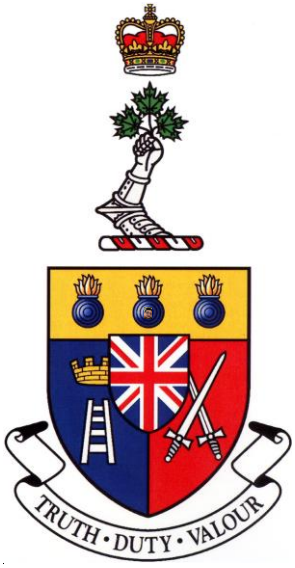
FUELLING STUDY OF A CANDU REACTOR USING FUEL CONTAINING BURNABLE NEUTRON ABSORBERS

ÉTUDE DE L'ALIMENTATION D'UN RÉACTEUR CANDU AVEC DU COMBUSTIBLE CONTENANT D'ABSORBANTS DE NEUTRONS



A Thesis Submitted

to the Division of Graduate Studies of the Royal Military College of Canada

by

Jason Jaeho Song, B.Eng, rmc
Sub-Lieutenant

In Partial Fulfillment of the Requirements for the Degree of
Master of Applied Science in Nuclear Engineering

April 2015

Acknowledgements

First and foremost, I express my greatest appreciation for my supervisors, Dr. Paul K. Chan and Dr. Hugues W. Bonin, for their excellent guidance and mentorship throughout the conduct of this research. Their profound knowledge and leadership were instrumental in ensuring the success of this work, and their personable characters made it truly a pleasure to work with them.

I extend my gratitude to Dr. Benjamin Rouben, Dr. Wei Shen, and Dr. Zlatko Catovic for their sincere and helpful assistance in utilizing and adapting the industry standard toolset codes for the methodology used in my research. I greatly appreciate their guidance as it has significantly accelerated the progress of my research. I also give my thanks to my fellow graduate students at RMCC who provided the best of company both in and outside the office, and ungrudgingly helped me refine the contents of my thesis and its oral defence.

I acknowledge the CANDU Owners Group Inc. (COG) and the Natural Sciences and Engineering Research Council of Canada (NSERC) for their provision of funds and resources that greatly helped propel this project forward. Their support is very much appreciated.

Finally, I am deeply grateful to the Defence Research and Development Canada (DRDC) for providing me with the opportunity to pursue this research endeavour. Without the support of DRDC, I would not have had the chance to pursue my post-graduate studies, so they have my utmost appreciation.

Abstract

Song, Jason (Chemical Engineering). Royal Military College of Canada. February 2015. Fuelling Study of a CANDU Reactor using fuel containing burnable neutron absorbers. Supervisors: Dr. Paul K. Chan and Dr. Hugues W. Bonin

A fuelling study for a CANDU reactor was conducted using 37-element natural uranium (NU) fuel, augmented with burnable neutron absorbers to mitigate the effects of fuelling transients and improve power compliance margin during refuelling. The burnable absorbers of interest include Gd_2O_3 and Eu_2O_3. Various quantities of the two neutron absorbers are considered, and they are added homogeneously to the CANLUB coating of fuel bundles. The study was conducted to demonstrate the gain in operating margins of CANDU reactors incurred by implementing the neutron absorber containing fuel design.

The computer code Reactor Fuelling Simulation Program (RFSP) developed by the Canadian Nuclear Laboratories (CNL), previously Atomic Energy Canada Limited (AECL), was utilized to simulate the fuelling of the 480 channels, with each channel containing 13 fuel bundles. This is one of the CANDU reactor designs used in Ontario. Simulations were conducted for the 37-element NU fuel bundle design, with and without the neutron absorber modification to demonstrate the relative gain in margin incurred by the addition of burnable absorbers.

Fuel safety and performance are always important topics of consideration for nuclear utilities and regulatory bodies. The results of this study present improvements in the operating margin and the economy of neutrons for the considered CANDU reactor at the expense of a reasonably simple change to the fuel design.

Keywords: CANDU, On-power Refuelling, Fuel Management, Neutron Absorbers, Gadolinium, Europium, Fuelling Transient, Plutonium Peak, Axial Flux Flattening, Radial Flux Flattening, Reactor Safety, Reactor Aging.

Résumé

Song, Jason (Génie Chimique). Collège Militaire Royal du Canada. Avril 2014. Étude de l'alimentation d'un Réacteur CANDU avec du Combustible Contenant d'absorbants de Neutrons. Superviseurs: Dr. Paul K. Chan et Dr. Hugues W. Bonin.

Une étude de l'alimentation en combustible pour les réacteurs CANDU a été réalisée pour des grappes à 37 crayons contenant de l'uranium naturel ainsi que de petites quantités d'absorbeurs de neutrons pour atténuer l'apparition de transitoires lors de l'insertion de grappes neuves et améliorer la marge de conformité de puissance pendant l'insertion de grappes neuves. Les absorbants de neutrons d'intérêt sont le Gd_2O_3 et le Eu_2O_3, qui sont ajoutés à la couche CANLUB du combustible en quantités variables. L'étude a été réalisée pour démontrer le gain de marge de réacteurs CANDU sujets à la mise en œuvre du combustible ainsi modifié par l'ajout d'absorbeurs de neutrons.

Le logiciel Reactor Fuelling Simulation Program (RFSP) développé par les Laboratoires Nucléaires Canadiens (anciennement Énergie Atomique du Canada Limitée) a été utilisé pour simuler l'alimentation en combustible d'un réacteur CANDU utilisé en Ontario. Les simulations ont été réalisées pour la grappe de combustible non modifiée à 37 crayons contenant de l'uranium naturel et le nouveau combustible muni d'absorbeurs de neutrons consommables pour tester le gain relatif de la marge opérative.

La sécurité et la performance du combustible sont toujours des considérations importantes pour les fournisseurs d'électricité et les organisations de règlementation pour l'énergie nucléaire. Les résultats de cette étude présentent les améliorations à la marge et l'économie de neutrons de réacteurs CANDU au prix d'une modification simple.

Mots-clefs : CANDU, Alimentation de combustible en-ligne, Gestion du combustible, Absorbeurs de neutrons, Gadolinium, Europium, Transitoire suite à l'insertion de grappes neuves, Pic de plutonium, Aplatissement du flux axial, Aplatissement du flux radial, Sûreté des réacteurs, Vieillissement des réacteurs.

