a function of burn-up, for all self-shielded energy groups (Fig. 12.5). In the innermost ring of UOX, the maximum variation amounts to $0.15 \frac{\text{GWd}}{\text{ton}}$, for ^{238}U . Variations of the same type are observed in the outermost ring of UOX/Gd, with exception of one group ($g = 79$), whose behaviour is less regular. This trend may be caused by numerical errors inherent to the self-shielding calculation, within the resonance at $g = 79$.

These preliminary results suggest the possibility to reduce the number of burn-up steps, and thus the memory occupation. Microscopic cross sections may be reconstructed by low-order piece-wise polynomial functions, by interpolation on a coarser burn-up mesh.

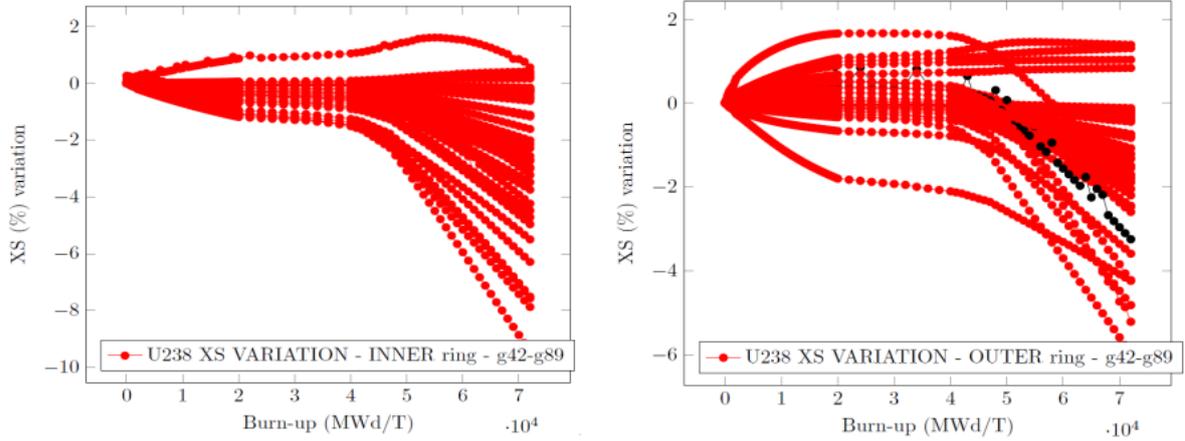


Figure 12.5: $\Delta_{U-238}^{g,in}(BU)$ and $\Delta_{U-238}^{g,out}(BU)$ vs burn-up, for $g \in [54, 92]$. The black curve on the right-hand side corresponds to $g = 79$.

This said, it is interesting to compare the presented strategy, based on direct cross-section interpolation (Sect. 12.3.4), with the one employed in standard two-step simulations (Sect. 12.3.3), based on spatial homogenization and energy condensation. Let us consider the case of ^{157}Gd . Figure 12.6 displays the variation of the effective microscopic cross section (Eq. 12.15), as a function of burn-up, if neither homogenization nor energy condensation are applied. Conversely,

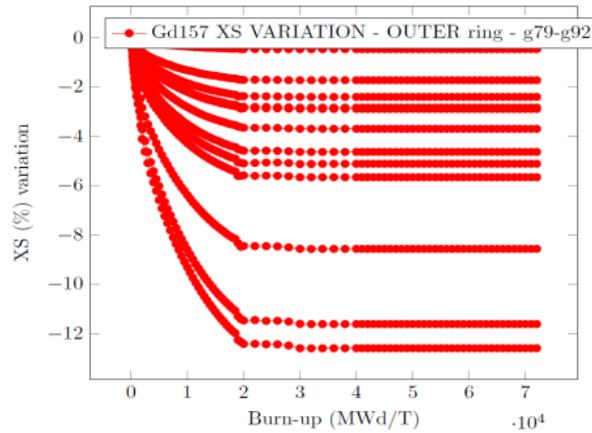


Figure 12.6: $\Delta_{Gd-157}^{g,out}(BU)$ vs burn-up, for $g \in [79, 92]$.

Fig. 12.7 provides the effective microscopic cross sections, used in standard core calculation, which underwent homogenization and energy condensation to $H = 2$ groups. The latter vary up to $\Delta_{Gd-157}^{0,out}(BU = 7 \cdot 10^4) \approx -20\%$ and $\Delta_{Gd-157}^{1,out}(BU = 7 \cdot 10^4) \approx -2900\%$, for $g = 0$ (fast group) and $g = 1$ (thermal group), respectively, hence they are much more sensitive to burn-up variations.

As the effective microscopic cross sections are, based on the prior observations, relatively regular functions of BU , one may propose to approximate their behaviour by a polynomial series development, $p_j^{g,e}(BU)$, and approximate $\sigma_j^{g,e}(BU^*)$ by $p_j^{g,e}(BU^*)$, where BU^* is an arbitrary value of burn-up. The advantage is clear: the coefficients of $p_j^{g,e}$ would be computed once for all, at the beginning of the simulation, after cross-section interpolation, and stored in memory. If the order of $p_j^{g,e}$ (i.e., the number of stored moments) is lower than N_{BU} , then the

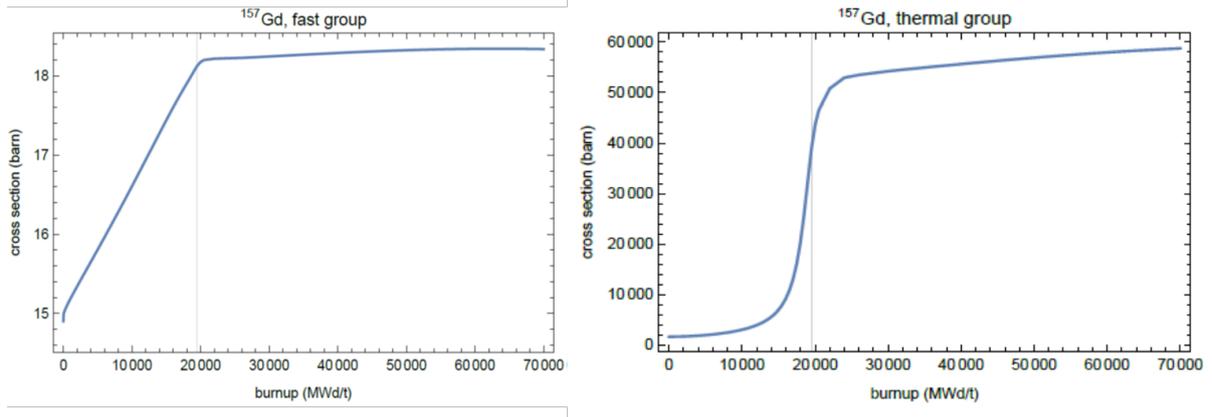


Figure 12.7: $\sigma_{Gd-157}^{0,out}(BU)$ and $\sigma_{Gd-157}^{1,out}(BU)$ vs burn-up, where 0 = fast group and 1 = thermal group. Figures from: I. Zmijarevic, 'Some aspects of creating and using multi-parameter cross section library', DES/ISAS/DM2S/SERMA/LLPR/NT/2022-70741/A, Courtesy of Igor Zmijarevic.

memory occupation is reduced.

Let us define

$$x = \frac{2 \cdot BU}{BU_{max}} - 1, \quad (12.16)$$

where $x \in [-1, 1]$ and BU_{max} denotes the value of burn-up at step $n = N_{BU}$, and project the interpolated $\sigma_j^{g,e}$ onto Legendre polynomials,

$$\sigma_{l,j}^{g,e} = \frac{1}{2} \int_{-1}^1 \sigma_j^{g,e}(x) P_l(x) dx. \quad (12.17)$$

Fig. 12.8 depicts the moments of the absorption cross section of ^{238}U , up to $l = 6$, within the innermost ring of UOX, as a function of energy, for $g \in [42, 89]$. If the parametric cross section is

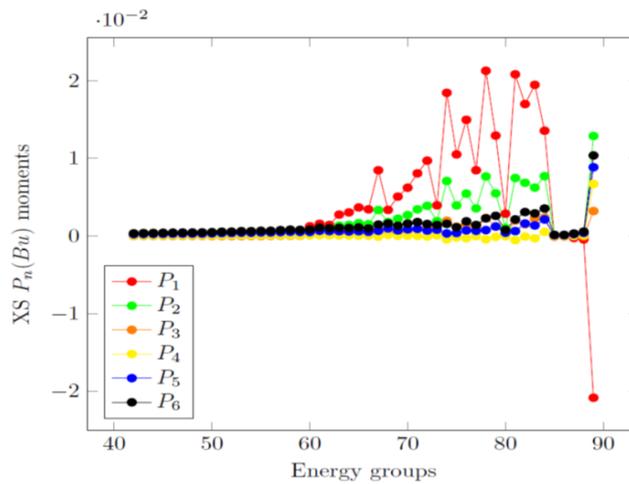


Figure 12.8: Legendre moments of $\sigma_{U-238,abs}^{g,in}$, up to $l = 6$.

approximated by Legendre series development, up to order $l = 6$, the maximum relative error with respect to the true value of $\sigma_j^{g,e}$ is less than 3%, in absolute value, for $g = 89$ (Fig. 12.9). In order to reduce the error, one may resort to one of the following strategies:

- Increase the order of $p_j^{g,e}(BU)$ and thus the number of moments stored in memory;
- Discretize the interval $[0, BU_{max}]$ into M sub-intervals and approximate $\sigma_j^{g,e}(BU)$ by piecewise polynomial expansions.

This method appears to be promising and may be further investigated in the future.

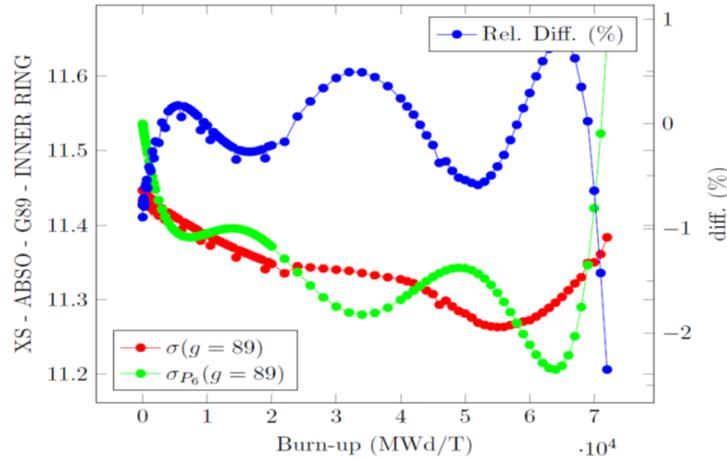


Figure 12.9: $\sigma_{U-238,abs}^{89,in}$ vs its approximation by Legendre series expansion of order 6 (in red and green, respectively), for group 89, as a function of burn-up. In blue, the relative error (%).

An alternative strategy to Eq. (12.17) may be the following. Based on Figs. 12.3-12.6, the parametric cross sections are a weak function of BU . One may propose to coarsen the burn-up mesh, as shown in Fig. 12.10, where only the steps marked with a black dot are retained. Hence, given three successive burn-up steps, namely $n, n + 1$ and $n + 2$, $\sigma_j^{g,e}(BU)$ is approximated by a linear spline, passing through the points $(BU_n, \sigma_j^{g,e}(BU_n))$ and $(BU_{n+2}, \sigma_j^{g,e}(BU_{n+2}))$. This model is referred as linear-tent approximation. A numerical comparison is provided in Fig. 12.11, where the effective microscopic absorption cross section of ^{238}U is compared to its linear-tent approximation, both in the inner and in the outer ring, for $g = 89$. The maximum relative error does not exceed 0.2%, in absolute value. Note that only one burn-up step out of two is plotted, as, for the other, the error is equal to zero, by definition. These results suggest that the BU grid of $\sigma_j^{g,e}$ can be greatly coarsened, with respect to the homogenized few-group cross sections, used in standard two-step reactor core calculations.

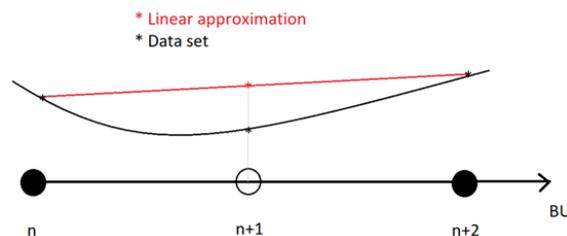


Figure 12.10: Linear-tent approximation. The parametric microscopic cross section is approximated by a linear spline, connecting the points $(BU_n, \sigma_j^{g,e}(BU_n))$ and $(BU_{n+2}, \sigma_j^{g,e}(BU_{n+2}))$, allowing for a major reduction of the number of burn-up steps.

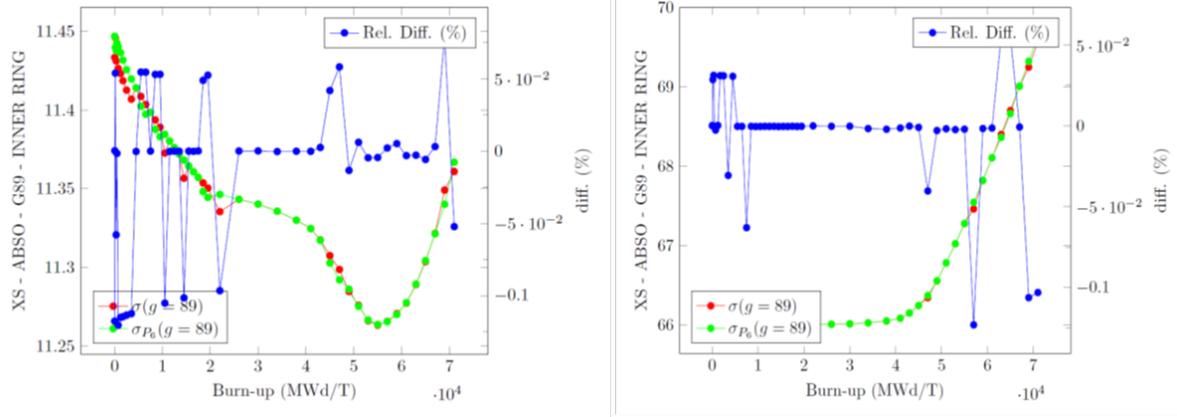


Figure 12.11: Linear-tent approximation. $\sigma_{U-238,abs}^{89,in/out}$ (in red) vs its linear-tent approximation (in green), as a function of BU . In blue, the relative error (%).

12.4.3 . Dependence on burn-up and fuel temperature

Analogously to Sect. 12.4.2, let us define the average microscopic cross section of isotope j and group g , as

$$\bar{\sigma}_j^g = \frac{1}{N_{T_f} N_e N_{BU}} \sum_b \sum_n \sum_e \sigma_j^{g,e}(BU_n, T_{f,b}), \quad (12.18)$$

and the associated RMS,

$$RMS_j^g = \frac{1}{N_{T_f} N_e N_{BU}} \sqrt{\sum_b \sum_n \sum_e (\sigma_j^{g,e}(BU_n, T_{f,b}) - \bar{\sigma}_j^g)^2}, \quad (12.19)$$

where N_{T_f} is the number of fuel-temperature points, N_{BU} is the number of burn-up steps and N_e is the number of rings containing the isotope. Once again, all the other parameters are fixed and are not explicitly indicated, for the sake of conciseness. Figures 12.12a-12.12b compare the first and second-order moments (Eqs. (12.18)-(12.19), respectively), of ^{238}U and ^{239}Pu , for all self-shielded energy groups. For ^{239}Pu , the ratio $RMS/\sigma \ll 1, \forall g \in [g_2, g_1]$ (Fig. 12.12b), hence the effective cross section shows little variability with respect to its mean value. This observation applies to ^{238}U as well, for a large number of self-shielded energy groups, with one exception, i.e., $g = 89$, where $\bar{\sigma}_j^g \sim 5 \cdot RMS$.

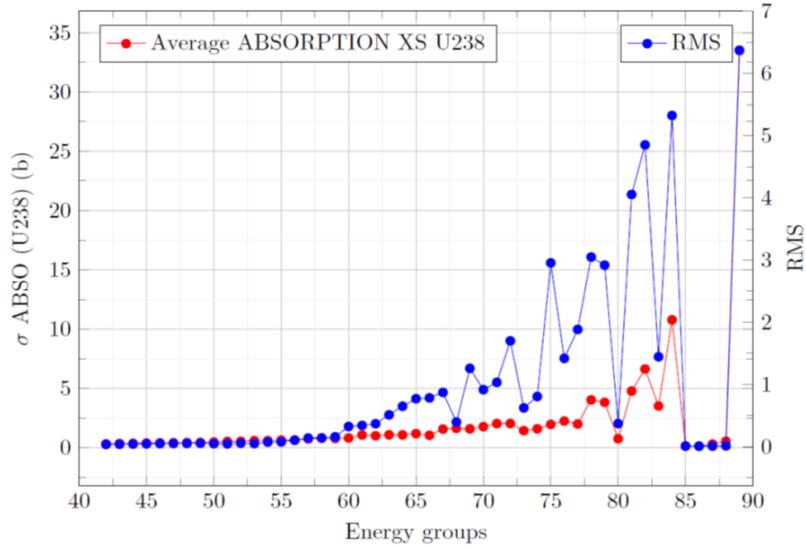
Figure 12.13 displays the effective microscopic absorption cross section of ^{238}U , in the innermost ring of UOX (on the left) and outermost ring UOX/Gd (on the right), as a function of BU , for 4 values of T_f . For the sake of brevity, only one group is shown, i.e., $g = 84$. The maximum variation of $\sigma_{U-238,abs}^{84,in/out}$ with respect to the fuel temperature amounts to $\sim 30 \text{ pcm} \cdot K^{-1}$.

Let us recall the definition of SSH factor, for one energy group, namely

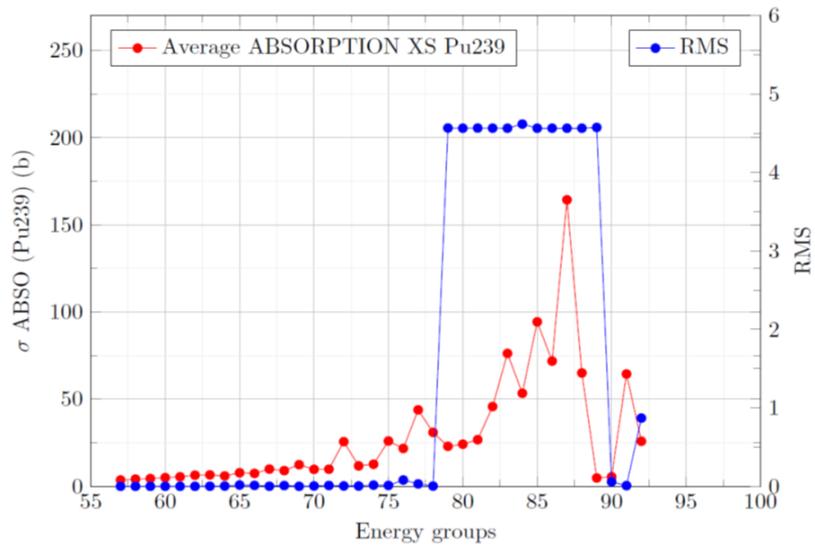
$$\mu_j^e(BU, T) = \frac{\sigma_j^e(BU, T)}{\sigma_j^\infty(T)}, \quad (12.20)$$

where all parameters are fixed, except for the burn-up and the local temperature. One may observe that most of the dependence on T is contained in $\sigma_j^\infty(T)$, due to Doppler effect. Hence, let us propose the following approximation:

$$\mu_j^e(BU, T) \approx \lambda_j^e(BU) \cdot \eta_j^e(T). \quad (12.21)$$



(a) Average microscopic absorption cross section of ^{238}U vs RMS, for groups between $g_2 = 42$ and $g_1 = 89$.



(b) Average microscopic absorption cross section of ^{239}Pu vs RMS, for groups between $g_2 = 57$ and $g_1 = 92$

Figure 12.12: Average microscopic absorption cross section vs RMS for two fuel isotopes.

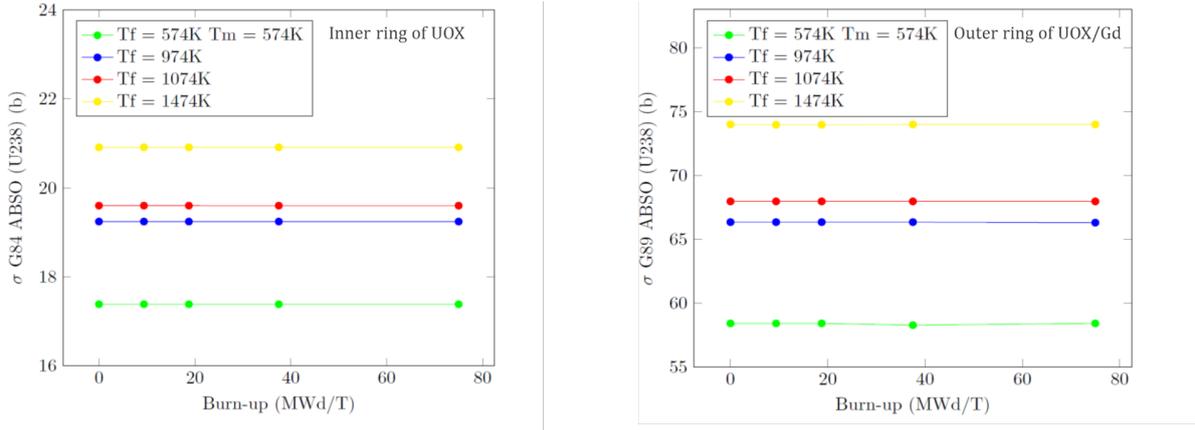


Figure 12.13: Dependence on burn-up and fuel temperature: $\sigma_{U-238,abs}^{84,in/out}$ vs BU and T_f .

Given $T = T^*$, one has

$$\frac{\sigma_j^e(BU, T^*)}{\sigma_j^e(BU = 0, T^*)} \approx \frac{\lambda_j^e(BU) \cdot \eta_j^e(T^*) \cdot \sigma_j^\infty(T^*)}{\lambda_j^e(BU = 0) \cdot \eta_j^e(T^*) \cdot \sigma_j^\infty(T^*)} = \frac{\lambda_j^e(BU)}{\lambda_j^e(BU = 0)}. \quad (12.22)$$

By setting $\lambda_j^e(BU = 0) = 1$, Eq. (12.22) becomes

$$\frac{\sigma_j^e(BU, T^*)}{\sigma_j^e(BU = 0, T^*)} \approx \lambda_j^e(BU). \quad (12.23)$$

This approximation, henceforth referred as $BU - T$ factorization, is supported by Figs. 12.14-12.15, displaying

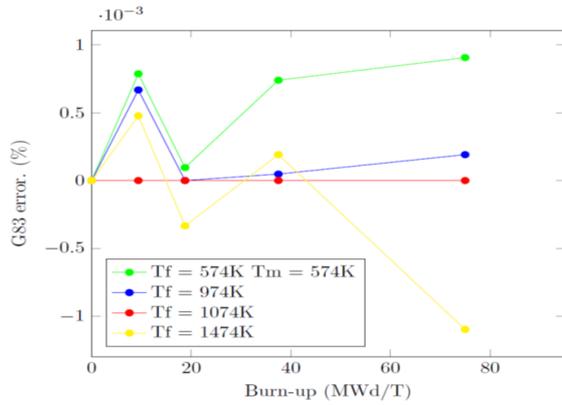
$$\epsilon_{\%}(BU, T) = \left(1 - \frac{\lambda_j^e(BU) \cdot \eta_j^e(T)}{\mu_j^e(BU, T)} \right) \cdot 100\%, \quad (12.24)$$

as a function of BU , for four values of temperature. The considered isotope is ^{238}U , within the outer ring of UOX/Gd, for four energy groups (indicated on the vertical axis), within the resonant interval. Figures 12.14 show that the relative error does not exceed few *pcm*, for $g = 83, 84$. Larger errors may be observed for $g = 79$ and $g = 89$, for some values of the burn-up (Fig. 12.15).