Table of Contents

List of Tables

List of Figures

Glossary

4-Bundle Shift:	Fuelling mode which involves pushing 4 fresh bundles in a given channel and withdrawing 4 irradiated bundles at the other end of the same channel
8-Bundle Shift:	Fuelling mode which involves pushing 8 fresh bundles in a given channel and withdrawing 8 irradiated bundles at the other end of the same channel
Adjuster Rod:	Reactivity devices which may be inserted to flatten the core flux shape in the axial direction and also assist in flattening the flux in the radial direction
Aging:	Time and environmental stress driven process via which the safe function of reactor systems, structures and components degrade
Axial Tilt	The ratio between the combined powers of fuel bundles contained within the frontal half of the channel and the rear half of the channel. This is used to gauge the degrees from which the power of the channel is offset from being level along its axis. The front and the rear halves of the reactor core are arbitrarily chosen
Bundle Power	The power produced by a single fuel bundle, which in this study is represented by the power of the lattice in the RFSP core representing the respective fuel bundle
Bi-directional Fuelling:	Fuelling methodology which involves fuelling adjacent pairs of channels in opposing directions to achieve axial symmetry

Channel Power	The total power of a fuel channel. The sum of the powers of all fuel bundles within a fuel channel
Channel Power Peaking Factor:	Ratio between a channel's current instantaneous operating power and its reference (desired) power
Control Rod:	Reactivity devices which may be inserted (usually part-way) to provide negative reactivity in its vicinity and lower neutron flux and reactor power as required
Core Flattening:	The practice of flattening the flux and power profile of the core
Derating:	The act of reducing a reactor's total power output by engaging its neutron absorbing devices
Differential Fuelling:	Fuelling high flux, high power regions at slower rates than for low flux, low power regions to achieve core flattening in the radial direction
Dryout:	Condition of a region of fuel channel that loses sufficient liquid coolant inventory such that the fuel bundle(s) in the region experience complete loss of contact with the liquid coolant, leaving vapours or gas to carry out the thermal energy, which is a much less efficient in cooling the fuel elements
Fuel Burnup:	A measure of energy produced by the fuel, usually given in MW hours per kg of uranium
Fuelling Outage:	Necessary reactor shutdown which occurs in the process of bulk refuelling operation for reactors that cannot be refuelled on-line
Fuelling Transient:	A phenomenon which occurs during the first 2-3 FPDs following refuelling when a transient peak in

	power occurs in the vicinity of freshly loaded fuel bundles
Full Power Day:	Operation of a reactor at full power for 24 hours
Irradiation:	Measure of exposure to neutron flux during a given time. It is given by the time integral of the flux and is expressed in units of cm^{-2} or "neutrons per kilo-barn."
Liquid Zone Controller:	Reactivity devices consisting of chambers which may be filled up with light water to provide negative reactivity in its vicinity and lower neutron flux and power as required.
Negative Reactivity:	Parasitic absorption of neutrons measured in terms of its reactivity sink. It is often used to describe the reactivity sink worth of reactivity devices or highly irradiated fuels
Neutron Absorber:	Generalized term for indicating materials possessing high neutron absorption cross-sections
Operating Margin:	The margin between the instantaneous value of an operating parameter and its licensed operating limit.
Plutonium Peak:	A phenomenon which occurs for NU fuels at approximately at 40-50 FPDs of in-core irradiation when a transient peak in power occurs in the vicinity of affected fuels due to the accumulation of fissile plutonium isotopes
Power Density:	Ratio of the local power level to the volume of the part of the reactor considered
Radial Form Factor:	The ratio between the value of the highest powered channel in the core to the average value of channel

	powers in the core. It is used as an indicator of how flat the distribution of power is in the core
Reactivity:	The measure of reactor's departure from criticality
Reactivity Bank:	The excess reactivity available within the core, which may be used to operate the reactor without refuelling for a period of time
Reactivity Device:	Device used to introduce positive or negative reactivity into the reactor core for means of control. Positive reactivity can be inserted by using neutron producing material. Negative reactivity can be inserted using neutron-absorbing materials
Reference Fuelling Scheme	The normal fuelling scheme for the model core used in the study. Consists of 8-bundle-shift in the outer core and 4-bundle-shift in the inner core (Figure 22)
Refuelling Ripple:	A phenomenon which occurs in CANDU reactors as the consequence of daily refuelling. On-power refuelling causes the power density within the core to continuously oscillate due to the effects of the fuelling transient and the plutonium peak
Safety Margin:	An additional margin adopted beyond the extent of the reactor's operating margin. The safety margin exists as a buffer to ensure safety and maintain control of the reactor in the event that the operating margin should be exceeded. The safety margin is not a margin in which the reactor may be operated.
Voiding:	Formation of air pockets (bubbles) or slugs in the coolant

List of Acronyms

AECL	Atomic Energy Canada Limited
AR	Adjuster Rod
AT	Axial Tilt
BNA	Burnable Neutron Absorber
BNAF	Burnable Neutron Absorber Fuel
CANDU™	CANada Deuterium Uranium
CCP	Critical Channel Power
CHF	Critical Heat Flux
CNL	Canadian Nuclear Laboratories
CPPF	Channel Power Peaking Factor
CR	Control Rod
DBA	Design Basis Accident
DBE	Design Basis Events
FFS	Fitness for Service
FPD	Full Power Day
HF	Heat Flux
IST	Industrial Standard Toolset
LLOCA	Large Loss of Cooling Accident
LOR	Loss of Regulation
LZC	Liquid Zone Controllers
NOP	Neutron Overpower Protection
NGS	Nuclear Generating Station
NU	Natural Uranium
OBPL	Operating Bundle Power Limit
OCPL	Operating Channel Power Limit
PHTS	Primary Heat Transport System
RFF	Radial Form Factor
RFSP	Reactor Fuelling Simulation Program
RTD	Resistance Temperature Detector

SDS	Shutdown System
SFP	Saturating Fission Product
SLOCA	Small Loss of Cooling Accident
SOR	Shut Off Rod
SSC	Systems, Structures and Components

List of Symbols

k_{eff}	Effective neutron multiplication factor
k_{inf}	Infinite neutron multiplication factor
ρ	Reactivity

Chapter 1: Introduction

In CANDU® (CANada Deuterium Uranium) reactors, refuelling operations are conducted while the reactor is at power. This type of refuelling is known as "on-power" refuelling, and it is a less common feature amongst the arsenal of contemporary reactors whose majority are "bulk" fuelled at the shutdown state. On-power refuelling involves the removal and the replacement of small, discrete quantities of fuel on a daily basis. This presents the advantage to avoid refuelling outages and prevents insertions of high excess reactivity into the lattice [1]. Also, on-power refuelling allows defective fuels to be replaced with fresh fuel bundles quickly after they are detected, which helps to limit the contamination of the primary coolant system. Due to this particular set of advantages, the scope of refuelling operations in CANDU reactors extends beyond the basic function of inserting reactivity into the core. In CANDU reactors, refuelling operations are extensively utilized as the primary mode of long-term reactivity control, as well as the management of the distribution of flux and power densities in the core [1]. Refuelling therefore plays a critical role in the day-to-day operation of CANDU reactors.

The present study investigates a novel modification to the existing CANDU fuel design to improve operating margins and relax some of the existing constraints in the current refuelling practice. The modification involves the doping of the current 37-element natural uranium (NU) fuel with minute quantities of neutron absorbing materials. The purpose of the doping is to allow the presence of neutron absorbers to reduce the magnitude of the refuelling power transients (fuelling transient and plutonium peak) which occur in NU fuels [2]. This is desirable because in many existing CANDU systems, the safety margins of limiting fuel channels have become significantly tighter than their initial design values due to the effects of aging. It can therefore be sometimes challenging to maintain the powers of some of the limiting channels within their operating envelopes during normal operation. For this reason, refuelling engineers devote a large portion of their focus during refuelling to ensure that transient powers caused by the refuelling transients do not exceed the limits of the channels' operating envelopes. In current practice, liquid zone controllers (LZCs) are used routinely to temporarily derate

the powers of channels which approach the limits of their operating envelope. This approach however, is disadvantageous for the economy of neutrons, because LZCs derate power within a large volume of the core rather than within the specific fuel or the channel affected by the refuelling transients. This study follows the recommendations of Paquette *et al* (2014) to use natural gadolinium and natural europium, in their oxide forms, as the added absorbers [2]. The research conducted by Paquette *et al* has demonstrated that minute quantities of Gd_2O_3 and Eu_2O_3 are effective in selectively reducing the magnitudes of the refuelling power transients in fuel lattices while imparting minimal losses to their useful reactivity. It is therefore expected that the effect of the two added absorbers will help to reduce the magnitude of the refuelling power ripples in CANDU cores while imparting minimal losses to the discharge burnup of fuels. In corollary, the transient powers can be managed with greater ease by the use of burnable neutron absorber fuels (BNAFs), which can conserve operating margins and relax the constraints on fuel management posed by tight operating power limits.