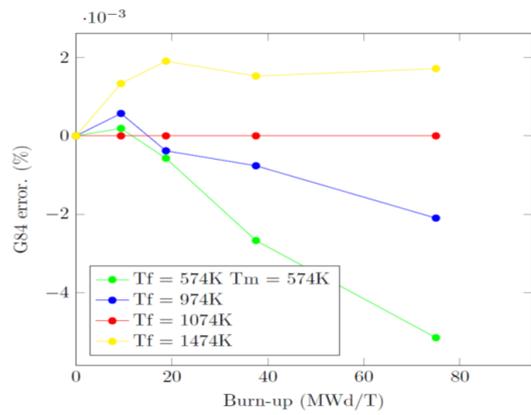
12.5 . Preliminary V&V and memory occupation of the parametric library of Shelf-Shielding Factors (EPL-SSF)

Table 12.1 reports a comparison between APOLLO3® and XSTOOL+IDT, for a 2D lattice calculation. The considered system is a 17×17 unrodded assembly of a PWR core, with 3.2%-enriched UOX fuel and 32 fuel-rods containing gadolinium. The assembly is depleted up to 72 *GWd/ton*, by 95 burn-up points, under nominal conditions. The remaining phase space is explored by 64 state-points, consisting of 4 fuel temperatures, 4 moderator temperatures and 4 boron concentrations, at beginning of cycle (i.e., $BU = 0$). The state point parameters are fixed as follows

- Fuel temperature, $T_f = \{474.4, 554, 754, 1074\} K$;
- Moderator temperature, $T_w = \{424, 524, 574, 614\} K$;

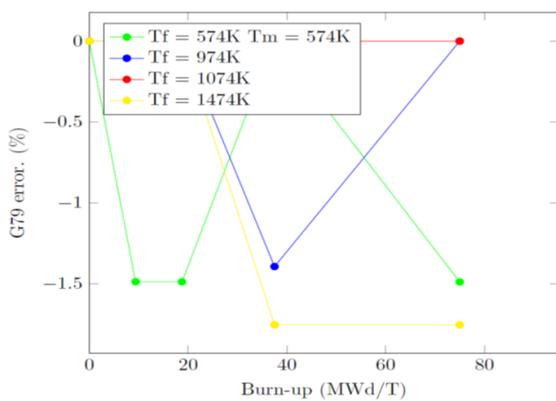


(a) $BU - T$ factorization, $g = 83$.

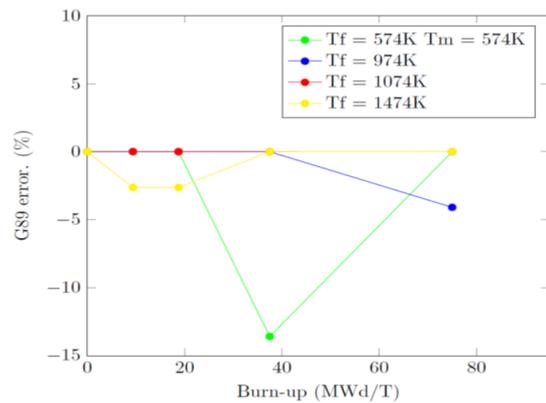


(b) $BU - T$ factorization, $g = 84$.

Figure 12.14: $BU - T$ factorization. Relative difference, i.e., $\epsilon_{\%}$ (Eq. (12.24)), observed for $\sigma_{U-238,abs}^{g,out}$ ($g = 83$ on the left-hand side, $g = 84$ on the right-hand side), as a function of BU , for four different values of temperature.



(a) $BU - T$ factorization, $g = 79$.



(b) $BU - T$ factorization, $g = 89$.

Figure 12.15: $BU - T$ factorization. Relative difference, i.e., $\epsilon_{\%}$ (Eq. (12.24)), observed for $\sigma_{U-238,abs}^{g,out}$ ($g = 79$ on the left-hand side, $g = 89$ on the right-hand side), as a function of BU , for four different values of temperature.

- Moderator density, $\rho_w = \{0.994, 0.804305, 0.734244, 0.68641\} g/cm^3$;
- Boron concentration, $C_B = \{0, 600, 1000, 1600\} ppm$.

For each point, APOLLO3®

- performs a self-shielding calculation by Multi-Cell current-coupled collision probabilities and Fine Structure model, with 281-group P_0 transport-corrected cross sections;
- runs a TDT-MOC lattice calculation, with 281-group P_1 cross sections;
- appends to the EPL-SSF output file the resonant isotope self-shielded cross sections and the neutron spectra, together with the concentrations per medium;
- depletes the fuel and updates the isotopic concentrations.

The dataset contained in the EPL-SSF file is constructed by taking advantage of the $BU - T$ factorization. More precisely, the EPL-SSF file contains 95 state-points, under nominal conditions, plus 6×64 points for the parameters T_f , T_c , and C_B , with 6 selected burn-up levels, namely $BU = \{0, 75, 1750, 10000, 20000, 40000\} MWd/ton$, subsequently denoted by a subscript I , assuming values from 0 to 5. These points are computed by the classical 'reprise' calculation.

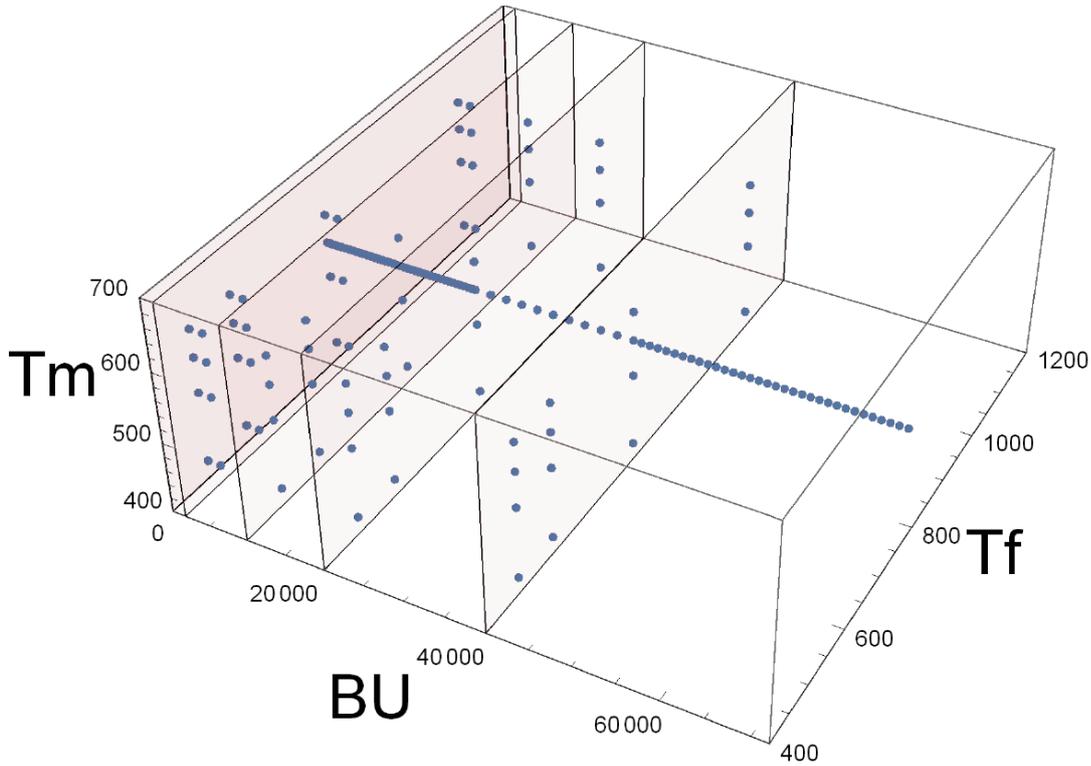


Figure 12.16: State-points grid in the parametric phase space ($BU, T_f, T_m, C_B = 0$).

Figure 12.16 sketches the stored state-points, at $C_B = 0$. Six macro-intervals define a coarse burn-up grid, namely

$$\Delta BU_I = \begin{cases} [0, \frac{BU_1}{2}[, & \text{for } I = 0, \\ [\frac{BU_I + BU_{I-1}}{2}, \frac{BU_I + BU_{I+1}}{2}[, & \text{for } I = 1, \dots, 4, \\ [\frac{BU_I + BU_{I-1}}{2}, \infty[, & \text{for } I = 5, \end{cases}$$

$BU \left[\frac{MWd}{ton} \right]$	A3®	XSTOOL+IDT	reac. rel. diff. (pcm)	max abs. pin-power diff. (%)
100	0.85430	0.85365	76	0.9
500	0.83595	0.83529	78	1.2
3500	0.86070	0.86127	-66	0.6
8500	0.88948	0.89021	-82	0.5
13500	0.91918	0.91980	-67	0.3
18500	0.95639	0.95750	-116	0.8
47000	0.98895	0.98830	65	0.4
57000	0.85999	0.86015	-18	0.8
72000	0.75946	0.76018	-94	0.7

Table 12.1: APOLLO3® vs XSTOOL+IDT, for a 2D 3.2% UOX-Gd32 assembly simulation.

in which the effective microscopic cross section is reconstructed by a piece-wise $BU - T$ factorization. Let us simplify the notation, by setting $T = (T_f, T_m, C_B)$. The effective self-shielded cross section of isotope j at a point (BU, T) is computed as

$$\sigma_j^e(BU, T) = \lambda_j^e(BU) \sigma_j^e(BU_I, T), \quad \text{for } BU \in \Delta BU_I, \quad (12.25)$$

where $\lambda_j^e(BU)$ is the burn-up shape function λ_j^e computed under nominal conditions, namely

$$T^* = (T_f = 754^\circ K, \quad T_m = 574^\circ K, \quad C_B = 0 \text{ ppm}),$$

as

$$\lambda_j^e(BU) = \frac{\sigma_j^e(BU, T^*)}{\sigma_j^e(0, T^*)}. \quad (12.26)$$

In Eq. (12.26), the effective microscopic cross section $\sigma_j^e(BU, T^*)$ is reconstructed at point BU , by linear interpolation, over the 95 burn-up points stored in the library. On the other hand, the cross section $\sigma_j^e(BU_I, T)$, defined in Eq. 12.25, is computed by multilinear tent-functions over the T -hyperplane (perpendicular to BU), by interpolating the points stored at $BU = BU_I$.

XSTOOL interpolation has been verified by performing a transport simulation by IDT in 9 burn-up check points, shown in the first column of Tab. 12.1, with

$$T = (T_f = 820 \text{ K}, \quad T_m = 614 \text{ K}, \quad C_B = 0 \text{ ppm}).$$

The results provided by IDT have been compared to those of APOLLO3, which are performed by self-shielding the resonant isotopes and running a MOC calculation at the same state point. As shown, the reactivity difference between APOLLO3® and XSTOOL+IDT remains bounded within approximately $\pm 100 \text{ pcm}$. The pin-power error is at most 1.2%.

Table 12.2 reports the memory occupation of the EPL-SSF file, for several PWR fuel assembly types. Each dataset contains the results of the lattice calculations performed by the APOLLO3®-SIGAR tool chain (REF), corresponding to a series of 3.2%-enriched UOX assemblies. The phase space is explored by the 64 state-points sketched previously.

3.2%-enriched UOX assembly	Memory (<i>MB</i>)	Memory/state-point (<i>MB</i>)
rod-out	19	0.47
AIC	30	0.33
B4C	22	0.29
Gd16-AIC	68	1.62
Gd16-B4C	60	0.93
Gd32-rod-out	56	0.85
Gd32-AIC	68	1.06
Gd32-B4C	60	0.93

Table 12.2: Memory occupation of the parametric HDF file (EPL-SSF) for several lattice types.

N. of points	Memory (<i>MB</i>)
95	41
$95 + 6 \times 64 = 479$	377

Table 12.3: Memory occupation of the EPL-SSF file for a depleted UOX-Gd32 assembly.

Table 12.3 contains the memory occupation for a depleted UOX-Gd32 assembly, with burn-up up to 72 GWd/ton , discretized by 95 points. The table also shows the total amount of memory of the library used to perform the 'XSTOOL+IDT' calculations of Tab. 12.1.

Considering 3 control-rod configurations per assembly, i.e., rod-out, B_4C and AIC , we can estimate to need approximately 300 to 500 *MB* to cover the parametric phase space of each assembly and few *GB* of data for a full core simulation.

12.6 . Xenon initial number density

XSTOOL can estimate the equilibrium xenon number density, $C_{Xe,m}$, for each fissile medium m , at zero burn-up, by means of a dedicated option. The initial xenon concentration is approximated by

$$C_{Xe,m} = \frac{Y_{Xe,m} I_{f,m}}{\tau_{a,Xe,m} + 10^{-24} \cdot \lambda_{Xe}}$$

with

- $Y_{Xe,m}$ as the fission yield,
- $I_{f,m} = \sum_{i \in m} \sum_{g=1}^{N_g} \Sigma_{f,m,i,g} \phi_{m,g}$ as the fission integral, evaluated by means of the macroscopic cross sections $\Sigma_{f,m,i,g}$ of all fissile isotopes $i \in m$, and of the spectrum $\phi_{m,g}$, computed by APOLLO3® and retrieved from the EPL-SSF file,

- $\tau_{a,Xe,m}$ as the xenon absorption rate,

$$\tau_{a,Xe,m} = \sum_{g=1}^{N_g} \sigma_{a,Xe,g} \phi_{m,g}, \quad (12.27)$$

- λ_{Xe} as the xenon decay constant.

The spectrum $\phi_{m,g}$ can be optionally normalized at a user-defined average power density, P_N (in W/g fuel), by means of the constant

$$\alpha_m = \frac{\rho_m P_N}{k_{fiss} I_{f,m}^*}, \quad (12.28)$$

where ρ_m is the heavy material density, k_{fiss} is the average energy released per fission (set to $0.324 \times 10^{-10} Ws/fission$), while $I_{f,m}^*$ is the fission integral without normalization.

12.7 . XSTOOL/IDT/THEDI COUPLING

Let us illustrate a novel algorithm allowing for coupled neutronics/thermal-hydraulics calculations, where the thermal feedback and the updated macroscopic cross section library are provided by THEDI and XSTOOL, respectively, while neutron transport is simulated by IDT. The proposed scheme is strongly related to Sect. 12.3.4, as interpolated microscopic cross sections are generated purely on the basis of the external reference library (GALILEE) and of the parametric self-shielding factors library (EPL-SSF), without preliminary homogenization and energy condensation. The obtained macroscopic cross sections may be condensed in energy, by means of a dedicated option of XSTOOL, and are provided to IDT in the form of ASCII files of the kind *subdo_kij.xml*, *subdo_kij* being a GCU (Sect. 6.4). Note that k, i, j define the z, x and y subdomain-lodging in the pivot grid of the domain decomposed (Sect. 6.4), while *subdo_kij.xml* contains the (condensed) macroscopic cross-section library and KERMA (Kinetic Energy Released in MAtter) necessary for its calculation.

The tight N-TH coupling implemented in IDT consists in introducing the boron concentration update and the thermal-hydraulic solution inside the CMFD-accelerated power iterations. Using the integer e as the fixed-point iteration index, the N-TH coupled equations are solved iteratively within a single external loop as follows,

$$\left\{ \begin{array}{l}
\text{Multigroup-transport (XSTOOL+IDT):} \\
\mathbf{B}(c^{(e)}, T^{(e)})\psi^{(e+1/2)} = \frac{1}{\lambda^{(e)}}\mathbf{F}(c^{(e)}, T^{(e)})\psi^{(e)}; \\
\\
\text{CMFD solver (IDT):} \\
\mathbf{A}(\psi^{(e+1/2)})\phi^{CMFD} = \frac{1}{\lambda^{(e+1)}}\mathbf{F}(\psi^{(e+1/2)})\phi^{CMFD}; \\
\\
\text{Flux update (IDT):} \\
\psi^{(e+1)} = \frac{\phi}{\int \psi^{(e+1/2)}}\psi^{(e+1/2)}; \tag{12.29} \\
\\
\text{Boron update (IDT):} \\
C_B^{(e+1)} = C_B^{(e)} + \left(\frac{C_B^{(e)} - C_B^{(e-1)}}{k^{(e+1)} - k^{(e)}} \right) (k_{ref} - k^{(e+1)}); \\
\\
\text{TH solver (THEDI):} \\
T^{(e+1)} = \mathbf{H}(\psi^{(e+1)}).
\end{array} \right.$$

Algorithm 5 provides details on the calculation scheme described in Eq. (12.29), with each point being discussed in a dedicated section (Sects. 12.7.1-12.7.10).

12.7.1 . IDT input file

IDT input consists of a file containing all geometrical and material composition information, as well as intuitive inputs for automatic domain decomposition and MOSC/ S_n solver options. The file consists of subsequent sections, which can be briefly summarized as follows:

1. **HCC XYZ Grid.** A mesh of Cartesian nodes is defined (Sect. 6.3.2), by means of simple instructions, allowing for X/Y/Z displacements and the construction of repetitive Cartesian patterns, with fixed side lengths;
2. **HCC Discretization.** Nodes of different types may be designed in this section. The user can input the TRM, by standard or customized options (Sect. 7.2), define the symmetry properties of the cell (Sect. 10.4), orient the cylinders (Sect. 7.5), indicate the number and spacing of the inner Cartesian planes (Sect. 7.2) and give instructions on the ray-tracing quadrature (Sect. 7.4);
3. **HCC-geometry-types/HCC-grid-lodging Association.** The Cartesian nodes defined in the previous point are lodged into the Cartesian mesh, by simple association between the HCC name and the label associated to a (repetitive) Cartesian pattern composing the macro-grid;
4. **Boundary Conditions.** The boundary conditions (BCs) are indicated for each side. The user can choose among a set of BC options, as mentioned in Sect. 6.1;
5. **Angular Quadrature Formula.** IDT offers a large set of angular quadrature formulas, including Gauss-Legendre and Chebyshev-Legendre product quadratures as well as Level Symmetric quadrature formulas;

Algorithm 5 : Proposed iterative scheme for coupled N/TH multiphysics.

- 1: **Read** IDT input file
 - 2: Generation of the domain decomposed (IDT-DDM input file), of XSTOOL input files (one for each GCU) and of the THEDI input file (TH channels' data)
 - 3: **Read** IDT-DDM input file, . Boron concentration and temperature field initialization,
$$C_B^{(0)}, \left. \frac{\Delta C_B}{\Delta k} \right|^{(0)} \mathcal{T}^{(0)} = \{T_f^{(0)}, T_w^{(0)}, \rho_w^{(0)}\}$$
 - 4: TH sub-channels definition and generation of IDT2THEDI input file
 - 5: **for** $u = 1 \rightarrow U$ {Loop over subdomains} **do**
 - 6: Generation of (condensed) macroscopic cross-section library, by XSTOOL
 - 7: CP Matrices evaluation
 - 8: **end for**
 - 9: Eigenpair and Fission Integral initialization, $k^{(0)}, \psi^{(0)}, I_f^{(0)}$
 - 10: **for** $e = 0 \rightarrow N_e$ {Outer Loop} **do**
 - 11: **for** $u = 1 \rightarrow U$ **do**
 - 12: Local Multigroup Transport Iterations: determination of $\{k^{(e+\frac{1}{2})}, \psi^{(e+\frac{1}{2})}\}$
 - 13: **end for**
 - 14: CMFD Initialization, $\{k_{CMFD}^0 = k^{(e+\frac{1}{2})}, \Phi_{CMFD}^{(0)} = \int \psi^{(e+\frac{1}{2})}(\xi) d\xi\}$
 - 15: Solve Global CMFD Diffusion with few-group cross-sections and homogenized pin-cells
 - 16: Flux and Eigenvalue update, $\left\{ k^{e+1} = k_{CMFD}, \psi^{(e+1)} = \psi^{(e+1/2)} \frac{\Phi_{CMFD}}{\Phi_{CMFD}^{(0)}} \right\}$
 - 17: Fission Integral update, $I_f^{(e+1)}$
 - 18: Check error condition on transport solution and **return**, if verified
 - 19: **if** $C_B^{(e)} > 0$ {otherwise boron search is skipped} **then**
 - 20: Critical Boron concentration search, $C^{(e+1)} = f\left(C^{(e)}, k_{ref}, k^{(e-1)}, k^{(e)}, \left. \frac{\Delta C}{\Delta k} \right|^{(e)}\right)$
 - 21: Set $lg_B = \mathbf{True}$, unless convergence within eigenvalue precision
 - 22: **end if**
 - 23: Generation of Power map input for TH solver
 - 24: Update TH field by THEDI, $\mathcal{T}^{(e+1)} = \mathcal{H}(P, p, T_{in}, \dot{m})$
 - 25: **for** $u = 1 \rightarrow U$ {Loop over subdomains} **do**
 - 26: **if** $\exists \mathbf{r} \in D_u$ such that $\delta T_f(\mathbf{r}), \delta \rho_w(\mathbf{r}) > \epsilon_f, \epsilon_w$, **then** $lg_u = \mathbf{True}$, **else** $lg_u = \mathbf{False}$
 - 27: Update XSTOOL input file
 - 28: **if** lg_u **or** lg_B **then**
 - 29: Update Macroscopic Cross Sections, by XSTOOL
 - 30: Recompute Neumann 'perturbation', $\delta \Sigma(\mathbf{r})$
 - 31: **if** $\delta \Sigma(\mathbf{r}) < 0$ **then**
 - 32: Recompute CP Matrices
 - 33: **end if**
 - 34: **end if**
 - 35: **end for**
 - 36: **end for**
-

6. **Solver Options.** In this section, details are provided on the solver options, e.g., MOSC order (Sects. 4.6.4-11.2), P_N -scattering order (Sect. 4.3), tolerances on the inner/thermal/outer iterations (Sect. 4.4), etc.
7. **Parametric Libraries Identification and Settings. Boron concentration.** The user has to define the path of the infinite-dilution cross section library, as well as the one of the external library containing the SSH-factors. The path to XSTOOL executable should also be provided, together with the parameter types involved in the calculation (T_f , T_w , ρ_w , %CR, C_B , BU, t) and respective ranges (e.g., for T_f , $[T_f^{min}, T_f^{max}]$). Moreover, one has to input the initial boron concentration, $C_B^{(0)}$, the interpolation option (polar, distance-weighted or multilinear) and the number of condensation groups ($G = 2, 4, 26, 30, 281$). If $C_B^{(0)} \geq 0$, IDT will search for the critical boron concentration, i.e., $C_B(k)$ such that $k = 1$ (Sect. 5.2.2);
8. **Thermal-Hydraulics Settings and Path.** The input also contains information on the discretization into thermal-hydraulic (TH) sub-channels, as well as on the path to the TH module, which runs THEDI. The TH mesh requires two parameters, namely S_x and S_y , representing the number of sub-channels along x and y, respectively. The algorithm allowing for TH discretization is illustrated in Sect. 12.7.4;
9. **Materials List.** The materials are indicated one by one, as $mat + e + b$, i.e., the combination of three labels, indicating the material type (e.g., steel, B₄C, AIC, UOX), the SSH region (e.g., ring a, b, c, d) and the temperature (e.g., T_f, T_w), respectively. In this section, the user should also initialize the state point of each material (e.g., the local temperature (T), the fuel temperature (T_f), the moderator temperature and density (T_w and ρ_w), the percentage of CR insertion (%CR), the burn-up (BU) and the time (t) together with the reference state-point in the parametric library to load the initial concentrations of the isotopes ;
10. **Radial geometries.** In order to design a reactor core, one should list all radial geometries, $\mathcal{G}_r, r = 1, 2, \dots, R$. Each \mathcal{G}_r is defined by a set of data, i.e., the number of circles (n), the pin-cell name and pitch, the presence of the spacer grid and its thickness and the full set of radii, $\rho_1, \rho_2, \dots, \rho_n$, composing \mathcal{G}_r ;
11. **Radial composition.** Each radial geometry may be associated to one or more 2D physical pin-cell types, namely $\mathcal{P}_l, l = 1, 2, \dots, L$. Each \mathcal{P}_l is defined by a set of information, including its name, its radial geometry, $\mathcal{G}_r(\mathcal{P}_l)$, and a list of materials, $\mu = 1, 2, \dots, \mu_{n+1}$, provided n is the number of circles composing $\mathcal{G}_r(\mathcal{P}_l)$;
12. **Axial geometries and compositions.** Each fuel/control rod and guide tube may be represented in real 3D geometry as follows. Suppose \mathcal{F} is a fuel rod, of length ℓ . In IDT, \mathcal{F} is represented by piling-up several axial layers, $\mathcal{A}_1, \dots, \mathcal{A}_K$, of different type and geometry, with length ℓ_1, \dots, ℓ_K , such that $\sum_{k=1}^K \ell_k = \ell$. Each axial layer \mathcal{A}_k is obtained by extrusion of one \mathcal{P}_l , with geometry $\mathcal{G}_r(\mathcal{P}_l)$, over an interval $[\sum_{k=1}^{k-1} \ell_k, \sum_{k=1}^k \ell_k]$. Note that all 3D elements should share the same length, i.e., ℓ , and that axial reflectors can be here included;
13. **Fuel assembly definition.** Each fuel assembly (F.A.) is a lattice of fuel rods, control rods and guide tubes, defined as in the previous point. In order to define their position in the F.A., one has to indicate the radial coordinates of each element, provided that the origin

of the frame of reference is located in the south/west corner. In this section, the user can also input a fuel bundle as well as a water gap;

14. **Core Definition.** The reactor core is given as an ensemble of F.A.s of defined radial positions;
15. **DDM Options.** DDM is automatically applied to the whole reactor, by defining (a) the number of axial subdivisions, M_x, M_y, M_z , as well as the number of fine steps (i.e., HCCs) for each axial subdivision (Sect. 6.4.3), (b) the number of processes, P , and (c) a complementary set of solver options (including the required precision, the maximum number of iterations, etc.).