Chapter 2: Research Objectives

The primary objective of this study is to demonstrate a gain in the operating margin by implementing the use of BNAF in an existing CANDU reactor. The secondary objective is to test the feasibility of operating a CANDU core with unconventional refuelling schemes by using the advantages created by refuelling with BNAFs. To achieve these objectives, the present study involves computer simulated refuelling and core-following of a CANDU core using BNAFs containing varied quantities of added absorbers, with 37-element NU fuel as reference, such that a relative gain in margin can be determined. The simulations are conducted using the Reactor Fuelling Simulation Program (RFSP), which is an industry standard toolset (IST). The reactor model used for the simulations is a typical CANDU model used at nuclear generating stations. The model core consists of 480 channels, with 13 bundles-per-channel design and the core power of approximately 2650 MW_{th}. The scope of the simulation is limited to equilibrium fuelling, and various scenarios encountered in equilibrium fuelling are simulated. In doing so, increases in ease or flexibility of fuelling to maintain compliant core state are observed for a wide range of fuelling scenarios. The increases in the ease or the flexibility of refuelling are then analyzed and compared with current operating margins of the model core to provide an indication for the overall gain in margin.

Chapter 3: Background

In this chapter, a literature review of the current CANDU fuel management practices, challenges associated with aging, and the concept basis for the research and development of the BNAF are provided. The chapter covers in detail the function performed by refuelling in the day-to-day operation of CANDU reactors, as well as some major limitations associated with its current practice. Also, the aging of CANDU reactors is discussed with regards to its effects on systems, structures and components important to the performance of fuels, and therefore its implications for fuel management. Then, the conceptual basis and implications associated with the BNAF is described to provide a perspective on how it may be used to recover operating margins and thereby relax some of the current limitations associated with the refuelling operations in CANDU reactors.

3.1 Refuelling in CANDU reactors

3.1.1 Bulk Reactivity Control

In nuclear reactor physics, the reactivity is defined as the measure of departure from the critical core state. The critical core state is the ideal state of operation when the rate of the core's neutron loss equals its generation [1]. Therefore, the bulk reactivity of a core essentially governs the rate of generation of neutrons within its boundaries. A well-established control of reactivity within the core is hence an important requirement for the safe and steady operation of all nuclear reactor types. The reactivity in nuclear reactors is primarily controlled in order to steadily operate the core at critical state. This involves ensuring the bulk reactivity of the core is maintained constant with minimal excess or deficiency from the critical state. In practice, the excess reactivity within reactor cores is mitigated by applying "reactivity devices" composed of neutron absorbing material that can be inserted and withdrawn to suppress or increase the reactivity of the core as required. In a typical nuclear reactor, as the fuel elements within the core accumulate fluence or irradiation, the reactivity in the core declines due to the consumption of fissile material and the accumulation of neutron-absorbing fission products (FP). To counteract

this phenomenon and maintain the reactor at critical, high-irradiated fuels must be removed and replaced with fresh fuels [1].

In bulk fuelled reactors (also called batch-fuelling reactors), large quantities of irradiated fuels are replaced at each instance of refuelling. Bulk fuelled reactors require refuelling at low frequencies, because each refuelling operation inserts a large sum of positive reactivity. Bulk refuelling therefore typically results in a large excess reactivity in the freshly fuelled core, which must initially be suppressed by a heavy use of reactivity devices. As fresh fuels become irradiated over time and the core's excess reactivity declines, the devices are disengaged proportionally to sustain the core's criticality. Once the excess reactivity is spent and disengagement of devices can no longer offset the decline in reactivity, the core must be refuelled again.

In CANDU reactors, refuelling operations replace small, discrete quantities of irradiated fuels with fresh fuels on a daily basis [1]. This type of discrete, daily refuelling allows CANDU reactors to operate at constant criticality by refuelling at a rate that is proportional to the rate of reactivity decline within the core [1]. Moreover, due to the small number of fuels replaced, refuelling in CANDU reactors typically results in a small excess reactivity in the core. This allows CANDU reactors to operate with significantly less assistance from reactivity devices than bulk refuelled reactors, which is a great advantage for the economy of neutrons and also safety. The small excess reactivity in CANDU cores are typically controlled using liquid zone controllers (LZCs), which are devices that actively exert spatial and bulk reactivity control throughout the core using incremental volumes of light water as the neutron sink [1]. In practice, however, the bulk reactivity of a CANDU core is primarily controlled by refuelling alone [1]. The use of LZCs provides an added assistance to the control of the bulk reactivity by suppressing the small excess reactivity that is inserted when refuelling [1].

3.1.2 Spatial Reactivity Control (Flux and Power Flattening)

For optimal power production, it is desirable to operate the core with a flat flux shape in both the radial and axial directions, which ensures the distribution of the heat sources within the reactor core is as uniform as possible. This is because a uniform distribution of flux and heat allows all fuel bundles within the core to operate at similar rates of reactions. Hence in an ideal case, the majority of fuel bundles and channels within the core are able reach their ideal operating powers without any exceeding their licensed operating power limits [1]. Otherwise, if the distribution of heat and flux is not uniform, fuel bundles in the core operate at different powers according to their associated amplitudes of heat and flux. The overall power at which the reactor can operate is then limited by the region experiencing the highest heat and neutron flux because its resident fuel bundles will encroach on operating limits while regions of low heat and flux operate at sub-optimal power levels. Hence, depending on the degree from which flux distribution departs from the ideal, flat shape, the power production at the reactor level as a whole can become increasingly sub-optimal.

In addition to controlling the bulk reactivity, refuelling in CANDU reactors is also used to manage the distribution of neutron flux and power density across the core [1]. Fuel bundles at high values of irradiation are known to act as neutron sink and therefore depress the flux of neutrons in their vicinity [1]. The opposite is known to be true for fresh fuel bundles and those with low irradiation [1]. Therefore, by controlling the distribution of highly irradiated fuels and fresh fuels within the core, the distribution of flux and power within the core can be controlled [1]. In practice, however, it is difficult to achieve fine control of the desired distribution of flux and power in the core via refuelling alone. Therefore, the distribution of flux and power within a CANDU core is controlled in practice using adjuster rods and LZCs in addition to strategic refuelling schemes [1]. Adjuster rods are typically fixed in position near the central region of the core to suppress flux and power at the centre of the core. This is done so because the central region of a reactor core experiences the highest magnitude of neutron flux while the periphery experiences lower flux values [1]. The LZCs on the other hand are

distributed symmetrically throughout the reactor core to provide an active control for the spatial distribution of flux and power over the entire core.

3.1.3 Operating Envelope and Margin

Each fuel channel within a CANDU system is licensed with different safe operating power limits depending on its capacity for cooling [3]. The licensing of the safety limits requires a rigorous and detailed safety analysis, typically using highly conservative assumptions [3]. In practice, the safety channel power limits are calculated with a significant margin of safety from the critical channel powers (CCP) at which the critical heat flux (CHF) required for the onset of dryout occurs [4]. The operating channel power limits (OCPL), furthermore, are licensed with additional margins of safety from the safety power limits [3]. To ensure the reactor is protected against excessively high powers within fuels (which could lead to a dryout), a feature known as the regional or neutron overpower protection (ROP/NOP) is included in the design of CANDU reactors [1]. In the event that the power density of a designated ROP/NOP safety channel reaches the trip set point (set below the safety power limits), the ROP/NOP detectors detect the threshold power density and engages two independent shutdown systems to shut down the reactor [1]. During normal operation, each channel in a CANDU core is operated within the envelope of its licensed OCPL. This is achieved by prudent refuelling practices and the assistance of LZCs which provide spatial control via temporarily derating regions of high power densities within the core. In practice, the temporary derating in power is undesirable, and channels are typically operated at optimal "reference powers" below the operating power limits. The difference between the typical operating powers of channels from their operating power limits essentially defines the "operating margins" of the channels. In addition to the OCPL, an operating bundle power limit (OBPL) is also licensed from the safety analysis of CANDU reactors. This is because the mechanism of fuel failure depends on a variety of factors in addition to the fuel and coolant temperatures. The OBPL is imposed on the fuel bundles themselves, because even without occurrences of channel dryout, it is possible for fuels to fail due to unforeseen

variations in manufacturing and operating practices [4]. Hence the licensed OBPL is used to specify the operating power envelope for fuel bundles themselves.

3.1.4 Fuelling Transient

In CANDU reactors, due to the practice of on-power refuelling, the effects of replacing highly irradiated fuels with fresh fuels are reflected immediately in the local flux and power density of the core [1]. The refuelling causes an immediate increase in the local flux and power density in the vicinity of the freshly inserted fuel bundles. This increase occurs in part due to the renewal of the fissile isotopes inventory within the refuelled lattice, but is mostly due to the absence of fission products within fresh fuels [1]. Many fission products possess large neutron absorbing cross sections and therefore act as significant drags on the reactivity of the fuel. There are two different types of fission products: saturating and non-saturating. The saturating fission products are typically generated quickly and reach equilibrium concentrations depending on the flux at which the fuels are irradiated [1]. Non-saturating fission products on the other hand accumulate over time and eventually cause the fuels to become subcritical and become a negative drag on the reactivity of the core [1]. The accumulation of non-saturating fission products in fresh fuels and their eventual lead to fuel sub-criticality are relatively slow and linear processes. The generation of saturating fission products to their equilibrium concentrations, however, is a relatively quick process which results in a transient in the power of fresh fuels [1]. The immediate increase in the local power density resulting from refuelling is quickly followed by a reduction to lower values of power density within the first two to three full power days (FPD) of irradiating the fresh fuels for a typical CANDU reactor. This is because the major saturating fission products such as xenon-135, samarium-149, samarium-151 and rhodium-103 reach their equilibrium concentrations during this relatively short period of time. The local reactivity of fresh fuels during the first two to three FPDs of in-core irradiation therefore follows the shape of a sharp peak, which is also mirrored by its local flux and power density. This phenomenon is referred to as the fuelling transient (FT). The fuelling transient poses an inconvenience for the operation of CANDU reactors because the transient powers in the

vicinity of fresh fuels can sometime approach the limits of their operating envelopes and necessitate a temporary derating in power.

3.1.5 Plutonium Peak

The plutonium peak is a phenomenon which occurs during the in-core lifetime of NU fuels due to their high uranium-238 content. As NU fuels are irradiated within a reactor core, uranium-238 is converted into plutonium-239 that accumulates to a peak value as the rate of its consumption begins to exceed the rate of its generation [1]. The development of this peak is accompanied by a rise in the reactivity of the affected fuel, which causes the power density of the fuel to also rise proportionally. Consequently, the resident fuel channel, as well as its neighbors, also experiences an increase in power density [1]. On average, the plutonium peak occurs between forty to fifty FPDs of in-core irradiation and it represents an inconvenience for power compliance similar to that of the fuelling transient [1]. Figure 1 below outlines the evolution of reactivity within fresh fuels as they are irradiated. The initial peak in reactivity constitutes the fuelling transient. The second, smaller transient in reactivity constitutes the plutonium peak.

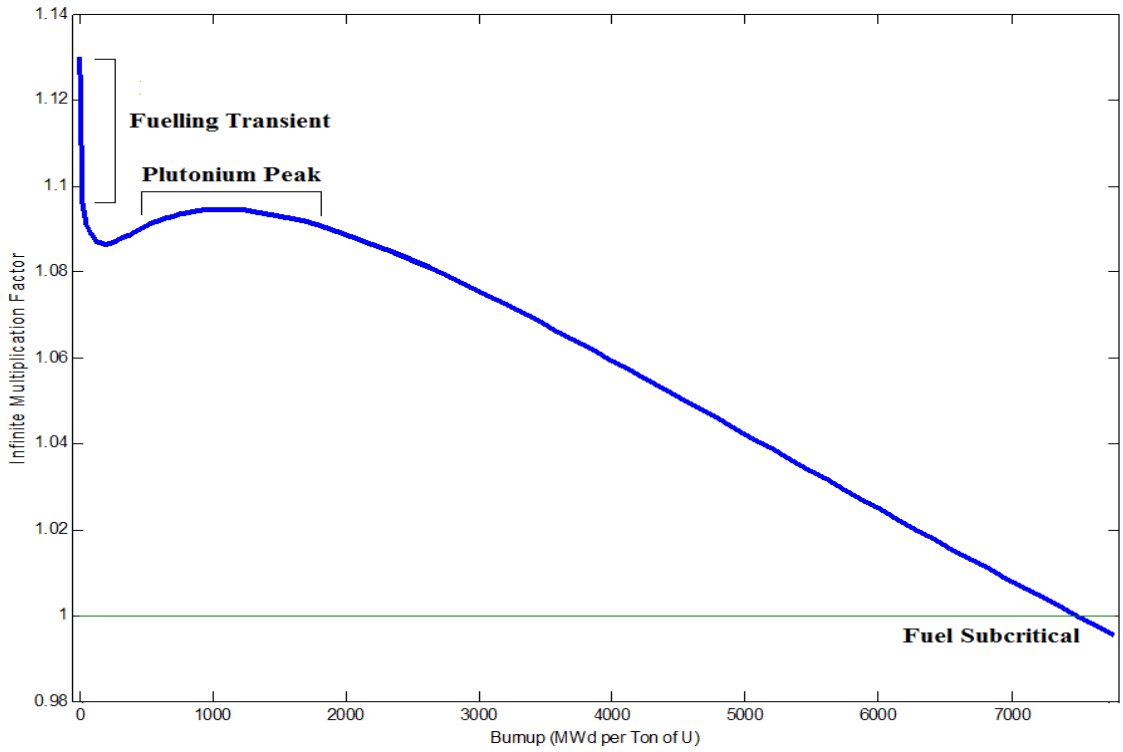


Figure 1. Evolution of reactivity in NU fuels during in-core irradiation.