IDT intersects the axial geometry defined by the fuel rods, the control rods and the guide tubes with the one provided by the HCC grid (1. **HCC XYZ Grid**). As a result, the HCC node in position (X, Y, Z) will contain a number of planes which depends on the number of axial layers \mathcal{A}_k within (X, Y, Z) (12. **Axial geometries and compositions**), plus the number of steps due to the inner Cartesian discretization (2. **HCC Discretization**).

12.7.2 . Automatic generation of the domain decomposed and of XSTOOL inputs

IDT is able to construct the domain decomposed, based on the instructions provided in the user input file (UIF), described in Sect. 12.7.1. Assuming the UIF, namely 'test.d', requests $M_x \times M_y \times M_z$ axial subdivisions for DDM, a new input file, of the kind 'test_ $M_x M_y M_z$.d', containing GCUs' description and solver options, is automatically generated by IDT. Each GCU is equipped with a macroscopic cross section library path, the HCC XYZ grid, HCCs' geometrical and material description, specular reflection boundary conditions, the angular quadrature formula, a set of solver options, including MOSC and P_N -scattering order. Once defined the whole set of GCUs, the pivot grid is specified, together with the number of maximum iterations, the required precision, the number of coarse energy groups for CMFD, etc. The user can also input a XYZ mesh different from the one indicated in Sect. 12.7.1 (1. **HCC XYZ grid**), in order to store the reaction rates on a customized XYZ grid, for inter-code comparisons.

Based on Sect. 12.7.1 (7. **Parametric Libraries Identification and Settings**, 9. **Materials List**), a file with extension '.xst' is automatically generated for each GCU of the domain decomposed. The content of this file is detailed in Sect. 12.3.5 and is provided as an input to XSTOOL, to produce a macroscopic cross-section library, for each GCU of the pivot grid.

12.7.3 . Boron concentration and temperature field initialization

The boron concentration is initialized to the value defined by the user (Sect. 12.7.1, 8. **Parametric Libraries Identification and Settings. Boron concentration**) in IDT input. Note that, if $C_B^{(0)} < 0$, no boron concentration search is performed. The TH field is initialized, according to user's inputs (Sect. 12.7.1, 9. **Materials List**). Note that $\mathcal{T}^{(0)}$ should be within the intervals defined in Sect. 12.7.1, 8. **Parametric Libraries Identification and Settings**.

12.7.4 . TH sub-channels definition

The discretization into TH sub-channels is performed by a simple algorithm, which resembles the one introduced in Sect. 6.4.3. Suppose N_k is the number of HCCs along axis k , where $k = x, y$, while S_k is the number of required sub-channels along the same axis. Let us define

$$\zeta_k = \left\lfloor \frac{N_k(\text{ mod } S_k)}{2} \right\rfloor. \quad (12.30)$$

If $\zeta_k = \frac{N_k(\text{ mod } S_k)}{2}$, then the HCCs are distributed symmetrically along axis k as follows:

- The first ζ_k sub-channels (i.e., the ones appearing on the left-hand side) contain $\lfloor N_k(\text{ mod } S_k) \rfloor + 1$ HCCs each;
- The sub-channels with order number between $\zeta_k + 1$ and $S_k - \zeta_k$ (i.e., the ones appearing in the middle) contain $\lfloor N_k(\text{ mod } S_k) \rfloor$ HCCs each;
- The last ζ_k sub-channels (appearing on the right-hand side) contain $\lfloor N_k(\text{ mod } S_k) \rfloor + 1$ HCCs each.

Conversely, if $\zeta_k \neq \frac{N_k(\text{ mod } S_k)}{2}$, the HCCs are, once again, distributed symmetrically along axis k , but with a different strategy:

- The first ζ_k sub-channels (appearing on the far left-hand side) contain $\lfloor N_k(\text{ mod } S_k) \rfloor + 1$ HCCs each;
- The sub-channels between $\zeta_k + 1, \frac{S_k}{2}$ (appearing on the left-hand side) contain $\lfloor N_k(\text{ mod } S_k) \rfloor$ HCCs each;
- The sub-channel with order number equal to $\frac{S_k}{2} + S_k(\text{ mod } 2)$ (i.e., the one appearing right in the middle) contains $\lfloor N_k(\text{ mod } S_k) \rfloor + 1$ HCCs;
- The sub-channels between $\frac{S_k}{2} + S_k(\text{ mod } 2) + 1, S_k - \zeta_k$ (appearing on the right-hand side) contain $\lfloor N_k(\text{ mod } S_k) \rfloor$ HCCs each;
- The last ζ_k sub-channels (appearing on the far right-hand side) contain $\lfloor N_k(\text{ mod } S_k) \rfloor + 1$ HCCs each.

Provided $S_x \cdot S_y$ is the total number of sub-channels, IDT computes a map of the kind illustrated in Algo. 6. This tool allows retrieving all HCC geometrical data, which can be accessed in memory by means of an integer number defined by

$$hcc_{nb} = (z_{pin} - 1) \cdot N_y \cdot N_x + (y_{pin} - 1) \cdot N_x + x_{pin}, \quad (12.31)$$

where $x_{pin}, y_{pin}, z_{pin}$ are provided by the prior map. The data listed in Tab. 12.4 are written on a file including all TH sub-channels, for IDT2THEDI usage. In particular:

- x_{pin}, y_{pin} and z_{pin} are three integers, defining the position of a cell, within the global HCC grid (previously indicated as (X, Y, Z));
- ch_{nb} denotes the channel order number, i.e., an integer identifying a TH sub-channel, to which the considered cell belongs;
- $w \in \mathbb{R}$ represents the contribution of the considered cell to the aforementioned sub-channel. This quantity is related to the inverse of the number of sub-channels to which the pin belongs. Thus, we store $\frac{1}{w} \in \mathbb{N}$, as integers are less expensive in terms of memory occupation, with respect to floating-point numbers;
- A_{pin} is the transverse area occupied by the pin, while A_{HCC} is the total transverse area occupied by the considered cell;
- p_{wet} and p_{heat} denote respectively the wetted perimeter and heating fraction;

x_{pin}	y_{pin}	z_{pin}	ch_{nb}	$\frac{1}{w}$	A_{pin}	A_{HCC}	p_{wet}	p_{heat}	lg_f	ρ_f	ρ_c^{in}	ρ_c^{out}	Δz	lg_{sg}
-----------	-----------	-----------	-----------	---------------	-----------	-----------	-----------	------------	--------	----------	---------------	----------------	------------	-----------

Table 12.4: Sub-channel geometry out/input file, generated by IDT, for N/TH coupling.

- lg_f is an integer number, assuming values 0/1, depending on the presence of nuclear fuel;
- $\rho_f, \rho_c^{in}, \rho_c^{out}$ represent the external radius of the fuel and the inner/outer radius of the clad, respectively;
- Δz is the HCC height;
- lg_{sg} is an integer number, assuming values 0/1, depending on the presence of a spacer grid.

Algorithm 6 : TH sub-channel map.

```

1: Allocate memory for TH sub-channel map,  $ch_{map}$ 
2: Initialize TH sub-channel map,  $ch_{map} = 0$ 
3: Initialize sub-channel order nb,  $ch_{nb} = 0$ 
4: for  $j = 1 \rightarrow M_y$  {Sub-channel coarse mesh along  $y$ } do
5:   Retrieve address 1st pin  $\in j$ ,  $adj_f$ 
6:   Retrieve address last pin  $\in j$ ,  $adj_l$ 
7:   for  $i = 1 \rightarrow M_x$  {Sub-channel coarse mesh along  $x$ } do
8:     Retrieve address 1st pin  $\in i$ ,  $adi_f$ 
9:     Retrieve address last pin  $\in i$ ,  $adi_l$ 
10:    Update sub-channel order nb,  $ch_{nb} \leftarrow ch_{nb} + 1$ 
11:    Initialize pin order nb per channel,  $pin_{nb} = 0$ 
12:    for  $y_{pin} = adj_f \rightarrow adj_l$  do
13:      for  $x_{pin} = adi_f \rightarrow adi_l$  do
14:        Update pin order nb per channel,  $pin_{nb} \leftarrow pin_{nb} + 1$ 
15:        for  $z_{pin} = 1 \rightarrow N_z$  do
16:          Store  $x_{pin}, y_{pin}, z_{pin}, \frac{1}{w}$  into TH sub-channel map, namely  $ch_{map}$ 
17:          { $w$  represents the weight of a pin in a given sub-channel and is given by
           the inverse of the number of sub-channels to which the pin belongs}
18:        end for
19:      end for
20:    end for
21:  end for
22: end for

```

12.7.5 . Macroscopic Cross-Section initialization by XSTOOL

Given

- the external library of infinite-dilution microscopic cross sections, $\sigma_j^\infty(T_k)$, (GALILEE), defined in Sect. 12.3.4,
- the library of SSH-factors, $\mu_j^e(\mathbf{T}_l, \mathbf{p}_n)$, (EPL-SSF), defined in Sect. 12.3.4,

- a set of files, of the kind 'subdo_kij.xst', one for each GCU, generated by IDT and described in Sect. 12.3.5,

XSTOOL performs

- linear interpolation of the infinite-dilution microscopic cross sections, with respect to \sqrt{T} ,
- multi-linear (for Cartesian grids), distance-based or polar (for non-Cartesian grids) interpolation of the effective microscopic cross sections, for resonant isotopes, with respect to T or \sqrt{T} ,

obtaining a set of interpolated nuclear data to the parameters indicated in Sect. 12.7.1, 9. **Materials List**, and $C_B = C_B^{(0)}$, indicated by the user (Sect. 12.7.1, 8. **Parametric Libraries Identification and Settings. Boron concentration**).

It is important to note that, contrarily to homogenized few-groups cross sections, the moderator (water) density is

- an interpolation parameter for microscopic cross sections of resonant isotopes,
- a physical parameter that allows for the renormalization of isotopic concentrations (e.g., 1H , 2H , ^{16}O , ^{10}B , ^{11}B) without any approximation.

Similarly the boron concentration is

- an interpolation parameter for resonant isotopes effective microscopic cross sections,
- a physical parameter that allows to precisely set the exact boron concentration to the prescribed *ppm*, for moderator cross sections.

This particular aspect of cross section setting is a peculiarity of the proposed scheme which allows the exact treatment of moderator cross sections. The macroscopic cross section library is derived, with possible energy condensation to $H = 2, 4, 26, 30$ groups, weighting by the APOLLO3 flux $\Phi^g(\mathbf{r})$ from lattice-calculation. The later is stored in the library per each medium and per state points. Also the spectrum has to be interpolated to the current set of parameters. A set of outputs of the kind 'subdo_kij.xsl', containing the interpolated (and condensed) multigroup macroscopic cross sections, is generated, for each GCU, for IDT usage (Sect. 12.7).

12.7.6 . CP Matrices evaluation

The transport operator reads as $\mathbf{L} = \mathbf{L}_0 - \delta\mathbf{L}$, $\mathbf{L}_0 = \mathbf{\Omega} \cdot \nabla + \Sigma_0$ and $\delta\mathbf{L} = \delta\Sigma$ being the reference (or unperturbed) operator and a '*perturbation*', respectively (see Sect. 9.2). \mathbf{L}_0 is inverted, at line 7 (Algo. 5), by MOSC, in the form of angular and energy-dependent CP matrices, namely $\underline{\mathbf{C}}, \underline{\mathbf{I}}, \underline{\mathbf{E}}, \underline{\mathbf{T}}$ (Sect. 4.6.4), computed using the macroscopic cross section library defined in 12.7.5. If $\|\mathbf{L}_0^{-1}\delta\mathbf{L}\| < 1$, the Neumann iterations ($n = 0, 1, 2, \dots$), performed within the inner loop by successive applications of $\mathbf{L}_0^{-1}\delta\mathbf{L}$ onto $\mathbf{L}_0^{-1}q$, converge to the transport solution, namely

$$\psi = \sum_{n \geq 0} (\mathbf{L}_0^{-1}\delta\mathbf{L})^n \mathbf{L}_0^{-1}q, \quad (12.32)$$

as detailed in Sect. 9.2.

12.7.7 . Eigenpair and Fission Integral initialization

Power iterations are initialized at line 9 (Algo. 5), by setting

$$k = k^{(0)}, \quad (12.33)$$

$$\psi = \psi^{(0)}, \quad (12.34)$$

$$I_f^{(0)}(\mathbf{r}) = \int dE \int d\Omega (\nu_f \Sigma_f)(\mathbf{r}, E) \psi^{(0)}(\mathbf{r}, \Omega, E), \quad (12.35)$$

$$\frac{\Delta C_B}{\Delta k} = \left. \frac{\Delta C_B}{\Delta k} \right|^{(0)}, \quad (12.36)$$

where $\left. \frac{\Delta C_B}{\Delta k} \right|^{(0)} = -10000$ is the initialization value of the derivative of $C_B = C_B(k)$.

12.7.8 . Outer Iterations

The thermal feedback is included within the outer loop. The idea is to generate a fine multigroup-transport spectrum, feeding a coarse operator, based on CMFD diffusion. The solution of the latter allows for the construction of the new power map, which is injected into THEDI, in order to update the TH field of each subdomain. Note that, in case of 'small' temperature variations with respect to the prior iteration, the subdomain cross-section library is not updated. In the following, outer iterations are illustrated, with particular reference to the calculation of the critical boron concentration and of the thermal-feedback (by THEDI), which may induce a subsequent change in the cross-section library (recomputed by XSTOOL) and possibly of the CP matrices. As customary, e is the outer iteration index.

12.7.9 . Multigroup Transport Iterations and CMFD

Let us consider the fixed-source boundary value problem of subdomain u ,

$$\left\{ \begin{array}{l} \mathbf{B}_u(C_B^{(e)}, \mathcal{T}^{(e)}) \psi_u^{(e+1/2)}(\xi) = \frac{1}{k^{(e)}} \mathbf{F}_u(C_B^{(e)}, \mathcal{T}^{(e)}) \psi_u^{(e)}(\xi), \quad \xi \in X_u, \\ \psi_u^{-(e+1/2)}(\xi) = \psi_v^{+(e)}(\xi), \quad \xi \in \partial X_u^- \cap \partial X_v^+, \quad \forall u \cap v \neq \emptyset, \\ \psi_u^{-(e+1/2)}(\xi) = 0, \quad \xi \in \partial X_u^- \cap \partial X^-. \end{array} \right. \quad (12.37)$$

with

$$\xi = (\mathbf{r}, \Omega, E) \in X, \partial X^\pm \quad (12.38)$$

$$X_u \equiv \{\mathbf{r} \in D_u, \Omega \in S^2, E \in \mathbb{R}^+\}, \quad (12.39)$$

$$\partial X_u^\pm \equiv \{\mathbf{r} \in \Gamma_u^\pm(\Omega), \Omega \in S^2, E \in \mathbb{R}^+\}, \quad (12.40)$$

$$u = 1, \dots, U, \quad (12.41)$$

where $\mathbf{B}_u = (\mathbf{L} - \mathbf{H})_u$ is the local net disappearance operator of subdomain u , with spatial domain D_u and incoming/outgoing border Γ_u^\pm . IDT inverts the operator \mathbf{B}_u by local multigroup transport iterations (Sect. 4.4). Currents are transmitted between neighbouring subdomains, in order to satisfy the continuity equation (Eq. 6.10). Once solved Eq. (12.37), the k -eigenvalue

and the local fission source are updated as follows

$$k^{(e+\frac{1}{2})} = k^{(e)} \frac{\sum_{u=1, \dots, U} (w, \mathbf{F}_u \psi_u^{(e+\frac{1}{2})})_u}{\sum_{u=1, \dots, U} (w, \mathbf{F}_u \psi_u^{(e)})_u}, \quad (12.42)$$

$$q_u^{(e+\frac{1}{2})}(\xi) = \frac{\mathbf{F}_u \psi_u^{(e+\frac{1}{2})}(\xi)}{k^{(e+\frac{1}{2})}}.$$

To accelerate neutron transport, multigroup CMFD is solved by successive global iterations, until full convergence. Let us consider the synthetic problem defined by

$$\mathbf{A}(\psi^{(e+\frac{1}{2})})\Phi(\mathbf{r}, E) = \frac{1}{k} \mathbf{F}_{CMFD}(\psi^{(e+\frac{1}{2})})\Phi(\mathbf{r}, E), \quad \mathbf{r} \in D, E \in \mathbb{R}^+, \quad (12.43)$$

with \mathbf{A} being the diffusion operator. Nuclear data are homogenized over coarse spatial meshes and condensed energy groups (Sect. 4.7.2). Eq. 12.43 is solved by power/thermal/inner iterations, the latter being approached by the BICGStab-Krylov method. The last transport iterate, namely $\{k^{(e+\frac{1}{2})}, \psi_u^{(e+\frac{1}{2})}(\xi)\}$, allows to initialize the CMFD solver, by setting

$$k^{CMFD,(0)} = k^{(e+\frac{1}{2})}, \quad \Phi_u^{CMFD,(0)} = \int d\xi \psi_u^{(e+\frac{1}{2})}, \quad (12.44)$$

where the transport solution is integrated over the CMFD mesh and over the whole solid angle. Once solved Eq. 12.43, the k -eigenvalue is updated, namely

$$k^{(e)} = k^{CMFD}, \quad (12.45)$$

and the transport solution is renormalized, i.e.,

$$\psi_u^{(e+1)} = \psi_u^{(e+\frac{1}{2})} \frac{\Phi_u^{CMFD}}{\Phi_u^{CMFD,(0)}}, \quad (12.46)$$

$$\psi_u^{\pm,(e+1)} = \psi_u^{\pm,(e+\frac{1}{2})} \frac{\Phi_u^{CMFD}}{\Phi_u^{CMFD,(0)}}, \quad (12.47)$$

where k^{CMFD} and Φ_u^{CMFD} represent the converged CMFD solution.

12.7.10 . Error Check

As mentioned in Sect. 6.4, outer iterations come to an end if the following set of conditions is satisfied:

$$\varepsilon_k^{rel} = \left| 1 - \frac{k^{(e+1)}}{k^{(e)}} \right| < \epsilon_k, \quad (12.48)$$

$$\varepsilon_{C_B}^{rel} = \left| 1 - \frac{C_B^{(e)}}{C_B^{(e-1)}} \right| < 0.1 \text{ ppm}, \quad (12.49)$$

$$\varepsilon_F^{L\infty} = \left| 1 - \frac{\sum_g (F_u \psi_u^{(e+1)})^g}{\sum_g (F_u \psi_u^{(e)})^g} \right| < \epsilon_F, \quad u = 1, \dots, U, \quad \mathbf{r} \in D_{u,R}, \quad (12.50)$$

$$\varepsilon_\psi^{L\infty}(\xi) = \left| 1 - \frac{\int_{2\pi^-} d\Omega |\mathbf{n} \cdot \Omega| \psi_u^{-,(e+1)}(\xi)}{\int_{2\pi^-} d^2\Omega |\mathbf{n} \cdot \Omega| \psi_u^{-,(e)}(\xi)} \right| < \epsilon_\psi, \quad u = 1, \dots, U, \quad \xi \in \partial X_u^-, \quad (12.51)$$

where $\epsilon_k, \epsilon_F, \epsilon_\psi$ are user-defined parameters, while the tolerance in 12.49 is fixed to 0.1ppm of error on boron concentration.

12.7.11 . Critical Boron Concentration

In IDT, the critical boron concentration is computed by means of a linear formula (Sect. 5.2.2), i.e. the Newton method. More in general, let C_B^{ref} be the boron concentration such that $k = k^{ref}$.¹¹ C_B is a function of k , namely $C_B = C_B(k)$. Assuming that $C_B \in \mathcal{C}^2[0, \nu_f]$, where ν_f is the average number of neutrons per fission ($\nu_f \sim 2 - 3$), as well as the maximum allowed value for k , C_B^{ref} is determined by successive outer iterations, namely

$$C_B^{(e+1)} = C_B^{(e)} + \frac{C_B^{(e)} - C_B^{(e-1)}}{k^{(e+1)} - k^{(e)}} (k^{ref} - k^{(e+1)}), \quad (12.52)$$

provided $C_B^{(0)} > 0$ (Sect. 12.7.3). In IDT, no boron concentration search is performed until the relative difference on the fission source has reached 1%, i.e. with the relative difference defined 12.48 $\epsilon_k^{rel} < 0.01$. This allows to prevent the appearance of error oscillations. If the boron concentration is such that

$$\left| C_B^{(e+1)} - C_B^{(e)} \right| < C_B^{(e+1)} \cdot \epsilon_k, \quad (12.53)$$

a logical variable, namely lg_B , is set equal to false, in order to inhibit the update of boron concentration, which is considered as converged.

12.7.12 . Neutronics/Thermal-Hydraulics coupling

The calculation scheme outlined in Algo. 5 and in 12.29 proposes a strong coupling between neutronics and thermal-hydraulics. Normally, at each outer iteration, the temperature field is updated. A dedicated python script, namely IDT2THEDI, allows data exchange between IDT and THEDI.

In particular, IDT produces an updated power field at each iteration, which feeds into the thermal-hydraulic problem (Sect. 12.7.13). The latter in turn, based on the new neutronic power distribution and the boundary conditions imposed in a specific input file (Sect. 12.7.14), generates new temperature fields for the fuel and the moderator as well as an updated moderator density map. At this point, the python script compares the temperature and density distributions between two successive multiphysics iterations, for each subdomain (Algo. 5, line 26). The difference between the respective TH fields can lie below or above a given threshold. Based on this, IDT2THEDI communicates, via file, the information to IDT, for each subdomain. If the subdomain under consideration has not undergone appreciable changes in terms of TH distributions, IDT sets a logical variable to false. If the boron concentration has not changed as well, IDT does not execute XSTOOL, i.e., the macroscopic cross sections are not updated, and the CP coefficients are not recomputed, for the given subdomain. Conversely, if the boron concentration has changed or if IDT2THEDI enforces a new calculation of cross sections for the current subdomain, IDT executes, by bash commands, XSTOOL which produces a new ASCII file of multigroup nuclear data (Algo. 5, line 29).