3.1.6 Refuelling Power Ripples

As fuel bundles in CANDU reactors pass through their respective irradiation cycles, they each experience periods of peak excess reactivity during the occurrence of the fuelling transient and the plutonium peak. Due to the practice of daily refuelling, CANDU reactors are consistently subject to the effects of the two transients on a daily basis. As different locations within the core are refuelled day-to-day, the local power transients manifest throughout the entire core in accordance with the refuelling history. Collectively, this phenomenon is referred to as the "refuelling power ripples." The refuelling power ripples are therefore a "consequence of the daily refuelling of channels and the irradiation cycle through which each channel travels" [1]. In the perspective of fuel management, refuelling power ripples pose a challenge to the task of operating all fuel channels within the core, including the limiting channels, within their licensed operating envelopes. CANDU fuel engineers therefore take into account the impact of the refuelling ripples when selecting channels for refuelling. In practice, the occurrence of high transient powers is managed effectively using LZCs. LZCs are a working solution to mitigating the transient powers caused by the refuelling ripples. However, LZCs derate power in a larger volume of the core than the volume that is typically affected by the fuels experiencing transient powers, which makes them disadvantageous for the reactor's economy of neutrons.

3.2 Aging of CANDU reactors

3.2.1 Effect on Safety

Maintaining a high standard of safety is a priority when operating nuclear generating stations. One key component of this process is the management of age-related degradation in plant systems, structures and components (SSCs) important to safety [5]. For CANDU reactors, this is an ongoing concern of present day, as the major proportion of CANDU fleet have been in service for long times, ranging from twenty to forty years [6]. During their operating lifetimes, many of their SSCs are exposed to the influence of

extreme stress, temperature, irradiation and other factors such as cycling that can cause material degradation over time [5]. Although not significant at first, the material degradation eventually results in levels of deterioration in SSCs that affect their safe function. At present, important components and structures of several existing CANDU reactors are approaching the end of their designed service life [5]. Hence to avoid accidents and ensure safe operation, it is necessary to re-evaluate the fitness-for-service (FFS) of aged plant systems with regards to their safety analysis margins.

3.2.2 Impact on Fuel Performance

Aging imparts multitudes of negative effects on the safe function of various SSCs. Numerous affected SSCs and the impact of aging on their performances are outlined in detail in Table 1. In this present study, the impact of aging of SSCs important to fuel performance is considered in particular, because it dictates how fuel management must be adapted. It is very difficult to define precisely how the performance of fuels in CANDU reactors is affected by aging. This is because the degradation of the fuel performance due to aging results from the combined effects of several aging phenomena that influence multiple components and structures, and degrades the operating envelopes of systems critical to fuel performance [7]. Moreover, the exact impact of aging is difficult to measure and quantify, because the aging phenomena are not fully understood [7]. In this context, it makes sense to assess the integrated impact of aging with respect to changes in safety analysis margins [7]. Therefore, the impact of aging on fuel performance is assessed in large with regards to the changes that it imparts to the primary heat transport system (PHTS), because they affect CCP calculations under accident conditions. This way, it is possible at the very least to ensure that any narrowing of margins for dryouts are compensated and fuel failures in both normal operating conditions (NOC) and design base accidents (DBA) are prevented with sufficient safety margins [7].

Table 1. Summary of aging issues: Effect on trip coverage and safety concerns (Tezel *et al*., 2010) [7].

Aging Issue	Effect	Safety Concern
PHTS pipes roughness changes	Core flow re-distribution (selective Magnetite dissolution & deposition)	Fuel/pressure boundary failures, safety margins
Aging of the pump bowl and impeller	Change in pump run-down curve	Low flow trip may be adversely effected
Steam Generator: Tube fouling, thinning, plugging or crimping, divider plate leak and/or break	Reduced heat transfer to boilers, higher pressure drop on the tube side (fitness requirement	Reduction in PHTS flows, fuel and pressure boundary failures, Reduction in safety margins, Fission products by-pass of containment
Feeder thinning (multiple feeder break is not a DBA)	Increased Channel flow, corrosion products and debris in the PHTS, Feeder Break is a DBE. However, Multiple feeders break as the initiating event has not been analyzed. It may be argued that the impact may be somewhere between a SLOCA and LLOCA, both of which have been analyzed	Low flow trip, Potential fuel and pressure boundary failures, Reduction of safety margins, Single or multiple channel flow blockage, Multiple channel flow blockage has not been analyzed
Pressure Tube axial & radial creep, and sag	Core volume increases, flow-bypass, decrease in yield strength, falling off channel bearing, CHF Reduction, Operating with depleted fuel bundles, (CHF and reactor physics issues), increase positive coolant void reactivity, bundle power increase, radial bundle power peak factor increase; pin power increase, changes to bundle powers uncertainty, changes in burn-up, and flux at detector locations; changes in the limiting flux shape in a slow LOR event, SDS1 shut off rod drop reactivity curve may change	Adversely affect trip coverage for all DBEs Overall trip coverage including LLOCA power pulse issues, Fuel/fuel channel integrity
Aging of other structures inside and outside the core	In-Core: Potential elongation and bending of the guide tube due to neutron fluence, temperature, etc., Out-Core: Support structures	Adverse effect on SDS performance (regular interval testing is performed)
Passive or active component aging (valve passing/seating, corrosion etc.)	Valve passing/seating issues, Change in the leakage characteristics of check valves, Thermocouple degradation, etc.	Safety and safety related systems (Surveillance and regular interval testing is done)
Aging of the Instrumentation	Corrosion (RTD wells), Orifice degradation, Changes in the impulse lines, Electronic component degradation	Trip coverage, reactor trip delays, Spurious trips, post-accident reactor monitoring issues.
Operation with empty channels	Empty Channels (BA U3), Operation with depleted fuel bundles at certain channels, HF issues	Safety has been re-assessed

3.2.3 Aging of PHTS

Aging of the PHTS of CANDU reactors plays a critical and arguably the most direct role in dictating the degradation of fuel performance. This is because the PHTS serves the function of circulating heavy water coolant through fuel channels at an optimal flow rate, flow distribution and pressurization [5]. Maintaining optimal values of coolant flow parameters is critical to fuel performance, because the coolant flow properties dictate the rates of heat removal from fuels as well as the onset of dryout (voiding). Formation of voids is a significant phenomenon because although the overall temperature-related coefficient of reactivity for CANDU reactors is negative, the coefficient associated with the formation of voids is positive. Hence when voids initially begin to form via nucleate boiling, a positive feedback on reactivity is established, which decreases the inherent safety factor provided by the overall negative reactivity coefficient to increases in temperature. In practice, the NOP feature of CANDU reactors is set to trip well before the onset of dryout and shut down the reactor with a significant margin of safety.