IDT compares the new physical HCCs, which contain the new cross sections, with the reference cells from the previous iteration (Algo. 5, line 30, Chap. 9). If the temperature change has generated a 'perturbation' $\delta\Sigma(\mathbf{r}) < 0$, the coefficients must be forcibly recalculated, in order to prevent the appearance of negative sources (Sect. 9.2). On the contrary, if this does not happen, there is no need to recalculate the coefficients: one may just update the 'perturbation'.

¹¹In practical problems, one may be interested in computing the critical boron concentration. In this case, $k^{ref} = 1$.

The change induced by thermal-hydraulics will thus be taken into account, along inner iterations, via Neumann series developments (Eq. (9.10)). The following paragraphs offer a more detailed picture of the key points of the N/TH coupling. Finally, it should be noted that IDT offers the user a further degree of freedom, i.e., the possibility to ‘weaken’ the coupling, performing the thermal-hydraulic calculation every \mathcal{N} outer iterations. This allows to converge the fission source and k -eigenvalue for \mathcal{N} outer iterations, without recomputing the cross sections and CP coefficients, and to perform the calculation of the new temperature field less frequently. This approach is also tested in Sect. 12.8.

12.7.13 . Generation of Power Map input for TH simulation

The power map is updated, by setting

$$P_f^{(e+1)}(\mathbf{r}) = \int dE \int d\Omega (K_t \Sigma_t)(\mathbf{r} \in f, E) \psi^{(e+1)}(\mathbf{r} \in f, \Omega, E), \quad (12.54)$$

$$P_{nf}^{(e+1)}(\mathbf{r}) = \int dE \int d\Omega (K_t \Sigma_t)(\mathbf{r} \notin f, E) \psi^{(e+1)}(\mathbf{r} \notin f, \Omega, E), \quad (12.55)$$

where

- K_t is the KERMA (Kinetic Energy Released in MAtter), due to a nuclear reaction, and is provided by XSTOOL;
- P_f is the local power generated in the fuel;
- P_{nf} is local power released in all media that do not contain fissionable isotopes.

IDT generates a file, namely ‘channel_power.txt’, containing the power map over the whole reactor, which is organized as indicated in Tab. 12.5. The file contains a number of rows equal to the number of HCCs composing the whole geometry. The position of each HCC is indicated by three integer numbers, namely (x, y, z) , followed by the power generated in the fuel (if present), namely P_f , and the power generated elsewhere, i.e., P_{nf} , within the node (x, y, z) .

12.7.14 . IDT2THEDI. Thermal-Hydraulic feedback by THEDI

THEDI (App. A) is a multi-1D thermal-hydraulic solver of APOLLO3®. It may be linked as a Python/C++ library by external codes. In order to perform coupled N/TH multiphysics simulations, IDT may execute, by bash commands, a simple Python script, namely IDT2THEDI, importing THEDI and the water $p - T$ diagram. IDT2THEDI scans three files, i.e.,

- the sub-channels geometry input file, generated by IDT, for N/TH coupling, to retrieve all HCC geometrical features, for each TH sub-channel (Sect. 12.7.4);
- ‘channel_power.txt’, containing the power map (Sect. 12.7.13);
- ‘channel_input.txt’, providing a complementary set of user-defined entries, comprising
 - a) the water inlet temperature ($T_{in} [^{\circ}C]$),
 - b) the pressure ($p [Pa]$),

x	y	z	P_f	P_{nf}
-----	-----	-----	-------	----------

Table 12.5: Power-map out/input file, generated by IDT, for N/TH coupling.

- c) the flow rate (\dot{m} [kg/s]),
- d) the total power P_{tot} [W],
- e) the laminar/turbulent intervals, expressed in terms of Re (Reynolds' number, [113]),
- f) the local pressure drop due to a spacer grid,
- g) the radial-mesh ratios for ρ_f , ρ_c^{in} and ρ_c^{out} [cm] (Sect. 12.7.4),
- h) the path to the water $p - T$ diagram.

Once acquired the aforementioned sets of raw data, IDT2XSTOOL computes

- The number of HCCs within each sub-channel;
- The axial mesh of each sub-channel. Note that, in principle, IDT2XSTOOL may support non-conformal sub-channel discretizations along the z-axis, but this capacity is not exploited yet, as HCCs boundaries have to be conformal in IDT;
- The cross-sectional area of the flow and the wetted perimeter, for each sub-channel and z-step, which allow for the calculation of the hydraulic diameter;
- A logical variable, defining the presence of a spacer grid, within a given z-step, for the considered sub-channel;
- A set of variables, depending on the HCC order number and sub-channel identifier, expressing respectively the presence of nuclear fuel (logical), the cell weight (or, more precisely, its inverse, which is an integer number), the radial discretization (floats) and the Cartesian coordinates (x, y, z) defining the location of the cell;
- The cross-sectional mass flow of each sub-channel,

$$\dot{m}_{ch} = \dot{m} \cdot \frac{A_{ch}}{A_{core}}, \quad (12.56)$$

where A_{ch} and A_{core} are the sub-channel and the core cross-sectional areas of the flow, respectively, which are easily derived from the above-mentioned quantities.

As anticipated in Sect. 12.7.13, 'channel_power.txt' provides, for each HCC, two different types of power, i.e., the one generated in the fuel, P_f , and the one produced in non-fissile materials, P_{nf} . In particular, for fuel cells, the latter is redistributed between the moderator and the fuel-clad, in proportions dependent on their cross-sectional area,

$$q'_f(x, y, z) = \frac{1}{\Delta(z)} P_{f,t}(x, y, z) \cdot w_{ch}(x, y, z) \quad (12.57)$$

$$q'_c(x, y, z) = \frac{1}{\Delta(z)} P_{nf,t}(x, y, z) \cdot w_{ch}(x, y, z) \cdot \frac{A_c(x, y, z)}{A_w(x, y, z) + A_c(x, y, z)} \quad (12.58)$$

$$q'_w(x, y, z) = \frac{1}{\Delta(z)} P_{nf,t}(x, y, z) \cdot w_{ch}(x, y, z) \cdot \frac{A_w(x, y, z)}{A_w(x, y, z) + A_c(x, y, z)} \quad (12.59)$$

which represent the linear power produced in the fuel, in the clad and in the moderator, respectively, and

$$q'_{nf}(x, y, z) = \frac{1}{\Delta(z)} P_{nf,t}(x, y, z) \cdot w_{ch}(x, y, z) \quad (12.60)$$

denoting the linear power generated in a cell that does not contain fuel (e.g., a control-rod, a guide-tube, a water-gap cell, etc.). More specifically,

- $\Delta(z)$ denotes the z-step length, for all HCCs with axial coordinate equal to z ;
- $P_{f,t}$ and $P_{nf,t}$ are normalized to the total power (defined by the user, P_{tot});
- $w_{ch}(x, y, z)$ is the cell weight for a given sub-channel;
- $A_c(x, y, z)$ and $A_w(x, y, z)$ are respectively the cross-sectional areas of the clad and of the moderator, for a fuel cell in position (x, y, z) .

The prior quantities are injected into THEDI, which, provided the boundary conditions (T_{in}, p, \dot{m}_{ch}) and few other parameters (mostly related to the material properties, such as the thermal conductivity and capacity), returns the temperature and density map for every given sub-channel, in steady-state conditions. In particular, one has $T_f = T_f(x, y, z, r)$, r being the radius, $T_w = T_w(z)$ and $\rho_w = \rho_w(z)$. The fuel output is rearranged, so that one has a single value of fuel temperature, for a given cell position, namely $T_f(x, y, z)$, by computing the Rowlands average over the whole set of fuel radii, [78], [114]. The obtained TH map, containing all effective fuel temperatures and moderator temperatures and mass densities, is used to update the whole set of XSTOOL input files, namely 'subdo_ijk.xst', whose structure is detailed in Sect. 12.3.5. IDT2THEDI may also produce a collection of files, with 'boolean' extension, of the kind 'subdo_ijk.True'. The condition for this to occur, for a given GCU, is defined by

$$\exists\{x, y, z\} \text{ such that } \delta T_f(x, y, z) > \epsilon_f \text{ or } \delta \rho_w(k) > \epsilon_w, \quad (12.61)$$

where

$$\delta T_f(x, y, z) = T_f^{(e+\mathcal{N})}(x, y, z) - T_f^{(e)}(x, y, z) \quad (12.62)$$

$$\delta \rho_w(k) = \rho_w^{(e+\mathcal{N})}(k) - \rho_w^{(e)}(k) \quad (12.63)$$

and ϵ_f, ϵ_w are a fixed tolerance, whose definition is addressed in the upcoming section. As mentioned in Sect. 12.7.12, \mathcal{N} defines the number of outer iterations between two consecutive TH field updates.

12.7.15 . Macroscopic Cross Section Update by XSTOOL

The updated XSTOOL input file, for a given subdomain, namely 'subdo_ijk.xst', may or may not be used for macroscopic cross-section update, by XSTOOL, depending on the occurrence of the condition defined by Eq. (12.61). Here, two parameters are introduced, ϵ_f and ϵ_w , defining the admitted fuel temperature and moderator density errors, respectively. The results provided in Sect. 12.8 are obtained by setting $\epsilon_f = 5^\circ C$ and $\epsilon_w = 1 \text{ pcm}$.

If Eq. 12.61 is verified, the macroscopic cross libraries are re-initialized, as in Sect. 12.7.5. In this case, $\delta\Sigma(\mathbf{r})$ is recomputed for Neumann iterations (Sect. 9.2). Conversely, should the temperature difference be negligible, XSTOOL is not run, for the given GCU, thus lowering the simulation runtime.

12.7.16 . CP coefficients recalculation

As mentioned in Sect. 12.7.12, once recomputed the macroscopic cross sections library of the current subdomain, IDT updates the 'perturbations' $\delta\Sigma(\mathbf{r})$ between the current physical cells, containing a new set of cross sections, and the past physical reference cells (Sect. 9.2). This difference is required not to be negative, in order to prevent the possible onset of negative neutron sources. If this is not the case, even for a single region or energy group, the entire set of probability matrices is recalculated for the given subdomain.

12.8 . NUMERICAL RESULTS

In the following, a few numerical tests are presented. First, let us ensure that energy condensation does not deteriorate the numerical accuracy of the solution. If observed on real-sized systems, such as a 3D fuel assembly, this result may validate the assumption that there is no need to solve an equivalence problem to correct the reaction rates. For the moment, due to lack of time, only a simple test, involving a small hexagonal pattern, is proposed. Once verified the numerical accuracy of the solution obtained by a 26-group cross section library (Sect. 12.8.1), a few multiphysics calculations including the thermal feedback within each power iteration (or within a subset of power iterations) are presented (Sects. 12.8.2-12.8.3). Differently from other works [115], the idea is not to converge the neutron flux between one multiphysics iteration and another, but rather to resort to tightly-coupled N/TH iterations. The preliminary results presented in Sects. 12.8.2-12.8.3 highlight that the proposed scheme requires only few outer iterations (roughly around 30), while calculations implementing weak N/TH coupling typically require few hundreds of outer iterations, to converge the multiphysics solution to the required precision.

12.8.1 . Energy condensation - Verification test based on a small hexagonal pattern

This test is aimed at verifying whether the energy condensation of the macroscopic cross sections, performed by XSTOOL, may lead to an equivalence problem, [5][38]. Similarly to Eq. (12.1), as the macroscopic cross sections are weighted by the angular flux, an angular dependence is artificially introduced in the multigroup data library. Moreover, the upstream and downstream codes, namely the one generating the parametric cross-section library, i.e., APOLLO3® TDT-MOC, and the one solving neutron transport over the whole core, i.e., IDT, do not use the same discretization. In particular, the discretization of the leakage term may induce a significant error, when condensing the macroscopic cross sections. In order to lower the aforementioned problems, SPH equivalence is generally used to correct the reaction rates, [116]. The present section provides a comparison between the results obtained by condensing the cross sections to 26 and 30 groups, respectively, and the ones derived with no energy condensation, i.e., in this specific case, 281 energy groups.

Let us consider a case study involving a 3D hexagonal pattern, with specular reflection boundary conditions at all edges, composed of 19 fuel pins of unit height, arranged as in Fig. 12.17. Let us search for the critical boron concentration, without including the thermal feedback. The external reference library is provided by GALILEE and contains 281 energy groups. The parametric EPL-SSF library is produced by APOLLO3®, by CP Fine-Structure Self-Shielding. In order to simulate a hexagonal pattern, we use quarter cells, coaxial to one of the HCC edges, with sides equal to 0.6375 cm and 0.55209 cm, in order to generate longitudinal and transverse pitches of 1.275 cm and 1.10418 cm.

In the following, three simulations are considered, using 281, 30 and 26-group macroscopic cross-section libraries, respectively, with P_1 scattering, generated by XSTOOL through polar interpolation of the microscopic cross sections (and subsequent condensation), and injected into IDT for neutron transport. The boron concentration is initialized to $C_B^{(0)} = 0.01$ ppm, in all cases. The spatial discretization includes 10×12 HCCs, containing a quarter of cylinder, with no inner Cartesian mesh. Linear short characteristics are applied to determine the spatial solution of the Boltzmann equation. The solid angle is discretized into 10 directions per octant, using a S_8 Level-Symmetric angular quadrature formula. For all calculations, we use the following

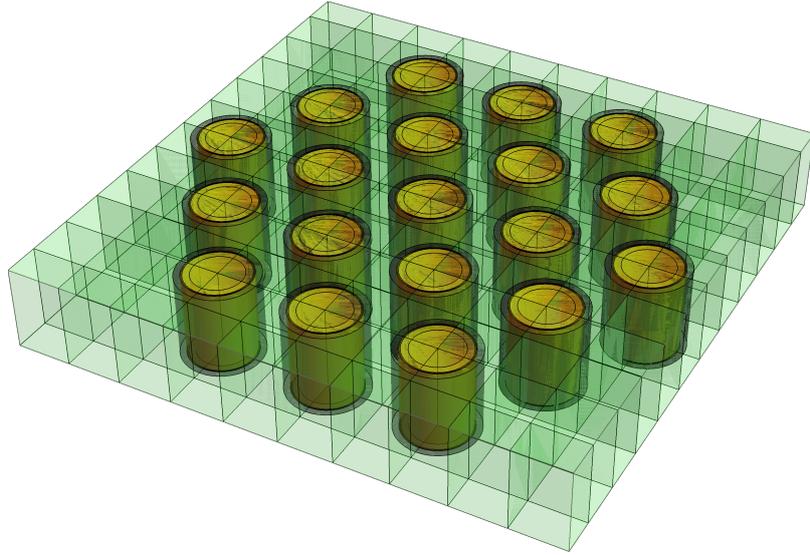


Figure 12.17: Hexagonal pattern geometry

Nb. of groups	Eigenvalue	Critical boron concentration (ppm)	Relative error (%)
26	1.000006	336.06	5
30	1.000027	321.15	0.4
281	0.9999965	319.86	—

Table 12.6: Computed eigenvalues and critical boron concentrations, using 26/30/281-group macroscopic cross section libraries. In the last column, the relative error on the critical boron concentration, assuming the calculation with no energy condensation (i.e., 281 groups) as the reference.

convergence criteria:

- Relative error on the k -eigenvalue: $\epsilon_k = 1 \text{ pcm}$;
- L^∞ -error on the fission source: $\epsilon_F^{L^\infty} = 10 \text{ pcm}$;
- Point-wise relative error on the angular flux moments: $\epsilon_{in} = 1 \text{ pcm}$;
- L^∞ -error on the interface flux: $\epsilon_\psi^{L^\infty} = 1 \text{ pcm}$;
- Relative error on the boron concentration: 1 ppm .

Provided the calculation using non-condensed macroscopic cross sections may be taken as reference, let us evaluate the accuracy of the numerical solutions with 26 and 30 energy groups.

Table 12.6 shows the computed eigenvalue and associated critical boron concentration, for 26, 30 and 281 energy groups, as well as the relative error on the computed critical boron concentration. Figs. 12.18-12.19 display the relative error on the fuel power distributions obtained by 26 and 30 energy groups. The fuel power relative error is at most 0.221% and 0.127%, for $H = 26$ and $H = 30$, respectively, H being the number of condensed groups. Finally, for the sake of completeness, Tab. 12.7 provides the CPU time for each calculation.

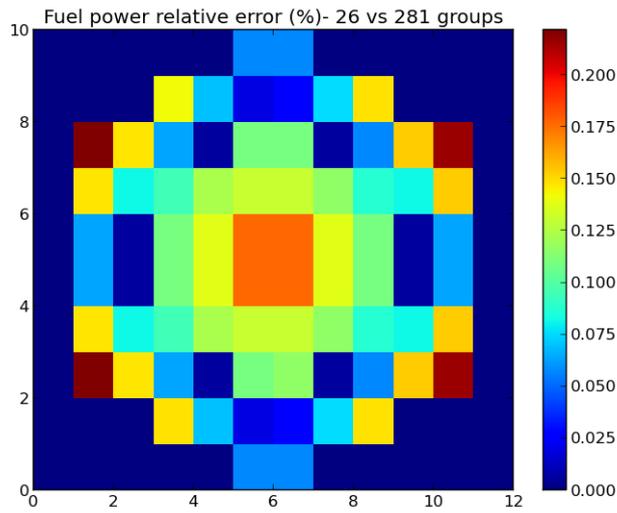


Figure 12.18: 26 vs 281 energy groups: fuel power relative error (%).

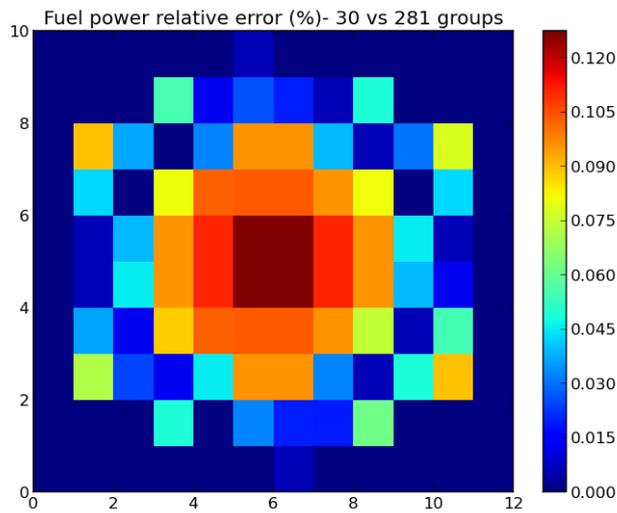


Figure 12.19: 30 vs 281 energy groups: fuel power relative error (%).

Nb. of groups	CPU time [min]
26	16
30	23
281	191

Table 12.7: Simulation runtime. The spatial domain is divided into 3×4 subdomains and run on 12 OpenMP threads.

It should be noted that the proposed tests only provide very preliminary results. More in-depth analyses, based on real-sized assemblies, will be necessary to demonstrate that no equivalence procedure is actually needed for the presented calculation scheme.

12.8.2 . Neutronics/Thermal-Hydraulics coupling - 3D 17x17 UOX/B₄C lattice

Let us consider a 17x17 3D assembly, with a 0.02 *cm* water blade and 0.0194 *cm* zirconium-alloy E110 spacer grids, powered by UOX fuel, regulated via *B₄C* control rods and moderated and cooled by light water. Specular reflection boundary conditions are imposed on the radial edges, while vacuum boundary conditions are set on the bottom and top borders. Table 12.8 describes the axial geometry and material composition of the considered benchmark, whereas Fig. 12.20 provides a sketch of the radial geometry. The guide tube is located in the centre. All control rods are inserted up to $z_{tip} = 330$ *cm*.

Fuel rod	Control rod	Guide tube
400 <i>cm</i>	400 <i>cm</i>	400 <i>cm</i>
Axial Reflector	B_4C	Axial Reflector
380 <i>cm</i>	380 <i>cm</i>	380 <i>cm</i>
Fuel pin-cell + Grid	B_4C + Guide Tube + Grid	Guide Tube + Grid
370 <i>cm</i>	370 <i>cm</i>	370 <i>cm</i>
Fuel pin-cell	B_4C + Guide Tube	Guide Tube
330 <i>cm</i>	330 <i>cm</i>	330 <i>cm</i>
Fuel pin-cell + Grid	Guide Tube + Grid	Guide Tube + Grid
320 <i>cm</i>	320 <i>cm</i>	320 <i>cm</i>
Fuel pin-cell	Guide Tube	Guide Tube
280 <i>cm</i>	280 <i>cm</i>	280 <i>cm</i>
Fuel pin-cell + Grid	Guide Tube + Grid	Guide Tube + Grid
270 <i>cm</i>	270 <i>cm</i>	270 <i>cm</i>
Fuel pin-cell	Guide Tube	Guide Tube
230 <i>cm</i>	230 <i>cm</i>	230 <i>cm</i>
Fuel pin-cell + Grid	Guide Tube + Grid	Guide Tube + Grid
220 <i>cm</i>	220 <i>cm</i>	220 <i>cm</i>
Fuel pin-cell	Guide Tube	Guide Tube
180 <i>cm</i>	180 <i>cm</i>	180 <i>cm</i>
Fuel pin-cell + Grid	Guide Tube + Grid	Guide Tube + Grid
170 <i>cm</i>	170 <i>cm</i>	170 <i>cm</i>
Fuel pin-cell	Guide Tube	Guide Tube
130 <i>cm</i>	130 <i>cm</i>	130 <i>cm</i>
Fuel pin-cell + Grid	Guide Tube + Grid	Guide Tube + Grid
120 <i>cm</i>	120 <i>cm</i>	120 <i>cm</i>
Fuel pin-cell	Guide Tube	Guide Tube
80 <i>cm</i>	80 <i>cm</i>	80 <i>cm</i>
Fuel pin-cell + Grid	Guide Tube + Grid	Guide Tube + Grid
70 <i>cm</i>	70 <i>cm</i>	70 <i>cm</i>
Fuel pin-cell	Guide Tube	Guide Tube
30 <i>cm</i>	30 <i>cm</i>	30 <i>cm</i>
Fuel pin-cell + Grid	Guide Tube + Grid	Guide Tube + Grid
20 <i>cm</i>	20 <i>cm</i>	20 <i>cm</i>
Axial Reflector	Axial Reflector	Axial Reflector

Table 12.8: 17x17 tridimensional UOX/ B_4C lattice: axial geometry and composition.

The external reference library is provided by GALILEE and contains 281 energy groups. The parametric EPL-SSF library is recycled from the hexagonal lattice described in Sect. 12.8.1. Cross sections are interpolated and condensed to 26 energy groups, by XSTOOL, using APOLLO3® TDT-MOC neutron flux, from the lattice calculation, and include P_1 scattering matrices. No critical boron concentration search is performed.

Regarding neutron transport, the solid angle is discretized by a S_8 Level-Symmetric angular quadrature formula. The spatial domain is subdivided into $19 \times 19 \times 40$ HCCs, with no Cartesian refinement. The total number of cells is equal to 14440. The total number of media is 72868. The solution is obtained by Linear Short Characteristics. DDM is also applied, with $3 \times 3 \times 10$ symmetrically-distributed subdomains. Concerning the error criteria, the k-eigenvalue and the fission integral are required to converge (at least) up to 1 pcm and 100 pcm, respectively, while the tolerance on the inner loop is set equal to 10 pcm.

Thermal-Hydraulics is solved using a single TH-channel, for stationary single-phase regime, using the following boundary conditions:

- Water inlet temperature, $T_{in} = 550 \text{ K}$;
- Pressure, $p = 155 \text{ MPa}$;
- Flow rate, $\dot{m} = 96 \frac{\text{kg}}{\text{s}}$.