The issue of aging with regards to the PHTS is that components (that are important to coolant flow parameters) such as feeders become foul or rough due to flow-accelerated corrosion, and pressure tubes sag and creep over time due to exposure to heat, the load of fuels and coolant within, and pressure [7]. Also, the steam generators may experience tube fouling, thinning, plugging or crimping, which can result in reduction of flow in the PHTS. Even the instrumentation of any given components may degrade and produce inaccurate information [7]. If any of such phenomena occurs, the coolant flow parameters within fuel channels will depart from ideal values, which will affect the heat flux and power conditions at which voiding will commence. In the case that flow rates within fuel channels are impaired due to feeder corrosion or steam generator tube fouling, the rate of heat removal from the fuel bundles decreases accordingly and causes the reactor inlet header temperature to increase. Subsequently, the coolant's temperature margin to the onset of dryout is decreased. In terms of safety analysis, this reduces the critical channel power (CCP) at which the critical heat flux (CHF), the heat required for

onset of dryout, occurs. Therefore, to compensate this loss in safety margin, the NOP set point must be lowered to a more conservative value, thereby essentially reducing the operating margins. Pressure tube diametral creep, on the other hand, results in reduced coolant flow in the bottom and central regions of affected fuel channels. This is because increases in the diameters of fuel channels create paths of lower resistance (for coolant flow) in the top regions as fuel bundles shift from being positioned concentric to the fuel channels to resting on the bottom. This results in bottom and central regions of affected fuel bundles experiencing less effective cooling, therefore causing decreases in the safety margins. In all cases, the NOP set point must be updated to adequately account for these losses in safety margin described above.

3.2.4 Implication for Fuel Management

The implication for fuel management from the aging of the CANDU PHTS is that the power density across the lattice must be controlled more closely. This is because the safety margins will have decreased in proportion to the reduced capacity of the PHTS. This means ensuring compliance to operating power limits during transient powers caused by refuelling ripples becomes more challenging. The LZCs will therefore see a larger use to ensure compliance to the tighter operating envelopes against transient powers. Overall, the demand for increased conservatism in the control of power density leads to tighter constraints on refuelling operations to ensure compliance to power limits. The degree of freedom available in selecting fuel channels for refuelling is therefore significantly reduced. Accordingly, flattening the power distribution in the core, or optimizing other decision parameters, when making refuelling decisions, becomes even more difficult.

3.3 Burnable Neutron Absorber Fuel

3.3.1 Concept Basis for BNAF

Due to the aging of many existing CANDU reactor systems, there is a large incentive to developing improved fuel designs to assist with meeting compliance to operating power limits during refuelling. The BNAF was first conceived in consideration of the use of burnable absorbers in light water reactors to suppress and control high excess reactivities at initial startup in high fuel exposure cores [8], [9]. In considering this utility for reactivity control, it is proposed that minute quantities of burnable absorbers, inserted directly into CANDU fuels, could potentially mitigate the effects of the refuelling transients with negligible losses in fuel burnup. The burnable absorbers to be incorporated would include minute quantities of fast-burning absorbers to specifically suppress the initial fuelling transient, as well as minute quantities of slow-burning absorbers to reduce and delay the plutonium peak. It is expected that by adding the right combination and quantity of fast-burning and slow-burning absorbers, it is possible to significantly reduce the effects of the two refuelling transients and thereby recover operating margins lost to transient powers during refuelling.

3.3.2 Implications for Fuel Management

The BNAF presents several advantages for fuel management in CANDU reactors by reducing the magnitudes of the two refuelling transients. One of the advantages is that it will allow the core to operate closer to a steady state, as the magnitudes of refuelling power ripples will be reduced proportionally. Additionally, it will also relax many constraints associated with the process of channel selection for refuelling. This means some high-power, limiting channels that otherwise cannot be refuelled, due to the risk of exceeding the operating envelope, may be available for refuelling using BNAFs. Additionally, regions containing high-burnup fuels that otherwise cannot be refuelled, because high power ramps expedite stress corrosion cracking, can be refuelled with BNAFs. Another advantage is the possibility of inserting larger quantities of fuel bundles

when refuelling, without causing unduly large power transients. This would allow a mode of refuelling based on a larger number of fresh fuel bundles than for the current established reference fuelling scheme. Such increases in the degrees of freedom associated with refuelling are desirable because they allow fuel engineers to make refuelling decisions with improved impact on the economy of fuel utilization, reactor safety, with an increased flexibility of the refuelling process. The economic impact comes from maximizing the discharged fuel burnup and flattening the power profile of the core via refuelling to produce power more efficiently. The increased freedom allows the refuelling engineers to optimize the refuelling schedule to minimize strain on the fuelling machines and bank reactivity when necessary. Moreover, it provides the refuelling engineers with increased flexibility for refuelling during some uncommon operating situations, such as refuelling a startup core, or replacing all the fuel bundles within a given fuel channel when defective fuels are detected.

In addition to increased freedom in selecting channels for refuelling, the use of BNAFs also decreases the involvement of LZCs within the reactor core. This is because the reduction of refuelling power ripples decreases the requirement for zonal power adjustments. The reduction of the involvement of LZCs provides a significant advantage for CANDU reactors because it increases the economy of neutrons while conserving the safety margin provided by the LZCs. Although this is essentially being done by adding another means of parasitic neutron absorption (BNAs instead of LZCs), suppressing the refuelling ripples with BNAs is more neutron efficient than LZCs. This is because BNAs provide a much finer level of reactivity control at the fuel level, whereas LZCs exert influence over large regions as zonal and bulk control.

Chapter 4: State of the Art

In this chapter, the current state of methods, technologies and limitations associated with the management of fuels in CANDU reactors is discussed. Additionally, the current state of the burnable neutron absorber fuel design and its impact on fuelling ripples, observed via simulations using WIMS-AECL and RFSP, are presented.

4.1 Fuel Management

4.1.1 Bi-directional Fuelling

One of the two major methodologies of fuelling currently in-use for CANDU reactors for spatial reactivity control is bi-directional fuelling. Bi-directional fuelling is incorporated into the current CANDU channel design such that neighboring pairs of channels are refuelled in opposite directions [1]. The pairs of bi-directionally fuelled channels are otherwise refuelled and irradiated under similar conditions, so their axial flux and power distributions manifest in point symmetry to one another. The result of this methodology therefore yields a combined symmetry for the distribution of axial flux and power, in the shape of a cosine, between bi-directionally paired channels [1]. This is desirable as it prevents the axial flux and power from peaking towards a single direction, which would otherwise occur for unidirectional fuelling schemes. Channels are paired bi-directionally to a well-distributed interlocking pattern in the azimuthal direction, such that the cosine shaped, symmetrical distribution of flux is reflected along the axis of the entire core. The actual distribution of flux and power in the axial direction for CANDU reactors follows a flattened cosine shape from the application of reactivity devices known as adjuster rods along the axial centre of the core [1]. Theoretically, it is also possible to shuffle fuel bundles within channels to position high-burnup fuels near the centre of the core to flatten the axial flux and power distribution. However, as CANDU fuel channels are refuelled using a “push through” methodology, and fresh NU fuels with low excess reactivity are used to fuel the core, extensive fuel shuffling is difficult and impractical to be used as a means of spatial control due to the limitations of the refuelling machines [10].