The total power P_{tot} is normalized to 20 MW. The TH channel is discretized by 40 axial planes of equal size (10 cm). The fuel pin-cells comprise

- 4 rings for the fuel, distributed according to the following ratios: 0.4, 0.3, 0.2, 0.1 (i.e., 40%, 30%, 20% and 10% of the total fuel radius, ρ_f);
- 4 rings for the helium gap, with ratios 0.4, 0.3, 0.2, 0.1;
- 3 rings for the Zr-clad, with ratios 0.5, 0.3, 0.2.

In the following, two simulations are proposed, sharing the same set of N/TH input parameters, except for the aforementioned number \mathcal{N} (Sect. 12.7.12), determining the frequency at which the temperature map is updated, along the outer loop. In particular, for the simulation labelled as 'Strong', \mathcal{N} is set equal to 1 (i.e., the temperature distribution is updated every outer iteration), whereas, for the simulation marked as 'Weak', \mathcal{N} is equal to 5. In reality, both tests should be referred as 'strong', as the multigroup transport iterations are not converged to the required precision, before solving thermal-hydraulics. This abuse of notation only serves to underline that the proposed N/TH coupling scheme can be 'weakened', depending on user's requirements, by adjusting a simple input parameter.

The two calculations ('Strong' and 'Weak') are compared to each other, in order to test the convergence of the algorithm to the required precision, as the frequency of the multiphysics iterations varies along the outer loop. The numerical results are expected to be the same, with an error of the order of the required precision. Note that no comparison with a Monte Carlo or deterministic reference has been conducted yet, due to time constraints.

Let us commence by comparing the convergence rate of the two simulations. Figure 12.21 shows the error on the k-eigenvalue, for $\mathcal{N} = 1$ and $\mathcal{N} = 5$, respectively, as a function of the

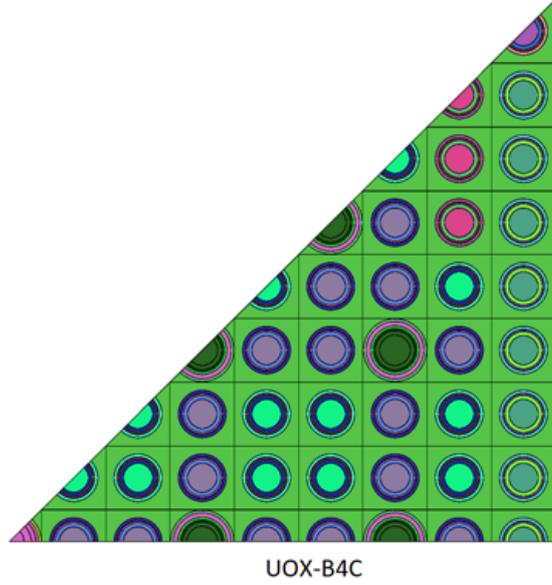


Figure 12.20: 17x17 tridimensional UOX/ B_4C lattice: one eighth of the radial geometry.

number of outer iterations, in semi-logarithmic scale. ‘Strong’ and ‘Weak’ converge in just 27 and 45 outer iterations, respectively. One can notice that, ‘Weak’ may show large cusps at iterations with index number equal to $k \cdot \mathcal{N} + 1$, with $k \in \mathbb{N}^*$, due to a possible change of cross sections (and CP coefficients), induced by the thermal feedback.

Figure 12.22 compares the two calculations, in terms of error on the fission source (defined as in Eq. (12.50)), as a function of the number of outer iterations. The curve corresponding to $\mathcal{N} = 1$ shows a trend close to an exponential decay, starting from $e = 4$, with e being the outer iteration index. The fission source error goes below 100 pcm (the required accuracy) in only 2 iterations, for both calculations.

Finally, Fig. 12.23 compares ‘Strong’ and ‘Weak’ in terms of inner loop error, defined as the point-wise relative error on the angular flux moments,

$$\varepsilon_{in} = \max_{\text{for } k < K \text{ and } |l| \leq k \text{ and } \mathbf{r} \in D} \left| 1 - \frac{\phi_{k,l}^{(i+1)}(\mathbf{r})}{\phi_{k,l}^{(i)}(\mathbf{r})} \right|, \quad (12.64)$$

where i is the inner iteration index.

Based on our experience, in order to converge the multiphysics iterations to the required precision and have results within the error criteria, the tolerance on the fuel temperature, ε_{T_f} , and the one on the moderator density, ε_{ρ_m} , have to be set equal to $5^\circ C$ and 1 pcm, respectively. The number of multiphysics iterations (i.e., outer iterations including the thermal feedback) necessary to converge thermal-hydraulics, within the required error, amounts to 27 iterations, for $\mathcal{N} = 1$, and 9 iterations (distributed one every 5 outer iterations), for $\mathcal{N} = 5$. Table 12.9 shows the computed eigenvalues for $\mathcal{N} = 1$ (‘Strong’) and $\mathcal{N} = 5$ (‘Weak’) as well as the relative difference between the two, assuming ‘Strong’ as the reference.

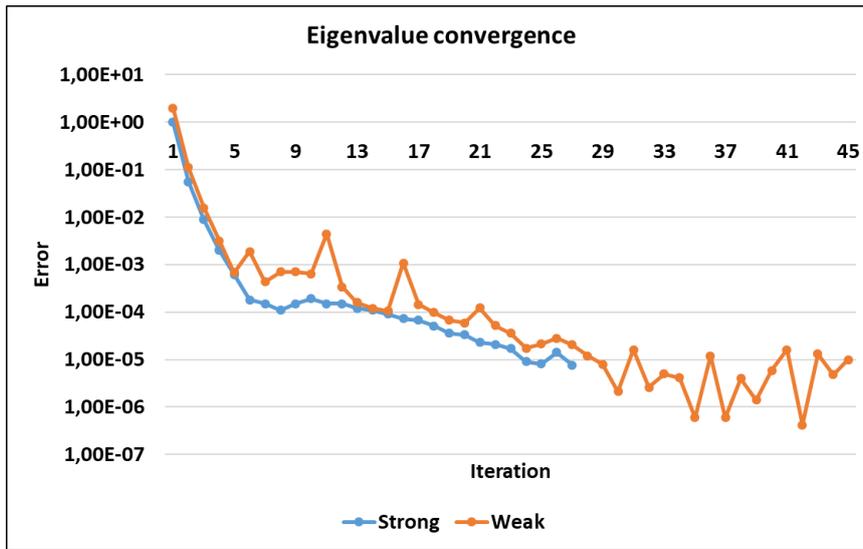


Figure 12.21: k -eigenvalue error vs iteration number, for $\mathcal{N} = 1$ ('Strong') and $\mathcal{N} = 5$ ('Weak').

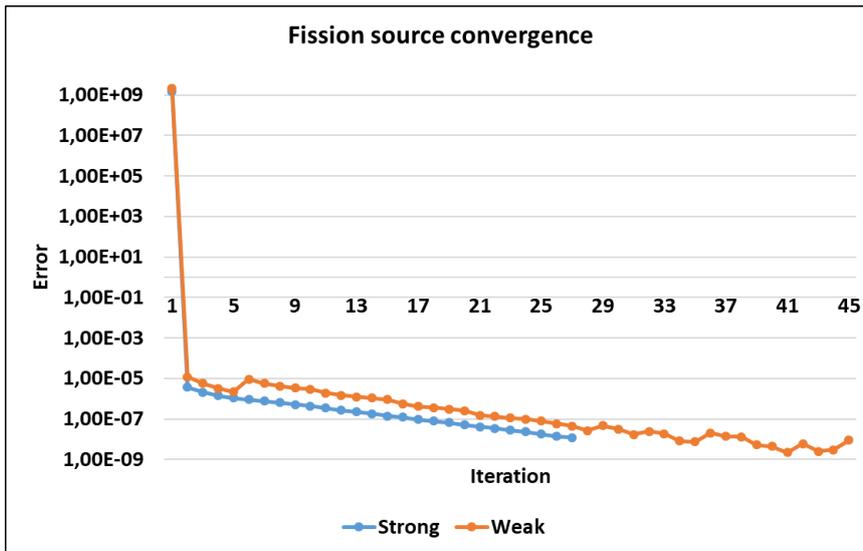


Figure 12.22: Fission source error vs iteration number, for $\mathcal{N} = 1$ ('Strong') and $\mathcal{N} = 5$ ('Weak').

Eigenvalue of 'Strong'	Eigenvalue of 'Weak'	Relative difference (<i>pcm</i>)
0.9652548	0.9652081	4.8

Table 12.9: Computed eigenvalues for $\mathcal{N} = 1$ and $\mathcal{N} = 5$ and relative difference between the two, calculated by taking the simulation corresponding to $\mathcal{N} = 1$ as the reference.

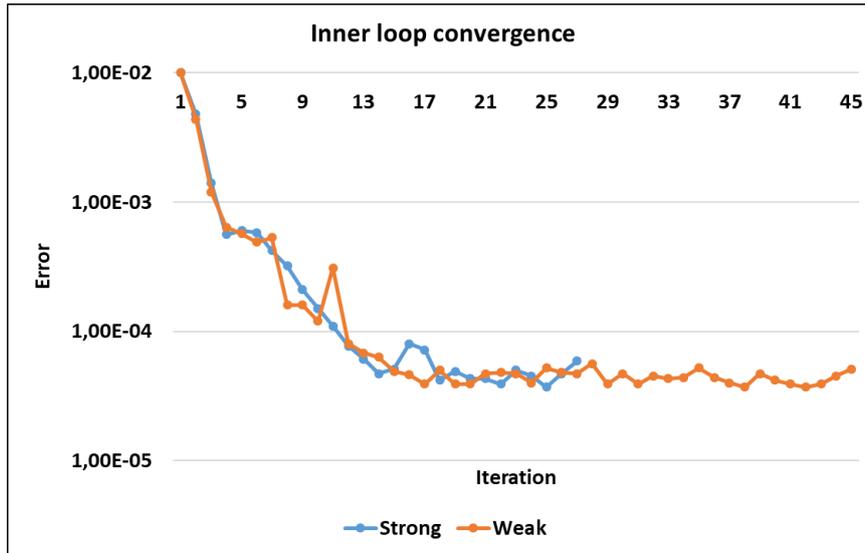


Figure 12.23: Inner loop error vs iteration number, for $\mathcal{N} = 1$ ('Strong') and $\mathcal{N} = 5$ ('Weak').

Let us now compare 'Strong' and 'Weak', in terms of fuel temperature, moderator temperature and fuel power distributions. For the sake of conciseness, only few results are displayed for the fuel temperature and power maps. First, let us observe that the hot spot is located at $x = 6, y = 5, z = 11$, where x, y and z denote the (integer) coordinates defining the position of a HCC. For symmetry reasons, the same temperature and power are observed in other 7 cells of the same plane. Let us consider the axial fuel power trend of the rod located at $x = 6, y = 5$, and compare the curves corresponding to $\mathcal{N} = 1$ and $\mathcal{N} = 5$ (Figs. 12.24-12.25). Fig. 12.24 displays the fuel power as a function of z . One can notice that, due to the B_4C control rods, the fuel power is skewed downwards. Fig. 12.25 shows the relative difference between 'Weak' and 'Strong', in absolute value, assuming 'Strong' as the reference. The maximum error barely exceeds 0.6%.

Let us compare the detailed fuel-pin power distribution, for $\mathcal{N} = 1$ and $\mathcal{N} = 5$, in the planes $z = 11$ and $z = 3$. The former is the plane releasing the largest amount of power (about 1.34 MW), while the latter is the plane showing the largest pin-wise relative difference between 'Strong' and 'Weak'. Figs. 12.26-12.27 show the pin-wise relative difference, assuming 'Strong' as the reference. The largest difference is observed in the plane $z = 3$ and is significantly less than 1%.

Regarding the fuel temperature distribution, let us compare the results for 'Strong' and 'Weak', in the planes $z = 11$, containing the hot spot, and $z = 5$, where the largest difference between the two calculations is observed (Fig. 12.28-12.29). The latter amounts to approximately 1.6 K (Fig. 12.29).

Finally, let us display the moderator temperature distribution for 'Strong' and 'Weak' and analyse the difference between the two. As shown in Fig. 12.30, the temperature of the moderator raises up to 587 K. The maximum difference between the two calculations is approximately 0.028 K (Fig. 12.31).

Concerning the simulation runtime, the results reported here are derived by OpenMP paral-

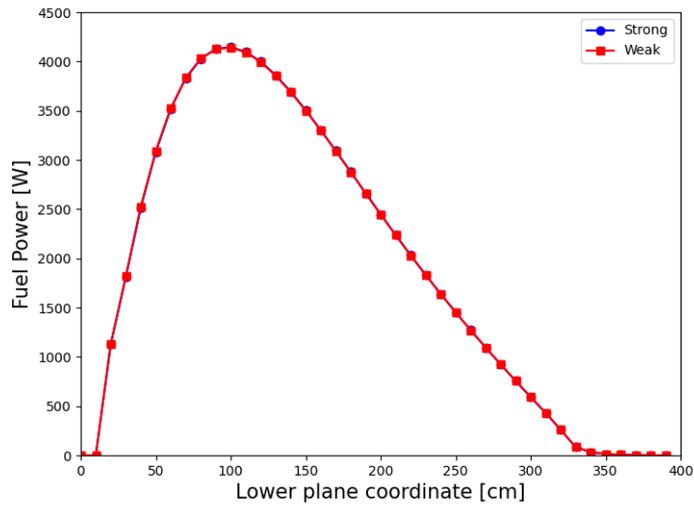


Figure 12.24: 'Strong' vs 'Weak': fuel power distribution for the rod in position $x = 6, y = 5$.

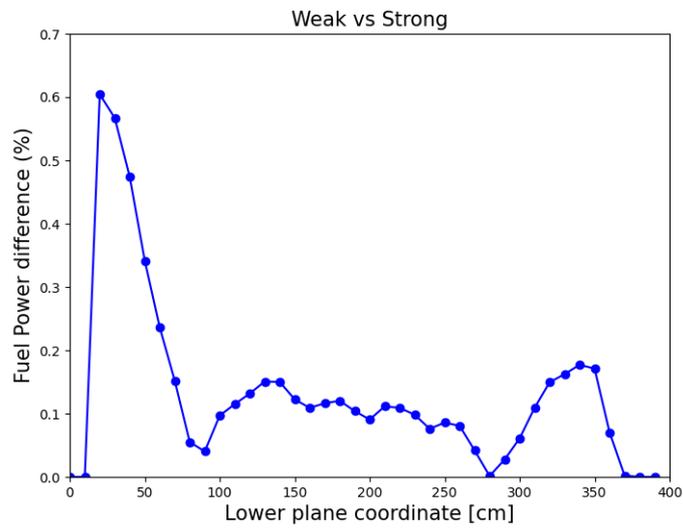


Figure 12.25: 'Strong' vs 'Weak': fuel power relative difference (%) for the rod in position $x = 6, y = 5$, assuming 'Strong' as the reference.

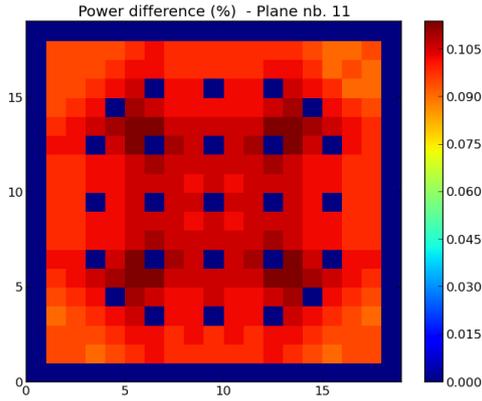


Figure 12.26: Fuel power relative difference, between $\mathcal{N} = 1$ and $\mathcal{N} = 5$, assuming $\mathcal{N} = 1$ as the reference, in the plane $z = 11$.

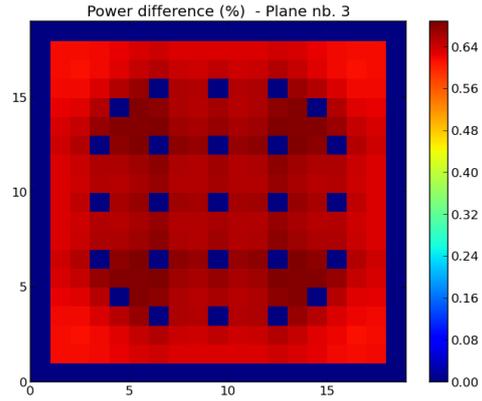


Figure 12.27: Fuel power relative difference, between $\mathcal{N} = 1$ and $\mathcal{N} = 5$, assuming $\mathcal{N} = 1$ as the reference, in the plane $z = 3$.

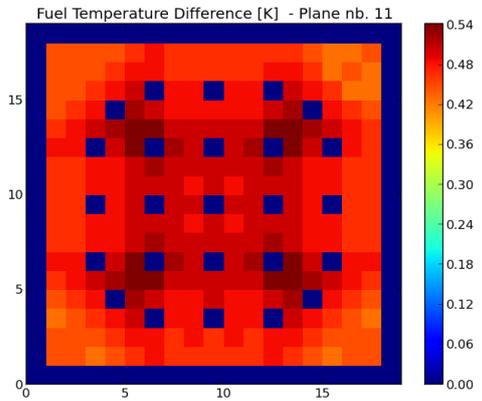


Figure 12.28: Fuel temperature difference [K], between $\mathcal{N} = 1$ and $\mathcal{N} = 5$, in the plane $z = 11$.

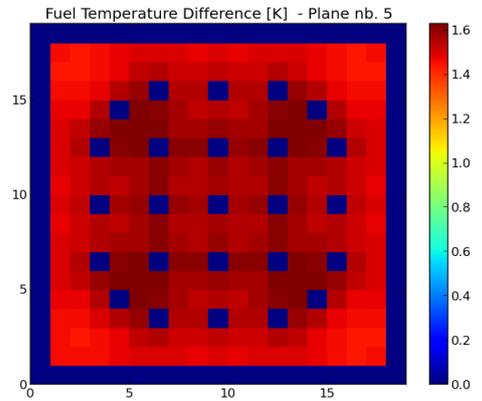


Figure 12.29: Fuel temperature difference [K], between $\mathcal{N} = 1$ and $\mathcal{N} = 5$, in the plane $z = 5$.

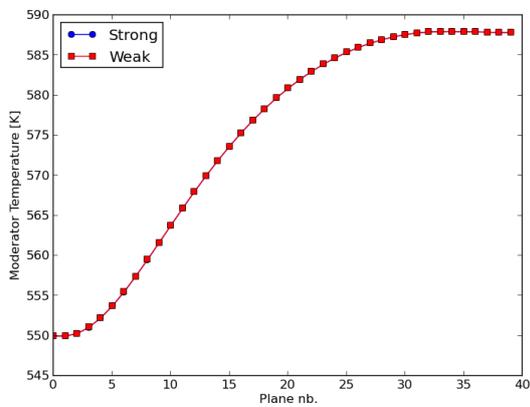


Figure 12.30: Moderator temperature [K]: comparison between 'Strong' and 'Weak'.

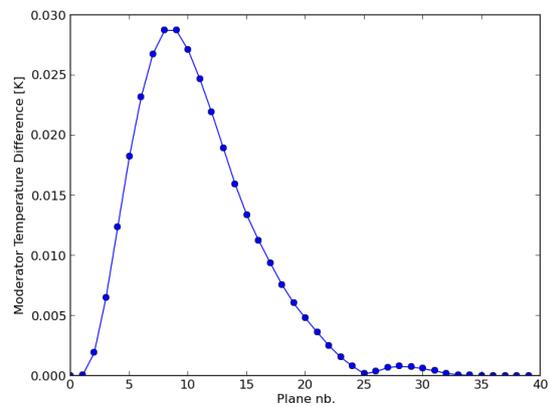


Figure 12.31: Moderator temperature difference [K] between 'Strong' and 'Weak', as a function of z .

Calculation	'Strong'	'Weak'
Total CPU time [min]	103	87
% for coefficients	24	8
% for cross sections	3.6	1.3

Table 12.10: Total CPU time [min] and percentage of time spent to recompute the probability matrices and the macroscopic cross section libraries.

lelism, with a maximum of 48 threads. Tab. 12.10 shows the total CPU time for $\mathcal{N} = 1$ and $\mathcal{N} = 5$, as well as the percentage of time spent to recompute the CP coefficients (by IDT) and the macroscopic cross sections (by XSTOOL), for each case. Finally, the time spent in thermal-hydraulic section (i.e., time spent by IDT2THEDI + temperature map calculation by THEDI) amounts to very few seconds, for each multiphysics iteration.

12.8.3 . Multiphysics iterations, with critical boron concentration search - 17x17 UOX/B₄C 3D lattice

Let us consider the same case study as the one in Sect. 12.8.2. The control-rod tips are now located at $z_{tip} = 370 \text{ cm}$ (Tab. 12.8), so that the system, without boron, is super-critical. In the following, we search for the boron concentration that makes k equal to 1. The thermal-hydraulic feedback is still present, and is included at each outer iteration, by setting \mathcal{N} equal to 1.

All discretization parameters are unchanged with respect to the case study analyzed in Sect. 12.8.2. The total number of media is 72580, while the total number of cells is 14440. The subdomains macroscopic cross section libraries are regenerated, whenever required by the TH solver (Eq. (12.61)), at each outer iteration, by microscopic cross-section polar interpolation and subsequent condensation to 26 energy groups, with P_1 scattering matrices. The boron concentration is initialized to $C_B^{(0)} = 0.01 \text{ ppm}$. The eigenvalue, fission source and inner iterations are converged up to 1 pcm, 100 pcm and 10 pcm, respectively, with the errors defined as in Eqq. (12.48), (12.50) and (12.64). The boron concentration is converged up to 1 ppm (Eq. (12.48)), while the tolerances on the fuel temperature and on the moderator density are set equal to 5°C and 1 pcm (Eq. (12.63)), respectively.

The fission source and the point-wise relative error on the angular flux moments rapidly converge to the required precision (Figs. 12.32-12.33). The k -eigenvalue converges in 33 outer iterations, showing large initial peaks, due to thermal-hydraulics and boron concentration update (Fig. 12.34). Table 12.11 reports the computed eigenvalue and associated critical boron concentration.

It is interesting to compare the axial distributions of the calculation marked as 'Strong' in Sect. 12.8.2 with the current ones. Due to the higher position of the control rods, we ex-

Eigenvalue	Critical boron concentration
0.9999962	514 ppm

Table 12.11: Computed eigenvalue and critical boron concentration.

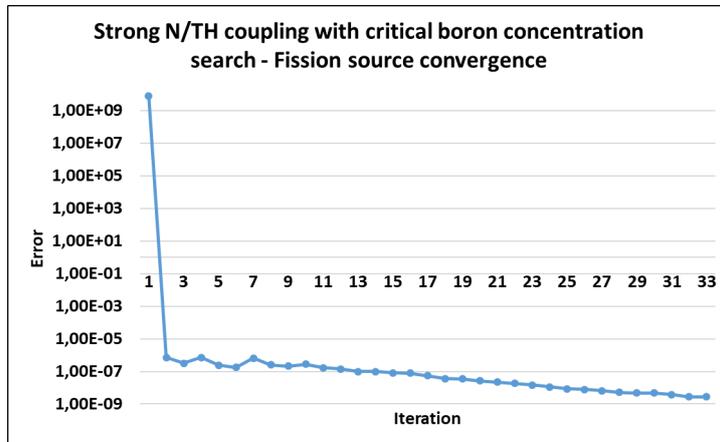


Figure 12.32: Fission source error vs number of outer iterations, for a 17x17 UOX- B_4C 3D lattice, with control rod tips at 370 cm, in tightly coupled N-TH iterations, with critical boron concentration search.

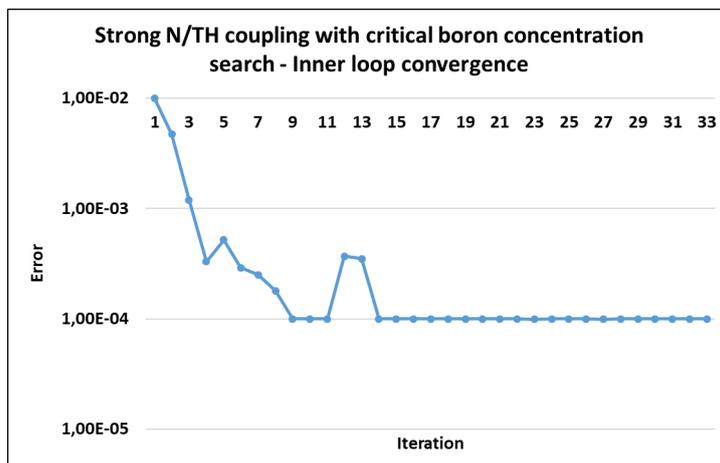


Figure 12.33: Inner loop error vs number of outer iterations, for a 17x17 UOX- B_4C 3D lattice, with control rod tips at 370 cm, in tightly coupled N-TH iterations, with critical boron concentration search.

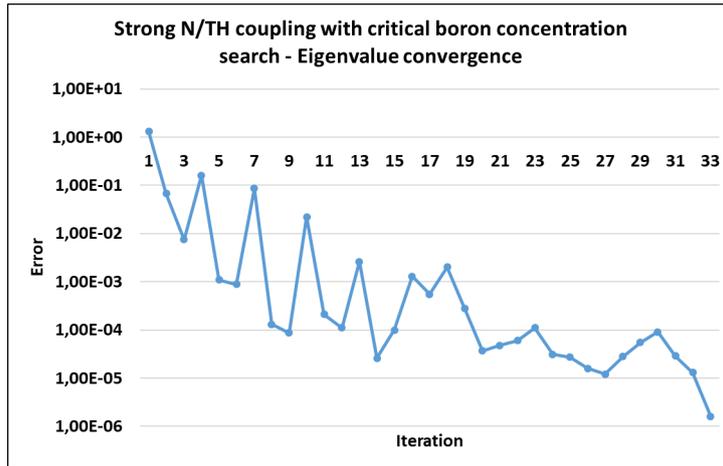


Figure 12.34: Eigenvalue error vs number of outer iterations, for a 17x17 UOX- B_4C 3D lattice, with control rod tips at 370 cm, in tightly coupled N-TH iterations, with critical boron concentration search.

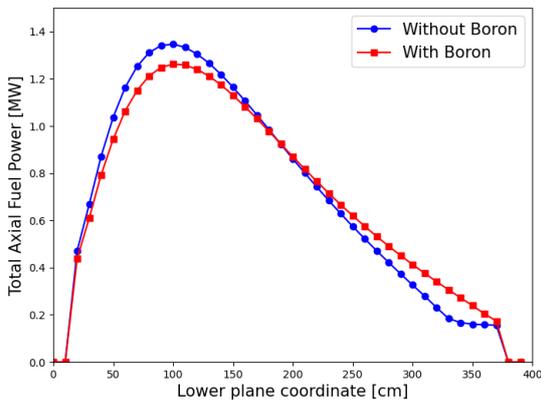


Figure 12.35: Total axial fuel power [MW]: comparison between 'Strong' and 'Weak'.

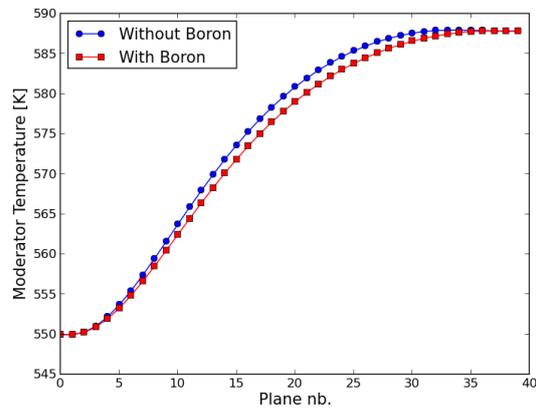


Figure 12.36: Moderator temperature [K]: comparison between 'Strong' and 'Weak'.

pect the axial power distribution to be less markedly skewed downwards. This condition is clearly verified, as shown in Fig. 12.35, displaying the total axial fuel power, namely $P_f(z) = \int dx \int dy P_f(x, y, z)$. Regarding the axial trend of the moderator temperature, we expect the inlet-outlet temperature difference to be the same. In fact, the power injected into the system is, in both cases, $P_{tot} = \dot{m}c_p\Delta T = 20 \text{ MW}$ and the imposed flow rate does not change from one calculation to another, $\dot{m} = 96 \frac{\text{kg}}{\text{s}}$. This physical consideration on the moderator temperature distribution appears to be satisfied, as shown in Fig. 12.36.

Concerning the simulation runtime, once again, the results are obtained by OpenMP parallel computing, with a maximum of 48 threads. The total CPU time amounts to 140 min. The fractions of time spent to recompute the CP coefficients and the macroscopic cross section libraries amount respectively to 14% and 4.2% of the total CPU time.

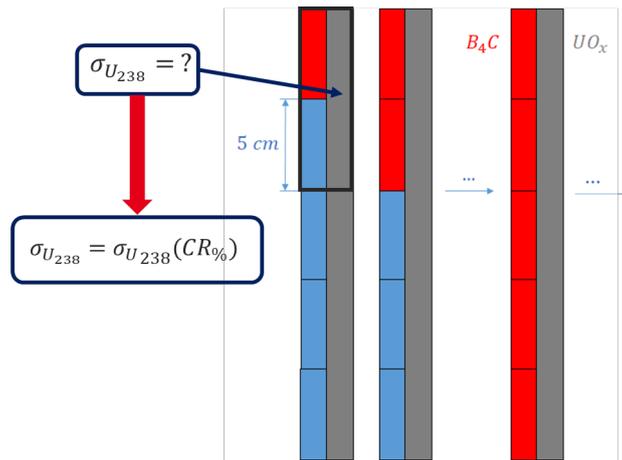


Figure 12.37: Resonant isotope microscopic cross-section interpolation, as a function of the percentage of the control rod inserted in the plane. In bold, an axial plane, housing a non-conforming mesh, which allows to reduce the spatial discretization of the fuel rods, but leaves the nodes that may host a control rod finely discretized.

12.8.4 . Coupled neutronics/thermal-hydraulics simulations, with control-rod movement

Let us consider the same lattice as the one in Sects. 12.8.2-12.8.3, i.e., a three-dimensional PWR assembly, with water blades, UOX fuel and B_4C control rods. The latter are gradually displaced along the z -axis, with a step-size of 5 cm . Control-rod tips occupy 10 successive axial positions, from 285 to 240 cm . Thermal feedback is included. In particular, \mathcal{N} is set equal to 1.

As shown in Chap. 8, control-rod movements may be simulated using a non-conforming mesh. Hence, let us consider an axial plane, of the kind illustrated in Fig. 12.37: nodes that host or may host a control rod (in red) are finely discretized, allowing for an exact representation of the rod-tip/moderator interface for each position of the control rod. Conversely, the neighbouring cells are less finely discretized, to reduce the number of computational regions (Sect. 8.3). In order to determine the set of cross sections for the latter, one may resort to a simple approach, which consists in obtaining resonant isotope microscopic cross sections by interpolation with respect to an operating parameter, namely $CR\%$, indicating the percentage of control rod inserted in the considered axial plane (Sects. 12.3.1-12.3.4).

The purpose of the presented case study is, at least, fourfold, i.e.,

- To test the proposed multi-physic scheme;
- To test microscopic cross-section interpolation, as a function of the physical parameters (T_f, T_w, ρ_w) ;
- To show that the control-rod reactivity worth curve is smooth;
- To test cross-section interpolation, as a function of the percentage of control-rod inserted in the plane, $CR\%$.

In the following, control-rod movements are simulated in two ways.

- First approach (Reference): the spatial domain is subdivided into 80 conformal planes, of 5 cm each. The control-rod parameter is equal either to 0 (unrodded plane) or 1 (rodded

plane). This approach serves as a "reference". Its solution is computed only for a subset of configurations;

- Second approach (Target): the spatial domain is subdivided into 40 non-conformal planes of 10 *cm* each. Nodes that host (or may host) a control rod contain two planes, of 5 *cm* each, so that the rod-tip/moderator interface is represented in an exact way. The control-rod parameter is equal to 0 in all planes below the control-rod tip, 1 in all planes above the control-rod tip, a fraction of 1 in partially-rodded planes.

The external reference library (GALILEE) contains 281 energy groups. The parametric library of self-shielding factors is obtained using APOLLO3®. Resonant microscopic cross sections are interpolated over an unstructured grid of state points, using radial basis kernel functions, and afterwards condensed to 26 groups, by XSTOOL. The generated library includes P_1 scattering matrices. No critical boron concentration search is performed.

Thermal-hydraulics is solved using a single channel and the following boundary conditions:

- Water inlet temperature $T_{in} = 550\text{ K}$;
- Pressure, $p = 155\text{ bar}$;
- Flow rate, $\dot{m} = 96\frac{\text{kg}}{\text{s}}$.

The total power is normalized to 20 *MW*. Thermal-hydraulics is converged according to the following criteria:

- Local error on the effective fuel temperature: $\epsilon_{T_f} = 5^\circ\text{C}$;
- Local error on the moderator mass density: $\epsilon_{\rho_m} = 1\text{ pcm}$.

Neutron transport is solved along 80 discrete directions, by S_8 Level-Symmetric angular quadrature. The spatial solution is obtained by Linear Short Characteristics. The required tolerances on the eigenvalue, fission integral and angular flux moments are set equal to 1 *pcm*, 100 *pcm* and 10 *pcm*. All calculations converge in less than 30 power iterations.

For the sake of conciseness, let us compare the two calculations (Reference and Target) for a single point of the reactivity worth curve, more specifically the one where the control-rod tips are located at 245 *cm*. First, let us compare the axial fuel power distributions (Fig. 12.38). The largest error is observed at plane 37 (i.e., $z \in [360, 370]\text{ cm}$) and amounts to less than 1.2% (Fig. 12.39). Thus, let us compare the detailed pin-by-pin fuel power distributions over this axial layer (Fig. 12.40). The maximum error is about 1.35%. The hot spot is located at position (6, 5, 10) in the macro-grid lodging the heterogeneous Cartesian nodes. The corresponding (pin-wise) relative error is about 0.08%. The relative error on the k -eigenvalue amounts to about 40 *pcm* for all compared configurations. The reactivity worth curve (of the calculation referred as 'Target') shows no cusping effect (Fig. 12.42). Good agreement is also observed in terms of fuel temperature, the maximum error being approximately 1 *K* (Fig. 12.41).

Finally, let us compare 'Target' and 'Reference' for the same configuration, i.e., the one where the control-rod tips are located at 245 *cm*, in terms of CPU time and memory occupation. Note that similar results are obtained for all configurations. The solution is obtained by DDM, with 90 and 180 subdomains, for 'Target' and 'Reference', respectively. In both cases,

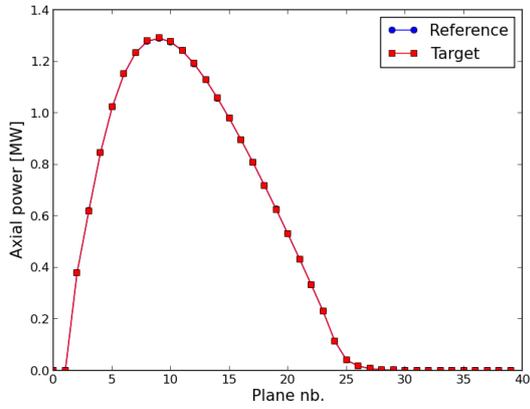


Figure 12.38: Reference vs Target: axial fuel-power distributions.

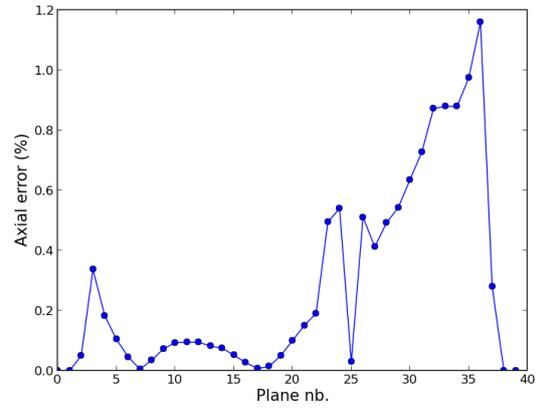


Figure 12.39: Relative difference on axial fuel-power distributions.

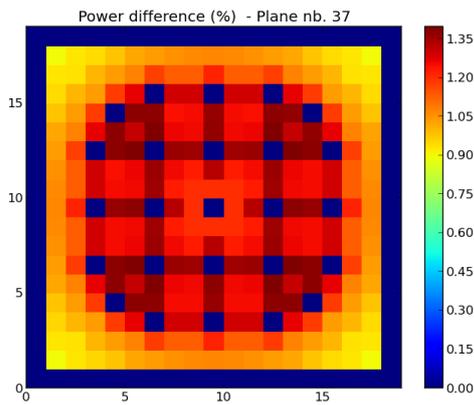


Figure 12.40: Relative difference on pin-by-pin fuel-power distributions over plane 37 out of 40.

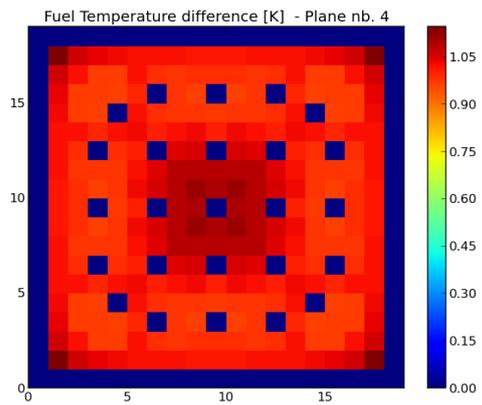


Figure 12.41: Absolute difference on pin-by-pin fuel temperature distributions, over plane 4 out of 40.

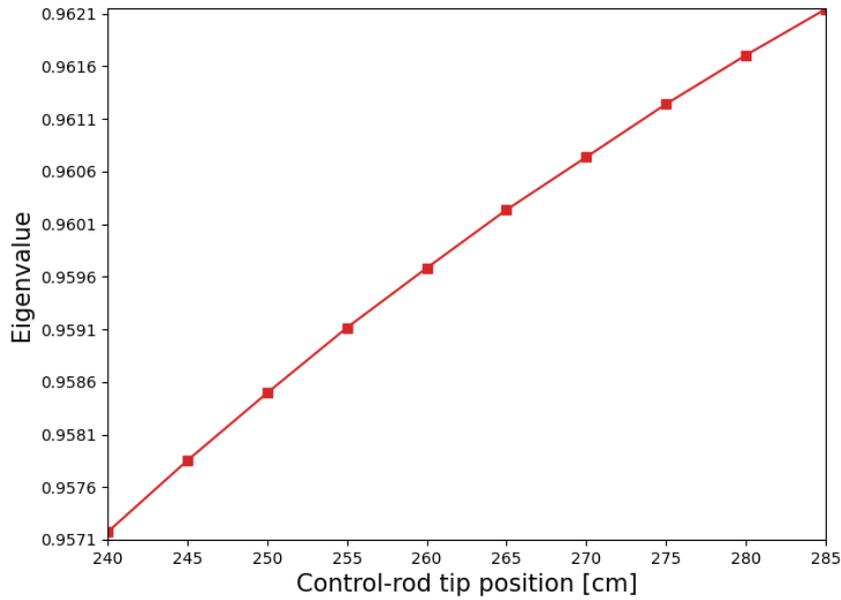


Figure 12.42: Reactivity-worth curve obtained by 'Target'.

subdomains may contain 144, 168 or 196 cells (blue lines in Fig. 12.43). Thanks to the asymptotic method, the number of unperturbed/computed cells amounts only to few units per subdomain, for all iterations, with a consequent memory (and computational) saving up to a factor 20 – 50, in both scenarios, i.e., 'Reference' and 'Target' (Fig. 12.43). Note that the tolerance ε_{Σ} is set equal to 10% (Eq. (9.14)). The total number of media is 146960 for 'Reference' and 73516 for 'Target'. The total memory occupation amounts to 12 and 9 GB, respectively.

The total number of regions used in 'Reference' is two times larger than the one used in 'Target'. The same applies to the number of cells. As the average number of region per cell is about the same (~ 5), the number of collision coefficients computed (and stored) in 'Reference' is two times larger than the one in 'Target', due to the relation shown in Tab. 11.1.

Power iterations are initialized in about 19 ('Reference') and 10 min ('Target'). The multi-physic solution is converged in 12 ('Reference') and 27 ('Target') power iterations. The overall computational time amounts to 1 h 30 min and 1 h 47 min, in OpenMP parallelism (the workstation is an Intel(R) Xeon(R) Silver 4214 CPU @ 2.20 GHz). The ultimate reason of the different time to solution is due to the CMFD method, as the performance of CMFD tends to deteriorate for nodes of the order of 10 cm. As thermal-hydraulics is converged in 11 and 26 iterations, the time spent for the coefficients and for cross-section interpolation (along the outer loop) is rather different, as reported in Tab. 12.12.

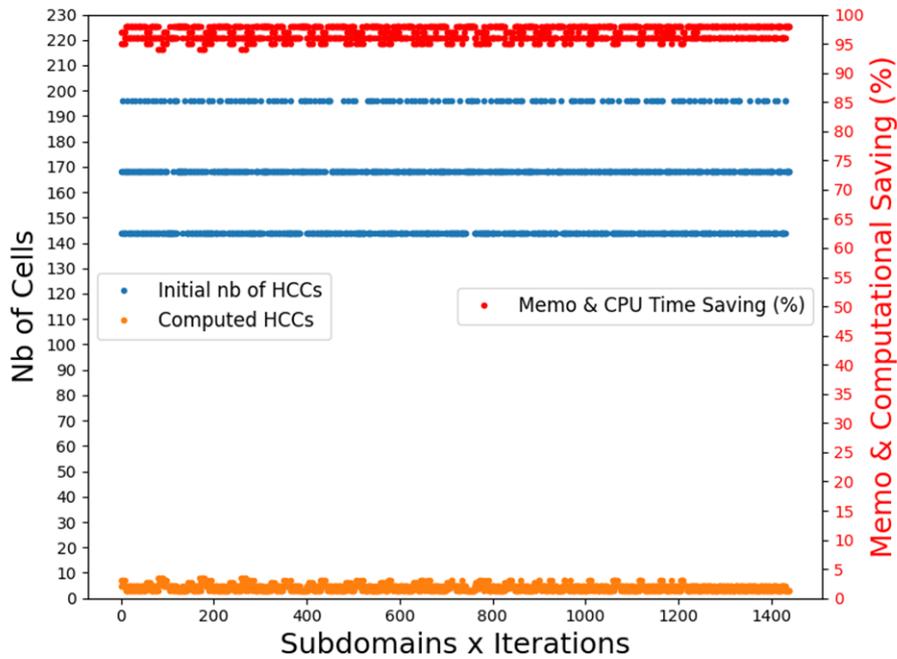


Figure 12.43: 'Reference' vs 'Target': observed memory saving for a single configuration, more specifically the one where the control rod tips are located at 245 cm. Note that the figure displays the number of input and computed cells (blue lines and orange line, respectively), for each subdomain, over all power iterations.

Estimated CPU time	Reference	Target
For microscopic cross section interpolation	2 min (2%)	3 min (2%)
For coefficients	22 min (21%)	59 min (53%)

Table 12.12: 'Reference' vs 'Target': time for microscopic cross section interpolation and coefficients and percentage with respect to the total CPU time.

13 - Conclusions and perspectives

13.1 . Objectives and requirements

This thesis work lays the foundations for a new simulation pathway for coupled neutronics/ thermal-hydraulics simulations, without resorting to cross-section homogenization, in real three-dimensional geometry. The proposed scheme involves three codes:

- A discrete-ordinate multigroup transport solver, IDT, projecting the Boltzmann equation onto heterogeneous modular cells, by means of the method of short characteristics (MOSC);
- A software performing microscopic cross-section interpolation and energy condensation (if required), namely XSTOOL, generating a macroscopic cross-section library, for neutron transport;
- A multi-1D, two-phase, thermal-hydraulic code of APOLLO3®, THEDI, here used to solve mass, momentum and energy conservation, in stationary monophasic flow, over independent sub-channels, as well as a radial diffusion equation for fuel heat conduction.

The present manuscript describes an advanced two-step scheme, based on direct microscopic cross-section interpolation and possible subsequent macroscopic cross-section condensation, by XSTOOL. Multiphysics is solved by tight-coupled multigroup-transport/thermal-hydraulics iterations. Priority is given to the neutronic solution, with the following requirements:

- Perform high-fidelity transport, capable of capturing large neutron flux gradients, in strong absorbers and markedly diffusive media;
- Design non-extruded/non-conformal/unstructured geometries to address increasingly diverse reactor geometries;
- Provide an accurate 3D transport model, in real heterogeneous geometry, without control-rod homogenization, in order to eliminate the control-rod cusping effect;
- Mitigate the memory needs of collision probability-based methods, such as MOSC, to deal with systems hosting thousands of different regions, in burn-up and/or multiphysics calculations.

To pursue these objectives, a new version of IDT has been developed, starting from its homonymous progenitor, hereinafter referred to as APOLLO3® IDT. Let us recall some key features and novelties of this new version of the code.

13.2 . Towards a new version of IDT

13.2.1 . New geometrical capabilities

These include

- The construction of upgraded modular Heterogeneous Cartesian Cells (HCCs), equipped with both coaxial cylinders and XYZ planes;

- The implementation of a novel combinatorial ray-tracing algorithm, intersecting each trajectory t with a set of geometrical patterns. For the time being, these comprise Cartesian bands and concentric rings, with the chord-lengths being computed in two steps, i.e., 2D-tracking over the XY -plane and 2D-tracking over the transverse tZ -slice;
- The option to pivot the cylinders along the X and Y -axis, as well as to design non-Cartesian geometries, such as hexagonal patterns, taking advantage of the possibility to move the rings centre;
- The introduction of the Tuple-to-Region Map (TRM), which allows to condense regions with different band or ring number, and thus to coarsen or even customize the spatial discretization of each HCC. This intuitive technique allows to manufacture unstructured, non-conformal and non-extruded geometries, provided that the HCC borders and surface discretizations are still conformal, for angular currents exchange.

This ensemble of tools permits, on the one hand, to enhance the numerical accuracy of IDT. In this respect, a preliminary validation campaign has been carried out, by means of quantitative comparisons with reference codes, mainly TRIPOLI-4® continuous-energy Monte Carlo and APOLLO3® TDT-MOC, but also MCNP, within the framework of the C5G7 benchmark. On the other hand, the implemented geometry permits to avoid the appearance of cusps in the reactivity worth curve, in control-rod insertion/withdrawal steady-state calculations. Differently from traditional nodal solvers, IDT heterogeneous non-extruded non-conforming geometry allows the control rod tip and the underlying moderator to be represented separately in the node, without significant increase in computational resources. In fact, only few cells, i.e., those that permit control-rod displacements, are finely discretized along Z , with one plane for each axial position of the rod tip.

13.2.2 . Novelties on the method of short characteristics

The main innovation on MOSC concerns the introduction of an asymptotic development to reduce the memory pressure of the probability matrices. This method consists in grouping HCCs with equal geometry and similar optical properties into a handful of reference cells. The probability coefficients are solely determined for the latter, with the other HCCs being interpreted as a 'perturbation' of a reference cell. Under relatively loose conditions, involving the deviation of the total cross sections of the 'perturbed' cells with respect to the corresponding reference HCC, the transport solution can be reconstructed by means of a Neumann series development. The memory burden is thus partially transferred onto the inner iterations, which perform more transport sweeps to reconstruct the angular flux. Based on early numerical evidence, this technique greatly reduces the memory requirements of the probability matrices, making collision probability (CP) and MOSC solvers, such as IDT, more attractive for three-dimensional lattice and core calculations.