4.1.2 Differential Fuelling

The second major methodology of fuelling currently in use for CANDU reactors for spatial reactivity control is differential fuelling. The purpose of differential fuelling is to flatten the radial flux and power distribution in the core by the means of controlling the distribution of burnup of fuels via selective and varied rates of refuelling [1]. High burnup fuels having accumulated large amounts of fission products with cumulative high neutron absorption cross-sections are known to act as neutron sinks and therefore depress the neutron flux and power distribution in their vicinity. Fresh fuels, on the other hand, behave in the opposite manner. Differential fuelling takes advantage of the above phenomena by positioning fuel bundles with high burnup into high-power regions of the core, and conversely for the low-power regions of the core [1]. Based on the core geometry, the high power region of a typical CANDU reactor occupies the centre of its core, and it declines in power towards the peripheries. CANDU fuel engineers therefore selectively choose channels for refuelling at specific rates, such that on average over time, the central region predominantly contains highly irradiated fuels whereas the periphery regions contain more low-irradiated fuels [1]. It is possible in theory to flatten the radial power profile of the core using differential fuelling alone. In practice however, differential fuelling is used in conjunction with adjuster rods, because it is challenging to flatten the radial power profile of the core using fuelling alone.

4.1.3 Axial Fuelling Schemes (Modes of Refuelling)

Conventionally, eight or four bundle shift refuelling schemes have been used predominantly in CANDU reactors although ten-bundle shifts have also been used [1]. Eight-bundle shifts are normally used in channels that possess sufficient margins to their compliance limit [11]. The advantage of the eight-bundle shift is that it allows larger quantities of reactivity to be banked in a single fuelling event such that sufficient reactivity can be supplied with lower number of channel visits [11]. As the fuelling machines are limited in their capacity to fuel multiple channels in short periods of time, eight-bundle shift fuelling scheme is usually desired over the four-bundle shift [11].

However, the potential concern for 8-bundle shift is that it inserts fresh fuel bundles directly into the central region of the channel where the flux and the power are at maximum values along the axis. In addition to the fact a high number of fresh fuel bundles are being added to the channel, this can cause certain limiting channels to reach their OCPL when refuelling, which necessitates a temporarily derating in power [11]. Considering this disadvantage, the four-bundle shift is more desirable when refuelling limiting channels with tight operating margins [11]. The four-bundle shift inserts a lower number of fuel bundles into axial positions of lower flux and power, which results in a smaller transient power after refuelling. However, when many channels are refuelled with four-bundle shift, the necessary frequency of refuelling can become very high. High frequency of refuelling is disadvantageous because it exposes fuelling machines to a greater risk of mechanical failure resulting in outages where they become unavailable due to repairs and added maintenance [11]. This is of particular concern for older reactors whose margins to compliance power limits have decayed significantly. In such reactors, a large portion of the high-power, inner-core region may be required to be fuelled with four-bundle shift [11].

4.1.4 Flux and Power Distribution

The flux and power distributions in CANDU cores depend on their geometry, burnup of fuels, and locations of reactivity devices in the lattice [1]. Based on geometry alone, the flux and power are greatest at the radial and axial centre of the core, because it receives neutron flux from adjacent sources in all directions with the greatest magnitude [1]. Conversely, the periphery of the core experiences the lowest magnitudes of flux and power because it is adjacent to the core boundary allowing a larger proportion of the neutrons to leak out [1]. If there were no reactivity devices engaged, and the core was not differentially fuelled, the flux and power profile would peak towards the centre of the core. In reality, however, adjuster rods are inserted in higher quantities near the centre of the core, and the core is fuelled differentially, which produces a flattened flux and power distribution about the centre of the core [1]. Moreover, because reactivity devices possess larger reactivity worths in regions of higher thermal neutron flux, devices located closer

to the centre of the core exert greater flux suppression than ones in the adjacent regions [1]. The flux and power are therefore found to be highest in regions near the centre of the core, but away from the adjuster rods lined along the core centre [1]. Hence typical core flux or power distributions in CANDU reactors consist of an inward peak from the periphery to regions adjacent to the centre, then a plateau or slight depression towards the centre depending on the positions of adjuster rods. An example of a typical radial power distribution for a CANDU core is shown below in Figure 2.

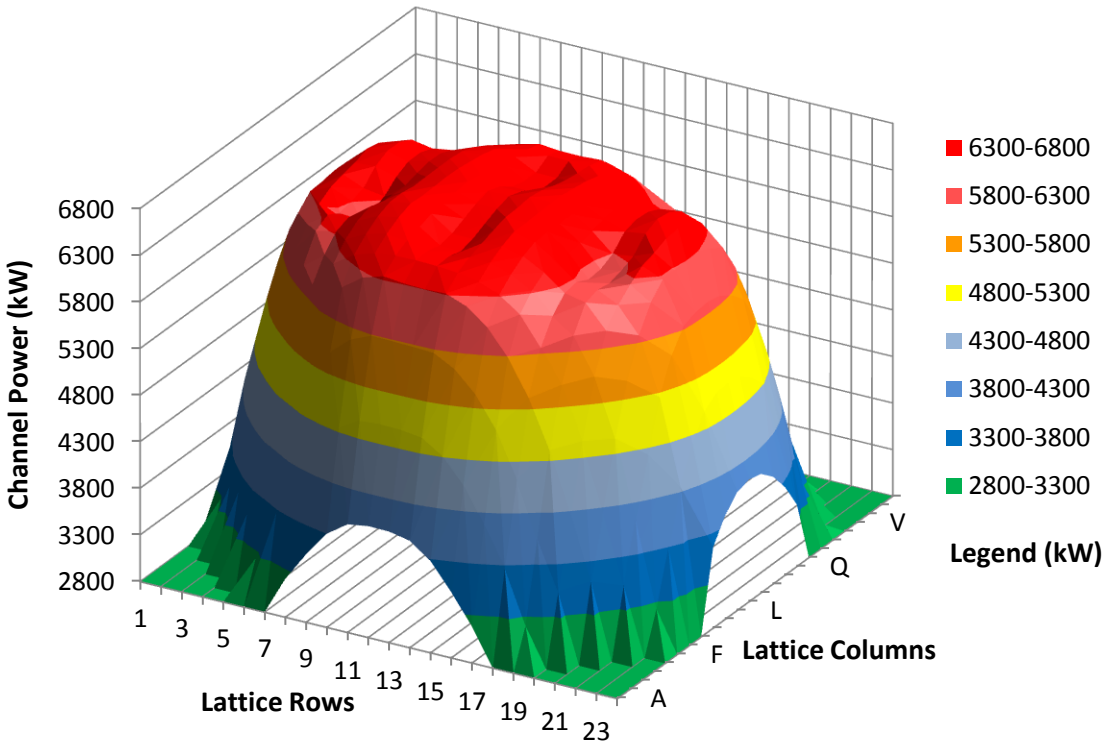


Figure 2. Time-averaged channel power distribution of the model CANDU core generated using RFSP

4.2 Compliance Power Limit

The operating power limits for channels and fuel bundles in CANDU reactors depend on a variety of interdependent factors. Fundamentally, the design capacity of the reactor's PHTS, in association with the thermal flux of its core, determines the critical powers at which the onset of dryout occurs in each fuel channel. However, as channels age, their margins to dryout decrease, and they become unable to maintain the same margin of safety when operating at their initial design capacities. Continued safety

analyses are therefore used to determine sufficiently tighter operating envelopes (OCPLs and OBPLs) that provide the same margin of safety. The values of licensed OCPLs and OBPLs vary from reactor to reactor, and also depend on the age of the reactor. At all times, they must be accepted by the regulating body holding the authority for licensing.