A dedicated study has highlighted the possibility of further reducing the memory imprint and CPU time of the angular-dependent CP matrices, by applying simple reciprocity and symmetry relations. While the former are always verified, the latter are valid only for symmetric cells, with piece-wise linear source and interface flux distributions. The application of these transformations allows to lower the memory burden by up to a factor of 16.

Finally, incomplete parabolic short characteristics (PSC) have been introduced in IDT, in order to retain high numerical accuracy while reducing the number of computational regions.

Preliminary results on 3×3 two-dimensional patterns and on the Watts Bar benchmark show good agreement with the Monte Carlo simulation. Unfortunately, the aforementioned symmetry relations, based on permutation and sign matrix application, are not satisfied by second-order coefficients. Hence, a mixed approach, using radial piece-wise linear source and interface flux distributions and parabolic piece-wise developments along the Z -axis, has been tested, to lower the memory footprint by up to a factor of 8.

13.3 . Multiphysics

A new advanced two-level calculation scheme has been presented. Unlike the classical two-step paradigm, non-homogenized cross sections are provided to the core calculation. In fact, it has been shown that the effective parametric microscopic cross sections exhibit much smoother variations as a function of the operational/physical parameters and, as such, can be easily interpolated, also offering the possibility of reducing the number of interpolation nodes. The use of the neutron flux, stored in the EPL-SSF libraries, from the lattice calculation step, also allows for multigroup cross-section condensation. Based on preliminary numerical tests, this does not deteriorate the accuracy of the reaction rates. It is expected that an intermediate step, based on equivalence theory, will not be necessary for reactor core calculations.

The scheme outlined in this work implements a tight neutronic/thermal-hydraulic coupling, the thermal feedback being taken into account directly in the power iterations. Nevertheless, a simple input parameter, which can be adjusted by the user, can 'weaken' the N/TH coupling. Initial results suggest that tight coupling, with thermal-hydraulics feedbacking on neutronics at each outer iteration, can significantly reduce the number of outer iterations. The CPU time necessary can be largely improved, using mixed MPI/OpenMP parallelism, on several MPI processes.

13.4 . Future work

Numerous improvements, research topics and numerical tests are still possible for IDT. In the following, a few are suggested.

13.4.1 . Geometric model

The geometric model can be greatly enriched, with the introduction of further patterns, such as spheres and azimuthal sectors. These could be easily integrated into the modular geometry of IDT, with little implementation effort. There still remains a long series of numerical tests to be carried out, including the validation of hexagonal geometries against Monte Carlo and the use of cells with mixed vertical/horizontal cylinders, with a projection toward nuclear systems such as VVER and CANDU reactors.

13.4.2 . PSC and hybrid-order short characteristics

In order to become competitive with the linear order approximation, a new set of symmetry relations shall be constructed for incomplete parabolic short characteristics. Meanwhile, multiple hybrid-order short-characteristics schemes may be tested, e.g., one implementing linear piece-wise sources and constant piece-wise interface angular fluxes, with the possibility to retain the most successful in the release version of IDT, for future calculations.

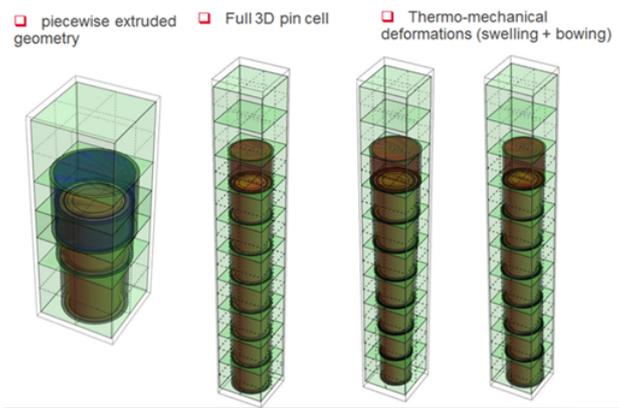


Figure 13.1: Neutron-transport/fuel-performance coupling: a possible line of research for future applications of IDT.

13.4.3 . Multiphysics and industrial-scale calculations

Finally, enormous work must be conducted for applications of IDT to multiphysics, both to confirm the first numerical evidence shown in Chap. 12, with further tests on different lattices, and to extensively verify and validate the calculation scheme. Numerical comparison with other codes will be crucial in this regard, to establish the reliability and accuracy of the models and algorithms described in the final part of this thesis manuscript.

13.4.4 . Control-rod withdrawal and insertion

With little effort, steady-state neutron transport calculations, including thermal-hydraulic feedback, can be performed to determine the control-rods position and boron concentration that make the reactor critical.

13.4.5 . IDT coupled to more detailed thermal-hydraulic codes

IDT may be coupled to more complex thermal-hydraulic solvers, such as

- CATHARE, [108], which uses a biphasic-flow six-equation model, for the average mass, momentum and energy of each phase,
- FLICA4 or FLICA5, [117], which use a biphasic-flow four-equation model, including vapor mass conservation and three conservation equations for the mass, the internal energy and the momentum of the liquid-vapor mixture.

13.4.6 . IDT coupled to thermal-mechanical codes

IDT may be coupled to a thermal-mechanical solver, such as ALCYONE, [118], which is a multi-dimensional fuel performance code for PWRs, to simulate fuel-pellets and cladding deformation, due to neutron irradiation. Also in this case, tight neutron-transport/fuel-performance coupling may be preferred. Figure 13.1 provides an application example.

APPENDICES

A - THEDI - A BRIEF OVERVIEW

A.1 . INTRODUCTION

Neutronics and thermal-hydraulics are coupled problems, as

- the macroscopic cross sections depend on temperature and density,
- the power released by nuclear reactions affects medium's temperature.

THEDI (**THE**rmohydraulique **DI**phasique) is a multi-1D two-phase thermal-hydraulic code of APOLLO3®, [77], [78]. It is equipped with 4 main solver options:

- (a) A 1D thermal-hydraulic model, consisting of 4 balance equations, for each core channel;
- (b) A 1D thermal-hydraulic model, consisting of 3 balance equation, each "ex-core" component;
- (c) A 1D thermal model, consisting of 1 balance equation, for nuclear-fuel heat conduction;
- (d) A point kinetics model, for monoenergetic-neutron and precursor concentrations.

In the following, some details are provided on model (a) and (c). In Sect. A.2, model (a) is introduced, pointing out the mass, momentum and energy conservation equations, for the vapor and liquid-vapor mixture. It is worth mentioning that THEDI is able to model two types of coolant, i.e., water and sodium. Channels are independent one from another, as they are not coupled by transverse terms. The balance equations are discretized into axial nodes and finite differences are applied on a semi-implicit model, to avoid instabilities caused by coarse time-steps.

In Sect. A.3, a brief overview of model (c) is proposed. By taking advantage of fuel-pellet symmetrical shape, the diffusion equation is discretized into radial layers. Finite-differences are applied on an implicit scheme, to determine the fuel temperature distribution.

A.2 . 1D AXIAL THERMAL-HYDRAULICS

A.2.1 . Notation

- Time, t
- Axial coordinate, z
- Void fraction, α
- Vapor subscript, v
- Liquid subscript, l
- Mass density, ρ
- Speed, w

- Mass rate exchange between liquid and vapor, Γ
- Shear coefficient, C
- Gravitational acceleration, g
- Internal energy, u
- Density of the liquid-vapor mixture, $\rho_m = \alpha\rho_v + (1 - \alpha)\rho_l$
- Speed of the liquid-vapor mixture, $w_m = \frac{1}{\rho_m}(\alpha\rho_v w_v + (1 - \alpha)\rho_l w_l)$
- Relative speed between phases, w_r
- Internal energy of the liquid-vapor mixture, $u_m = \frac{1}{\rho_m}(\alpha\rho_v u_v + (1 - \alpha)\rho_l u_l)$
- Wall temperature, T_p
- Liquid temperature, T_l
- Pressure, p
- Heat transfer coefficient, h
- Heat flux density, q

A.2.2 . Balance equations

THEDI solves the following set of 1D balance equations, for each sub-channel, [77]:

- Mass continuity equations for the liquid-vapor mixture and for the vapor, respectively,

$$\frac{\partial \rho_m}{\partial t} + \frac{\partial \rho_m w_m}{\partial z} = 0, \quad (\text{A.1})$$

$$\frac{\partial \alpha \rho_v}{\partial t} + \frac{\partial \alpha \rho_v w_v}{\partial z} = \Gamma, \quad (\text{A.2})$$

- Momentum conservation for the liquid-vapor mixture,

$$\frac{\partial \rho_m w_m}{\partial t} + \frac{\partial (\alpha \rho_v w_v^2 + (1 - \alpha) \rho_l w_l^2)}{\partial z} + \frac{\partial p}{\partial z} + \rho_m g - C w_m |w_m| = 0, \quad (\text{A.3})$$

- Energy conservation for the liquid-vapor mixture,

$$\frac{\partial \rho_m u_m}{\partial t} + \frac{\partial (\alpha \rho_v u_v w_v + (1 - \alpha) \rho_l u_l w_l)}{\partial z} + p \frac{\partial (\alpha w_v + (1 - \alpha) w_l)}{\partial z} = q + h(T_p - T_l). \quad (\text{A.4})$$

Note that

- The gravitational and kinetic energy are negligible with respect to the internal energy, thus they are disregarded in Eq. A.4;
- As mentioned in Sect. A.1, THEDI sub-channels do not exchange heat or mass.

A.2.3 . Numerical solution

The numerical solution is obtained by applying a semi-implicit Stability Enhancing Two-Steps approach (SETS), [79]. Eq. A.1-A.4 are discretized by finite differences. This scheme tends to foster numerical stability (particularly in the case of monophasic flow), while the numerical precision is only linear with respect both to t and z . As Eq. A.1-A.4 form a system of 4 equations, one can choose 4 unknowns, which, in the case of THEDI, are the temperature, $T(t, z)$, pressure, $p(t, z)$, void fraction, $\alpha(t, z)$, and speed of the liquid-vapor mixture, $w_m(t, z)$. Concerning the other quantities appearing in the 1D balance model,

- w_r, Γ, C and h are provided by correlations,
- g is a constant,
- q and T_w are given,
- ρ_l, ρ_v, u_l and u_v depend on p and T .

The boundary conditions are

- p_{in} and p_{out} (or, alternatively, p_{in} and \dot{m}_{out} or, alternatively, \dot{m}_{in} and p_{out}),
- T_{in} and α_{in} ,

where in/out denote the inlet/outlet sides, while \dot{m} represents the mass flow rate. Eq. A.1-A.4 are linearized and solved iteratively, until convergence.

A specific solution pathway is available for the particular case of stationary monophasic flow. This approach is particularly suitable for neutronics applications and requires little computational runtime.

A.3 . 1D RADIAL DIFFUSION

Heat conduction in nuclear fuel is described by the diffusion equation:

$$\rho c_p \frac{\partial T}{\partial t} = \nabla(\lambda T) + q''' , \quad (\text{A.5})$$

where c_p and λ denote the specific heat capacity and the thermal conductivity, while q''' is the volumetric heat power generation. Note that Eq. (A.5) is not linear, as ρc_p and λ generally depend on temperature, T . Assuming the fuel rod has infinite length, the Laplacian reduces to

$$\nabla f = \frac{1}{r} \frac{\partial}{\partial r} \left(r \frac{\partial f}{\partial r} \right) , \quad (\text{A.6})$$

where f is twice-differentiable. Eq. (A.5) becomes:

$$\rho c_p \frac{\partial T}{\partial t} = \frac{1}{r} \frac{\partial}{\partial r} \left(r \frac{\partial T}{\partial r} \right) + q''' , \quad (\text{A.7})$$

Eq. (A.7) is solved applying an implicit finite difference scheme. The discretized equation is linearized and solved iteratively.

B - ANALYTICAL COMPUTATION OF THE FLUX MOMENTS BY TAYLOR EXPANSION

The asymptotic expansion presented in Chap. 9 may be addressed by analytical treatment of moments ψ_n (Eq. (9.10)). This analysis is based on the study of the derivatives of the angular flux with respect to the total cross section. In the case of a homogeneous cell, the integral form of the transport equation reads as

$$\psi(x; \Sigma) = \psi(0)e^{-\Sigma x} + \int_0^x q(y)e^{-\Sigma(x-y)} dy, \quad (\text{B.o.1})$$

where the coordinate transformation $\mathbf{r} = \mathbf{r}_- + x \boldsymbol{\Omega}$ is introduced along the characteristic line, and cross sections are treated as parameters of the flux distribution, i.e., $\psi = \psi(x; \Sigma)$. The partial derivative with respect to Σ can be drawn from Eq. (B.o.1),

$$\begin{aligned} \frac{\partial \psi}{\partial \Sigma}(x; \Sigma) &= -x\psi(0)e^{-\Sigma x} + \int_0^x -(x-y)q(y)e^{-\Sigma(x-y)} dy \\ &= -x\psi(x) + \int_0^x yq(y)e^{-\Sigma(x-y)} dy. \end{aligned} \quad (\text{B.o.2})$$

As shown in Eq. (B.o.2), the first derivative generates a linear spatial moment along the trajectory. Thus, the first-order moment of the Neumann series (Eq. (9.10)) is equal to the first-order Taylor expansion around the unperturbed cross section, namely

$$\psi \simeq \psi_1 + O(\delta\Sigma^2) = \psi(x; \Sigma_0) + \frac{\partial \psi}{\partial \Sigma}(x; \Sigma_0) \delta\Sigma + O(\delta\Sigma^2), \quad (\text{B.o.3})$$

with $\delta\Sigma = (\Sigma - \Sigma_0)$. By replacing Eq. (B.o.2) in Eq. (B.o.3), one obtains

$$\psi_1(x) = (1 - x \delta\Sigma)\psi_0(x) + \delta\Sigma \int_0^x yq(y)e^{-\Sigma_0(x-y)} dy. \quad (\text{B.o.4})$$

The Taylor series can be extended up to order $n \geq 0$, with n -th derivative

$$\begin{aligned} \frac{\partial^n \psi}{\partial \Sigma^n}(x; \Sigma) &= (-x)^n \psi(0; \Sigma) e^{-\Sigma x} + \sum_{i=0}^n \binom{n}{i} (-x)^i \int_0^x y^{n-i} q(y) e^{-\Sigma(x-y)} dy \\ &= (-x)^n \psi(x; \Sigma) + \sum_{i=0}^{n-1} \binom{n}{i} (-x)^i \int_0^x y^{n-i} q(y) e^{-\Sigma(x-y)} dy. \end{aligned} \quad (\text{B.o.5})$$

Thus, the n -th moment can be computed analytically as

$$\psi_n(x; \Sigma) = \sum_{i=0}^n \frac{\partial^i \psi}{\partial \Sigma^i}(x; \Sigma_0) \frac{\delta\Sigma^i}{i!} \quad (\text{B.o.6})$$

$$= \left[\sum_{i=0}^n \frac{(-x \delta\Sigma)^i}{i!} \right] \psi(x; \Sigma_0) + \quad (\text{B.o.7})$$

$$\sum_{i=1}^n \left[\sum_{j=0}^{i-1} \binom{i}{j} (-x)^j \int_0^x y^{i-j} q(y) e^{-\Sigma_0(x-y)} dy \right] \frac{\delta\Sigma^i}{i!}. \quad (\text{B.o.8})$$

Since

$$\lim_{n \rightarrow \infty} \left[\sum_{i=0}^n \frac{(-\delta \Sigma x)^i}{i!} \right] = e^{-\delta \Sigma x} \quad (\text{B.0.9})$$

as $n \rightarrow \infty$, we obtain

$$\lim_{n \rightarrow \infty} \psi_n(x; \Sigma) = \psi(0) e^{-(\Sigma_0 - \delta \Sigma)x} + \int_0^x q(y) e^{-(\Sigma_0 - \delta \Sigma)(x-y)} dy, \quad (\text{B.0.10})$$

$$= \psi(x; \Sigma). \quad (\text{B.0.11})$$

Eq. (B.0.4) can be solved numerically and used to compute Taylor expansion of numerical coefficients around the unperturbed state, Σ_0 .

As an example of application to HCC framework, let us consider the linear Taylor expansion of the collision matrix, $\mathbf{C}_{d,\alpha,\beta}(\Sigma_{\mathbf{c}})$. Since every coefficient is a multivariable function of $\Sigma_{\mathbf{c}}$, the linear Taylor expansion around $\Sigma_{\mathbf{c}_0}$ reads as

$$\mathbf{C}_{d,\alpha,\beta}(\Sigma_{\mathbf{c}}) = \mathbf{C}_{d,\alpha,\beta}(\Sigma_{\mathbf{c}_0}) + \sum_{\gamma \in \mathcal{G}_{\mathbf{c}}} \left. \frac{\partial \mathbf{C}_{d,\alpha,\beta}}{\partial \Sigma_{\mathbf{c},\gamma}} \right|_{\Sigma_{\mathbf{c}} = \Sigma_{\mathbf{c}_0}} (\Sigma_{\mathbf{c},\gamma} - \Sigma_{\mathbf{c}_0,\gamma}). \quad (\text{B.0.12})$$

Eq. (B.0.12) requires the computation and storage of the unperturbed matrix, namely $\mathbf{C}_{d,\alpha,\beta}(\Sigma_{\mathbf{c}_0})$, but also of a number of matrices containing the partial derivatives, i.e., $\frac{\partial \mathbf{C}_{d,\alpha,\beta}}{\partial \Sigma_{\mathbf{c},\gamma}}$, whose number is equal to the number of regions of the HCC. The coefficients of matrix $\frac{\partial \mathbf{C}_{d,\alpha,\beta}}{\partial \Sigma_{\mathbf{c},\gamma}}$ are combination of linear moments of the flux, as it can be inferred by looking at Eq. (B.0.2). For example, it can be proven that the derivatives of the constant moment (first row of $\frac{\partial \mathbf{C}_{d,\alpha,\beta}}{\partial \Sigma_{\mathbf{c},\gamma}}$) can be obtained by combining linear moments of the flux, which are already contained in the matrix. In the past years, attempts have been made to manage efficient computations of the CP matrices. However, as cells are heterogeneous and HCC regions are correlated, there is a huge amplification of the computational effort and the memory burden of the derivatives. Hence, this approach has been dismissed, as the memory burden is not effectively reduced when applied to HCCs. Nevertheless, its application to homogeneous cells could be of some interest.

C - ASYMPTOTIC METHOD - SPECTRAL RADIUS ANALYSIS

The spectral radius of the operator $\mathbf{L}_0^{-1}\delta\mathbf{L}$ can be measured as

$$\rho(\boldsymbol{\Omega}) = \|\mathbf{L}_0^{-1}\delta\mathbf{L}\| \sim \max_n \left(\frac{\int_D \delta\Sigma \psi_n}{\int_D \boldsymbol{\Omega} \cdot \nabla \psi_n + \int_D \Sigma_0 \psi_n} \right).$$

At convergence,

$$\begin{aligned} \rho(\boldsymbol{\Omega}) &\sim \frac{\delta\tau(\boldsymbol{\Omega})}{|(J_+ - J_{in})(\boldsymbol{\Omega}) + \tau(\boldsymbol{\Omega})|} \\ &= \left| \frac{\int_D d^3r \delta\Sigma(\mathbf{r})\psi(\mathbf{r}, \boldsymbol{\Omega})}{\int_{\partial D_+} d^3r_+ |\boldsymbol{\Omega} \cdot \mathbf{n}| \psi(\mathbf{r}_+, \boldsymbol{\Omega}) - \int_{\partial D_-} d^3r_- |\boldsymbol{\Omega} \cdot \mathbf{n}| \psi_{in}(\mathbf{r}_-, \boldsymbol{\Omega}) + \int_D d^3r \Sigma(\mathbf{r})\psi(\mathbf{r}, \boldsymbol{\Omega})} \right|. \end{aligned}$$

We can distinguish three cases:

- Case with zero net leakage, $(J_+ - J_{in})(\boldsymbol{\Omega}) = 0$:

$$\rho(\boldsymbol{\Omega}) = \left| \frac{\delta\tau(\boldsymbol{\Omega})}{\tau(\boldsymbol{\Omega})} \right|.$$

Thus, it is sufficient to satisfy the condition

$$|\delta\tau(\boldsymbol{\Omega})| < \tau(\boldsymbol{\Omega}),$$

for the method to converge.

- Case with zero incoming flux, $\psi_{in}(\mathbf{r}_-, \boldsymbol{\Omega}) = 0$:

$$\rho(\boldsymbol{\Omega}) = \left| \frac{\delta\tau(\boldsymbol{\Omega})}{J_+(\boldsymbol{\Omega}) + \tau(\boldsymbol{\Omega})} \right| < \frac{|\delta\tau(\boldsymbol{\Omega})|}{\tau(\boldsymbol{\Omega})}$$

since $J_+(\boldsymbol{\Omega}) \geq 0$ by definition. Thus, it is sufficient to satisfy the condition

$$|\delta\tau(\boldsymbol{\Omega})| < \tau(\boldsymbol{\Omega}).$$

- Case with thick medium, i.e. with $J_+(\boldsymbol{\Omega}) \ll J_-(\boldsymbol{\Omega})$. Provided $J_+(\boldsymbol{\Omega})$ is negligible,

$$\rho(\boldsymbol{\Omega}) = \frac{|\delta\tau(\boldsymbol{\Omega})|}{|-J_{in}(\boldsymbol{\Omega}) + \tau(\boldsymbol{\Omega})|}.$$

Hence, the condition to satisfy is given by

$$\begin{cases} |\delta\tau(\boldsymbol{\Omega})| < \tau(\boldsymbol{\Omega}) - J_{in}(\boldsymbol{\Omega}), & \text{if } J_{in}(\boldsymbol{\Omega}) < \tau(\boldsymbol{\Omega}), \\ |\delta\tau(\boldsymbol{\Omega})| < J_{in}(\boldsymbol{\Omega}) - \tau(\boldsymbol{\Omega}), & \text{if } J_{in}(\boldsymbol{\Omega}) > \tau(\boldsymbol{\Omega}), \end{cases}$$

and is undetermined, if $\tau(\boldsymbol{\Omega}) = J_{in}(\boldsymbol{\Omega})$.

Bibliography

- [1] International Energy Agency, "Net Zero by 2050. A Roadmap for the Global Energy Sector", <https://www.iea.org/reports/net-zero-by-2050>, 2021.
- [2] Christophe Demazière, "Modelling of Nuclear Reactor Multi-physics - From Local Balance Equations to Macroscopic Models in Neutronics and Thermal-Hydraulics", *Elvisier Inc.*, 2019.
- [3] M. D. DeHart, "Advancements in Generalized-Geometry Discrete Ordinates Transport for Lattice Physics Calculations", *PHYSOR-2006: American Nuclear Society's Topical Meeting on Reactor Physics*, 2006.
- [4] G. A. Gunow, "Full core 3D neutron transport simulations using the method of characteristics with linear sources", PhD thesis, Massachusetts Institute of Technology, 2018.
- [5] A. Hébert, "Applied Reactor Physics", *Presses internationales Polytechnique*, Montréal, 2009.
- [6] K. Devan, A. Bachchan, "Homogeneous reactor and neutron diffusion equation", in "Physics of Nuclear Reactors", *Academic Press*, pp. 193-262, 2021.
- [7] L. Cai, Y. Pénéliou, J. Tommasi, J.-F. Vidal, C. M. Diop, "Leakage-corrected fast reactor assembly calculation with Monte-Carlo code TRIPOLI4® and its validation methodology", *Physor 2014 - The Role of Reactor Physics toward a Sustainable Future*, Kyoto, Japan, 2014.
- [8] International Atomic Energy Agency, <https://pris.iaea.org/PRIS>
- [9] World Nuclear Association, <https://world-nuclear.org>
- [10] Commissariat à l'énergie atomique et aux énergies alternatives, "Neutronics", *Editions Le Moniteur*, 2015.
- [11] Y. Wu, "Neutron Kinetics" in "Neutronics of Advanced Nuclear Systems", *Springer*, 2019.
- [12] A. Jinaphanh, A. Zoia, "Perturbation and Sensitivity of the Time Eigenvalue by a Generalized Iterated Fission Probability Method", *International Conference on Mathematics and Computational Methods Applied to Nuclear Science and Engineering, M&C 2019*, Portland, United States, 2019. .
- [13] B. R. Betzler and J. J. Powers, A. Worrall, "Molten salt reactor neutronics and fuel cycle modeling and simulation with SCALE", *Annals of Nuclear Energy*, 2017.
- [14] P. Reuss, "Précis de neutronique", *EDP Sciences*, 2003.
- [15] W. Ford, "The Advancement of Stable, Efficient and Parallel Acceleration Methods for the Neutron Transport Equation", PhD Thesis, Université Paris Saclay, 2019.
- [16] A. Gastelum-Strozzi, "Smoothing particle hydrodynamics parallel implementation for numerical modelling of solid-fluid interactions", PhD Thesis, p. 27, 2011.
- [17] J. Verbeke, C. Hagmann, D. Wright, "Simulation of Neutron and Gamma Ray Emission from Fission and Photofission", Lawrence Livermore National Laboratory, pp. 7-8, 2010.

- [18] R. Sanchez, L. Bourhrara, "Existence Result for the Kinetic Neutron Transport Problem with a General Albedo Boundary Condition", *Transport Theory and Statistical Physics*, Volume 40, Issue 2, pp. 69-84, 2011.
- [19] <https://www.nuclear-power.com/nuclear-power/reactor-physics/neutron-diffusion-theory/albedo-boundary-condition>
- [20] F. B. Brown, T. M. Sutton, "Monte Carlo Fundamentals", *Lockheed Martin Company*, Knolls Atomic Power Laboratory, New York, 1996.
- [21] G. I. Bell, S. Glasstone, "Nuclear Reactor Theory", *Van Nostrand Reinhold Company*, 1970.
- [22] D. Rozon, "Introduction to Nuclear Reactor Kinetics", *Presses internationales Polytechnique*, Montréal, 1998.
- [23] Z. M. Prince, "Improved quasi-static methods for time-dependent neutron diffusion and implementation in Rattlesnake", MSc Thesis, Texas A&M University, 2017.
- [24] G. Arfken, "Legendre Functions", in "Mathematical Methods for Physicists", 3rd edition, *Academic Press*, Orlando, pp. 637-711, 1985.
- [25] G. Arfken, "Spherical Harmonics" and "Integrals of the Products of Three Spherical Harmonics", in "Mathematical Methods for Physicists", 3rd edition, *Academic Press*, Orlando, pp. 680-685 and 698-700, 1985.
- [26] R. Sanchez, "Cross section self-shielding I", Seminar, 2023.
- [27] E. Rosier, "Study, development and evaluation of the subgroup methods based on the physical probability tables in APOLLO3® for thermal reactor calculations", PhD thesis, Université Paris-Saclay, 2022.
- [28] L. Bourhrara, "A new numerical method for solving the Boltzmann transport equation using the P_N method and the discontinuous finite elements on unstructured and curved meshes", *Journal of Computational Physics*, 2019.
- [29] E. E. Lewis, W. F. Miller, "Computational Methods of Neutron Transport", *Wiley-InterScience*, United States, 1984.
- [30] A. M. Baudron and J. J. Lautard, "Simplified P_N transport core calculations in the Apollo3 system", *International Conference on Mathematics and Computational Methods Applied to Nuclear Science and Engineering, M&C 2011*, Rio de Janeiro, 2011.
- [31] A. Hébert, "The collision probability method in 1D - part 1", lectures at Institut de génie nucléaire, Ecole Polytechnique, Montréal.
- [32] L. Mao, I. Zmijarevic, "A new Tone's method in APOLLO3® and its application to fast and thermal reactor calculations", *Nuclear Engineering and Technology*, 2017.
- [33] Q. He, L. Cao, "Current-coupled collision probability method", in "Deterministic Numerical Methods for Unstructured-Mesh Neutron Transport Calculation", *Woodhead Publishing*, pp. 61-71, 2021.
- [34] E. A. Villarino and R. J. J. Stamm'ler, "The heterogeneous response method in slab geometry", *Annals of Nuclear Energy*, pp. 429-440, 1984.

- [35] A. Gammicchia "Development and acceleration of a 3D characteristics method including an axial polynomial expansion of cross sections", PhD thesis, Université Paris-Saclay, pp. 33-35, 2021.
- [36] D. R. Gaston, B. Forget, K. S. Smith, L. H. Harbour, G. K. Ridley, G. G. Giudicelli, "Method of Characteristics for 3D, Full-Core Neutron Transport on Unstructured Mesh", *Nuclear Technology*, pp. 931-953, 2021.
- [37] D. Sciannandrone, S. Santandrea, "Tracking strategies in 3D axial geometries for a MOC solver", *SNA + MC 2013 - Joint International Conference on Supercomputing in Nuclear Applications + Monte Carlo*, Paris, 2013.
- [38] A. Galia, "A Dynamic Homogenization Method for Nuclear Reactor Core Calculations", PhD thesis, Université Paris-Saclay, pp. 64-65, 2020.
- [39] E. W. Larsen, R.E. Alcouffe, "Linear characteristic method for spatially discretizing the discrete ordinates equations in (X,Y)-geometry", United States, 1981.
- [40] K. A. Mathews, R. L. Miller and C. R. Brennan, "Split-Cell, Linear Characteristic Transport Method for Unstructured Tetrahedral Meshes", *Nuclear Science and Engineering*, pp. 34-89, 2000
- [41] B. M. Minor and K. A. Mathews, "Exponential Characteristic Spatial Quadrature for Discrete Ordinates Radiation Transport with Rectangular Cells", *Nuclear Science and Engineering*, pp. 165-186, 1995.
- [42] E. Masiello, R. Lenain, W. Ford, "3D heterogeneous Cartesian cells for transport-based core simulations", *Annals of Nuclear Energy*, Volume 142, 2020.
- [43] E. Masiello, R. Sanchez, I. Zmijarevic, "New Numerical Solution with the Method of Short Characteristics for 2-D Heterogeneous Cartesian Cells in the APOLLO2 Code: Numerical Analysis and Tests", *Nuclear Science and Engineering*, pp. 257-278, 2009.
- [44] Y. S. Ban, E. Masiello, R. Lenain, H. G. Joo, R. Sanchez, "Code-to-code comparisons on spatial solution capabilities and performances between nTRACER and the standalone IDT solver of APOLLO3", *Annals of Nuclear Energy*, pp. 573-594, 2018.
- [45] E. Masiello, T. Rossi, "Improvements of the boundary projection acceleration technique applied to the discrete-ordinates transport solver in XYZ geometries", *International Conference on Mathematics and Computational Methods Applied to Nuclear Science and Engineering, M&C 2013*, Sun Valley, United States, 2013.
- [46] E. Masiello, "On the fly stabilization of the Coarse-Mesh Finite Difference acceleration for multidimensional discrete-ordinates transport calculations", *Journal of Computational Physics*, Volume 373, pp. 1-27, 2017.
- [47] K. Devan, Abhitab Bachchan, "Physics of Nuclear Reactors", in "Homogeneous reactor and neutron diffusion equation", *Academic Press*, 2021.
- [48] K. Assogba, L. Bourhrara, I. Zmijarevic, G. Allaire, "Precise 3D reactor core calculation using Spherical Harmonics and Discontinuous Galerkin Finite Element Methods", *PHYSOR - International Conference on Physics of Reactors*, Pittsburgh, United States, 2022.

- [49] W. Boyd, S. Shaner, L. Li, B. Forget, K. Smith, "The OpenMOC method of characteristics neutral particle transport code", *Annals of Nuclear Energy*, pp. 43–52, 2014.
- [50] P.-L. Lions, "On the Schwarz alternating method III: a variant for nonoverlapping subdomains", *Third International Symposium on Domain Decomposition Methods for Partial Differential Equations*, pp. 202–223, Philadelphia, United States, 1990.
- [51] N. Z. Cho, G. S. Lee, C. J. Park, "A Fusion Technique of 2-D/1-D Methods for Three-Dimensional Whole-Core Transport Calculations", *Proceedings of the KNS-KARP Joint spring meeting*, Republic of Korea, 2002.
- [52] MPACT Team, "MPACT Theory Manual", Technical Report, Oak Ridge National Laboratory, United States, 2019.
- [53] P. Cattaneo, "Development of a Methodology for the Multiphysic Best-Estimate Modelling of PWR Core along irradiation", PhD thesis, Université Grenoble Alpes, 2020.
- [54] P. J. Turinsky, R. M. K. Al-Chalabi, P. Engrand, H. N. Sarsour, F. X. Faure and W. Guo, "NES-TLE: Few-group neutron diffusion equation solver utilizing the nodal expansion method for eigenvalue, adjoint, fixed-source steady-state and transient problem", Electric Power Research Center, North Carolina State University, Raleigh, NC 27695-7909, 1994.
- [55] O. C. Zienkiewicz, R. L. Taylor, J. Z. Zhu, "The Finite Element Method: Its Basis and Fundamentals", *Elsevier Butterworth-Heinemann*, 2005.
- [56] A. Moorthi, Anil Kumar Sharma, K. Velusamy, "A review of sub-channel thermal hydraulic codes for nuclear reactor core and future directions", *Nuclear Engineering and Design*, Volume 332, pp. 329-344, 2018.
- [57] Y. Jo, N. Z. Cho, "Inline critical boron concentration search with p-CMFD feedback in whole-core continuous-energy Monte Carlo simulation", *Annals of Nuclear Energy*, Volume 120, pp. 402-409, 2018.
- [58] G. Gunow, B. Forget, K. Smith, "Full core 3D simulation of the BEAVRS benchmark with OpenMOC", *Annals of Nuclear Energy*, Volume 134, pp. 299-304, 2019.
- [59] N. Horelik, B. Herman, B. Forget, K. Smith, "Benchmark for Evaluation and Validation of Reactor Simulations (BEAVRS)", *International Conference on Mathematics and Computational Methods Applied to Nuclear Science and Engineering, M&C 2013*, Sun Valley, United States, 2013.
- [60] H. G. Joo, J. Y. Cho, K. S. Kim, C. C. Lee, S. Q. Zee, "Methods and performance of a three-dimensional whole-core transport code DeCART", *Proceedings of PHYSOR*, American Nuclear Society, 2004
- [61] Y. S. Jung, C. B. Shim, C. H. Lim, H. G. Joo, "Practical numerical reactor employing direct whole core neutron transport and subchannel thermal/hydraulic solvers", *Annals of Nuclear Energy*, Volume 62, pp. 357-374, 2013.
- [62] B. Kochunas, B. Collins, S. Simpson, R. Salko et al., "VERA Core Simulator Methodology for Pressurized Water Reactor Cycle Depletion", *Nuclear Science and Engineering*, pp. 217-231, 2017.

- [63] J. C. Gehin, "A Quasi Static Polynomial Nodal Method for Nuclear Reactor Analysis", PhD Thesis, Massachusetts Institute of Technology, pp.59-60, 1992.
- [64] A. Dall'Osso, "Reducing Rod Cusping Effect in nodal expansion method calculations", *PHYSOR 2002*, Seoul, Korea, 2002.
- [65] A. M. Graham, B. S. Collins and T. Downar, "Subplane-based Control Rod Decusping Techniques for the 2D/1D Method in MPACT", *International Conference on Mathematics & Computational Methods Applied to Nuclear Science & Engineering, M&C 2017*, Korea, 2017.
- [66] R. D. Lawrence, "Progress in Nodal Methods for the solution of the neutron diffusion and transport equations", *Progress in Nuclear Energy*, Volume 17, No. 3, pp. 271-301, 1986.
- [67] I. Zmijarevic, "Resolution de l'équation de transport par des methodes nodales et des caracteristiques dans les domaines a deux et trois dimensions", PhD thesis, Université de Provence Aix-Marseille, 1998.
- [68] E. Masiello, "Resolution de l'équation du transport des neutrons par les methodes des elements finis et des caracteristiques structurees appliquees à des maillages heterogenes", PhD thesis, Université d'Evry Val d'Essonne, 2004.
- [69] I. Zmijarevic, "Multidimensional Discrete Ordinates Nodal and Characteristics Methods for APOLLO2 Code", *Proc. Mathematics & Computation, Reactor Physics and Environmental Analysis in Nuclear Applications*, Madrid, Spain, 1999.
- [70] R. Lenain, "Amélioration des méthodes de calcul de cœurs de réacteurs nucléaires dans APOLLO3 : décomposition de domaine en théorie du transport pour des géométries 2D et 3D avec une accélération non linéaire par la diffusion", PhD Thesis, Université Paris Sud - Paris XI, 2015.
- [71] D. Schneider, F. Dolci, F. Gabriel, J.-M. Palau, M. Guillo, B. Pothet, "APOLLO3®: CEA/DEN deterministic multi-purpose code for reactor physics analysis", *PHYSOR 2016 - Unifying Theory and Experiments in the 21st Century*, Sun Valley, United States, 2016.
- [72] M. Falabino, "Contribution à la mitigation de l'effet de raie dans les solveurs de transport basés sur les ordonnées discrètes", PhD Thesis, Université Grenoble Alpes, 2022.
- [73] M. Nowak, D. Mancusi, D. Sciannandrone, E. Masiello, H. Louvin, "Accelerating Monte Carlo Shielding Calculations in TRIPOLI-4 with a Deterministic Adjoint Flux", *Nuclear Science and Engineering*, 2019.
- [74] M. M. Scaffoni, M. J. Abbate, "A Group by Group Upscattering Scaling Method", *Annals of Nuclear Energy*, Volume 13, Issue 2, pp. 101-106, 1986.
- [75] R. Lenain, E. Masiello, F. Damian, R. Sanchez, "Domain decomposition method for 2D and 3D transport calculations using hybrid MPI/OpenMP parallelism", *Joint International Conference on Mathematics and Computation, Supercomputing in Nuclear Applications and the Monte Carlo Method*, Nashville, United States, 2015.
- [76] E. Masiello, B. Martin, J. M. Do, "Domain decomposition and CMFD acceleration applied to discrete-ordinate methods for the solution of the neutron transport equation in XYZ geometries", *International Conference on Mathematics and Computational Methods Applied to Nuclear Science and Engineering, M&C 2011*, Rio de Janeiro, Brazil, 2011.

- [77] C. Patricot, "THEDI: a multi-1D two-phase flow solver for neutronic codes", *Proceedings of ICAPP2019*, Juan-les-Pins, France, 2019.
- [78] C. Patricot, R. Lenain, D. Caron, "Upgrade of APOLLO3® internal thermohydraulic feedback model with THEDI, and application to a control-rod ejection accident", *International Conference on Mathematics and Computational Methods applied to Nuclear Science and Engineering, M&C 2019*, Portland, United States, 2019.
- [79] J. H. Mahaffy, "A stability-enhancing two-step method for fluid flow calculations", *Journal of Computational Physics*, Volume 46, Issue 3, pp. 329-341, 1982.
- [80] E. Masiello, R. Lenain, W. Ford, "Linear Short-Characteristics Applied to 3D Heterogeneous Cartesian Cells for Transport-Based Core Simulations", *International Conference on Mathematics and Computational Methods applied to Nuclear Science and Engineering, M&C 2019*, Portland, United States, 2019.
- [81] K. Takeuchi, "A Numerical Method for Solving the Neutron Transport Equation in Finite Cylindrical Geometry", *Journal of Nuclear Science and Technology*, Volume 6, pp. 446-473, 1969.
- [82] R. Grove, "The Slice Balance Approach (SBA): A Characteristic-Based, Multiple Balance S_N Approach on Unstructured Polyhedral Meshes", *International Topical Meeting on Mathematics and Computation, Supercomputing, Reactor Physics, and Nuclear and Biological Applications*, Avignon, France, 2005.
- [83] E. E. Lewis, M. A. Smith, B. C. Na, "OECD/NEA Benchmark on (MOX-fuelled Core) Transport Calculations without Spatial Homogenisation", *Progress in Nuclear Energy*, Volume 45, Issues 2-4, Guest Editors, 2004.
- [84] T. H. Fanning, G. Palmiotti, "Variational nodal transport methods with heterogeneous nodes", *Nuclear Science and Engineering*, Volume 127, pp. 154-168, 1997.
- [85] M. A. Smith, N. Tsoufanidis, E. E. Lewis, G. Palmiotti, T. A. Taiwo, "A finite subelement generalization of the variational nodal method", *Nuclear Science and Engineering*, Volume 144, pp. 36-46, 2003.
- [86] H. S. Joo, "Resolution of the Control Rod Cusping Problem for Nodal Methods", PhD Thesis, Department of Nuclear Engineering, MIT, Cambridge, MA, USA, 1984.
- [87] R. Sanchez, N. J. McCormick, "A Review of Neutron Transport Approximations", *Nuclear Science and Engineering*, Volume 80, Issue 4, pp. 481-535, 1982.
- [88] M. J. Halsall, "A Computer Code for Light Water Reactor Lattice Calculations," Tech. Rep. AEEW - R1498, AEEW, 1982.
- [89] R. Sanchez, J. Mondot, Z. Stankovski, A. Cossic, I. Zmijarevic, "Apollo2: A user-oriented, portable, modular code for multigroup transport assembly calculations", *Nuclear Science and Engineering*, Volume 100, Issue 3, pp. 352-362, 1988.
- [90] R. Stamm'ler, J. J. Abbate, J. Maximo, "Methods of Steady-State Reactor Physics in Nuclear Design", Academic Press Limited, 1983.

- [91] J. R. Askew, F. J. Fayers, P. B. Kemsell, "General Description of the Lattice Code WIMS", J. Brit. Nucl. Energy Soc., pp. 564-585, Volume 5, 1966.
- [92] R. Sanchez, I. Zmijarevic, M. Coste-Delclaux, E. Masiello, S. Santandrea, E. Martinolli, et al., "Apollo2 year 2010", *Nuclear Engineering and Technology*, Volume 42, pp. 474-499, 2010.
- [93] E. Masiello, R. Sanchez, I. Zmijarevic, "New Numerical Solution with the Method of Short Characteristics for 2-D Heterogeneous Cartesian Cells in the APOLLO2 Code: Numerical Analysis and Tests", *Nuclear Science and Engineering*, Volume 161, Issue 3, pp. 257-278, 2009.
- [94] D. Habault, "Chapter 5 - Analytic Expansions and Approximation Methods", in "Acoustics" (P. Filippi, D. Habault, J.-P. Lefebvre, A. Bergassoli, eds.), London: Academic Press, pp. 159-188, 1999.
- [95] W. Ford, E. Masiello, C. Calvin, F. Févotte, "The Response Matrix Acceleration: A New Non-Linear Method for the 3D Discrete-ordinates Transport Equation", *Annals of Nuclear Energy*, Volume 146, p. 107642, 2020.
- [96] E. Masiello, W. Ford, D. Tomatis, "Domain-Decomposition Method in the IDT solver of the APOLLO3 code: synergy of the CMFD and multigroup transport iterations for 3D Full- Core simulations", *PHYSOR 2020: Transition to a Scalable Nuclear Future*, Cambridge, United Kingdom, 2020.
- [97] J. Vidal, K. Folicher, P. Archier, A. Hebert, L. Buiron, J. Palau, S. Pastoris, D. Raynaud, "New Reference APOLLO3® Calculation Scheme for light Water Reactors – Analysis of the BEAVRS benchmark", *EPJ Web of Conferences* 247, Volume 161, Issue 3, pp. 257- 278, 2021.
- [98] A. Galia, R. Sanchez, I. Zmijarevic, "A Dynamic Homogenization Method for 3D Core Calculations", *Annals of Nuclear Energy*, Volume 160, p. 108360, 2021.
- [99] J. M. Palau, et al., "3D Analysis of the UH1.2 Mock-up Experiment Using the APOLLO3 IDT Method and Comparisons Against Tripoli4 Monte Carlo Reference Calculations", *PHYSOR 2018, Reactor Physics Paving the Way Towards More Efficient Systems*, Cancun, Mexico, 2018.
- [100] A. Godfrey, "Vera Core Physics Benchmark Progression Problem Specifications", tech. rep., CASL, 2014.
- [101] M. S. Milgram, "On the properties of collision probability integrals in annular geometry", *Journal of Mathematical Physics*, Volume 18, pp. 2456-2467, 1977.
- [102] G. Marleau, "Dragon Theory Manual - Part 1: Collision Probability Calculations", tech. rep., 2001.
- [103] J. Han-gyu, "Collision Probability Method", in "Analysis of Statistic Reactor Characteristics", Lecture Note 1, Seoul National University, Republic of Korea.
- [104] A. Yamamoto, Y. Kitamura, Y. Yamane, "Computational efficiencies of approximated exponential functions for transport calculations of the characteristics method", *Annals of Nuclear Energy*, Volume 31, Issue 9, pp. 1027-1037, 2004.
- [105] E. Brun, et al., "Tripoli-4®, CEA, EDF and AREVA reference Monte Carlo code", *Annals of Nuclear Energy*, 2012.

- [106] T. Albagami, et al., "TVA Watts Bar Unit 1 Multi-Physics Cycle Depletion Benchmark - Specification and Support Data, Version 2.1", *OECD Nuclear Energy Agency*, 2020
- [107] Nuclear Energy Agency, "Neutronics/Thermal-hydraulics Coupling in LWR Technology", Volume 1, CRISSE-S - WP1: Data Requirements and Databases Needed for Transient Simulations and Qualification - 5th EURATOM Framework Programme, 2004.
- [108] J.-M. Labit, N. Marie, O. Clamens, E. Merle, "Multiphysics CATHARE2 modeling and experimental validation methodology against CABRI transients", *Nuclear Engineering and Design*, Volume 373, 2021.
- [109] B. Faure, P. Archier, J.-F. Vidal, L. Buiron, "A 2D/1D algorithm for effective cross sections generation in fast reactor neutronic transport calculations", *Nuclear Science and Engineering*, Volume 192, pp. 40-51, 2018.
- [110] A. Galia, R. Sanchez, I. Zmijarevic, "A method of dynamic homogenization: application to 2D core calculation", *Annals of Nuclear Energy*, Volume 151, pp. 107774, 2021.
- [111] M. Coste-Delavaux, "Modélisation du phénomène d'autoprotection dans le code de transport multigroupe APOLLO2", PhD Thesis, 2006.
- [112] M. Coste-Delclaux, C. Jouanne, C. Mounier, "Current status of the verification and processing system GALILÉE-1 for Evaluated Data", *EPJ Web of Conferences*, Volume 284, 2023.
- [113] V. Uruba, "On Reynolds number physical interpretation", *AIP Conference Proceedings*, Volume 2000, 2018.
- [114] G. Rowlands, "Resonance absorption and non-uniform temperature distributions", *J. Nuclear Energy, Pts. A & B. Reactor Sci. and Technol*, 1962.
- [115] R. Delvaux, "Algorithmes de couplage entre neutronique, thermohydraulique et thermique", PhD thesis, 2022.
- [116] Alain Hébert, "Développement de la méthode SPH: Homogénéisation de cellules dans un réseau non uniforme et calcul des paramètres de réflecteur", PhD thesis, 1981.
- [117] I. Toumi, D. Gallo, and P. Bourgeois, "FLICA-4 code: 3D thermal-hydraulic calculations", Japan, 1997.
- [118] V. Marelle, P. Goldbronn, S. Bernaud, E. Castelier, J. Julien, et al., "New developments in ALCYONE 2.0 fuel performance code", *Top Fuel 2016 - Light Water Reactor (LWR) Fuel Performance Meeting*, United States, 2016.