4.3 BNAF Design

Following the recommendations of Paquette *et al* (2014) [1], this study investigates BNAFs containing up to 150 mg of Gd_2O_3 and 300 mg of Eu_2O_3. The two absorbers applied up to the above quantities are known to effectively eliminate the refuelling transient and reduce the plutonium peak while incurring minimal losses in the discharge burnup [2]. The design of the BNAF is based on the conventional 37-element NU fuel, but it incorporates BNAs homogeneously into the CANLUB layer [2]. This approach was adopted because it avoids major modifications to the current fuel design and its manufacturing processes, and also minimizes the impact on UO_2 thermochemistry. The method of incorporating the BNAs into the fuel is a part of an ongoing optimization study, because the effectiveness of the BNAs may vary depending on how they are incorporated into the fuel. In the present study, however, the absorbers are assumed to be evenly distributed in the CANLUB layer. Figure 3 below illustrates the cross-section view of a typical CANDU lattice (for WIMS calculations) containing a 37-element NU fuel bundle.

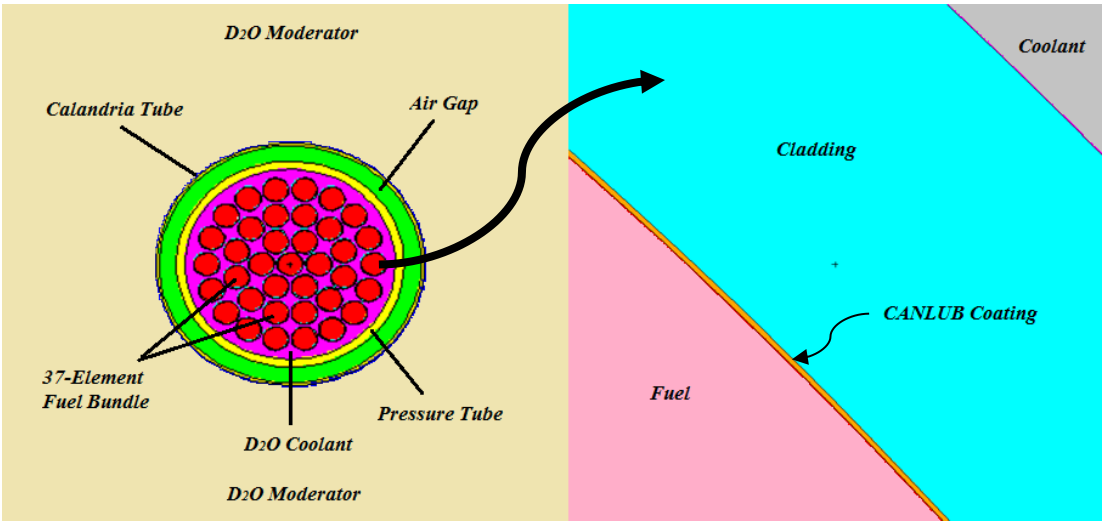


Figure 3. Cross-section view of a CANDU Nuclear Reactor fuel channel (lattice cell) model (Paquette *et al*., 2014) [12].

Chapter 5: Methods

The methods used in this study incorporate and utilize suitable industry standard toolset (IST) codes to conduct a generic fuelling study, using BNAFs containing varied quantities of added absorbers, for a CANDU reactor. The involved IST codes include the Reactor Fuelling Simulation Program (RFSP) [13], [14], the WIMS-AECL code [15], and the DRAGON-IST code [16]. To incorporate each of the IST codes into a single platform and automate the majority of the simulation process, the MATLAB^{TM} [17] programming environment and language were used to create several automation programs/scripts. To generate visualized diagrams of instantaneous core snapshot data, the Microsoft EXCEL^{TM} [18] program was used

5.1 IST Codes

5.1.1 RFSP-IST

RFSP is a multi-modular IST code used as the standard design and safety modelling tool in the CANDU industry [13]. RFSP is the main tool used for this study because it incorporates the fuel cell multi-group macroscopic cross-sections and diffusion coefficients generated by WIMS-AECL, and incremental cross-sections for reactivity devices generated by DRAGON-IST, to perform multitudes of neutron transport calculations for CANDU cores of specified geometries. In particular, RFSP is capable of simulating the effects of the local power transients caused by refuelling ripples. The effects of the added absorbers in reducing the magnitudes of the refuelling transients can therefore be captured via simulations using RFSP. Three RFSP modules, TIME-AVER, ISNTANTAN and SIMULATE, are used in this study to determine the time-averaged performance and generate core-following data for cores fuelled respectively with NU fuel and BNAFs containing different quantities of added absorbers. RFSP has been tested and validated against benchmark power-reactor measurements, and is known to give reasonable results when compared with site measurement data [14]. The version REL_3-04 of RFSP is used throughout the study.

5.1.2 WIMS-AECL

WIMS-AECL [15] is a two-dimensional multi-group neutron transport code used routinely for CANDU reactor lattice calculations. In the first portion of this study, the WIMS-AECL code is used to conduct burnup calculations for the BNAF at the lattice level to determine the impact of adding neutron absorbers at the fuel level (single lattice). In order to accurately capture the impact of all the isotopes of gadolinium and europium, the E70ACR IST library, which includes the most up-to-date properties of all the relevant gadolinium and europium isotopes, was used for the lattice simulations. The benchmarking of WIMS lattice calculations for fuels containing Gd_2O_3 and Eu_2O_3 were conducted by Paquette *et al.* (2014) using the Monte Carlo N-Particle transport code version 6 (MCNP6) [19], which demonstrated reasonable agreement between results [2]. In the second portion of this study, WIMS-AECL is used to generate fuel tables containing cell-averaged cross-sections and diffusion coefficients to be read by RFSP for full core three-dimension calculations. To generate the fuel tables in a format suitable for reading in RFSP, the WIMS-UTIL software package was used to post-process the TAPE 16 records (written in Fortran unformatted binary writes) produced by WIMS-AECL into an American Standard Code for Information Interchange (ASCII) format accepted by RFSP. WIMS version 3.1 (with E70ACR IST library) and WIMS-UTIL version 2.0.3 are used throughout the study.

5.1.3 DRAGON-IST

DRAGON-IST [16] is a three-dimensional multi-group neutron transport code used by the Canadian nuclear industry to conduct 3-D supercell transport calculations. It is capable of utilizing collision probability methods to solve neutron transport equations in various spatial regions and multiple neutron energy groups. This method is theoretically rigorous and the results are consistent with WIMS-IST cell calculations. DRAGON-IST can be used for various different applications, but in this study, it is used to calculate the incremental cross-sections of reactivity devices for the CANDU model used in the study.

5.2 Developed MATLAB Programs

5.2.1 Automatic WIMS Input and Output Generator

The automatic WIMS Input and Output Generator (AWIMSIOG) was created by the principle investigator of this study in order to quickly generate desired WIMS input and output files without the repetitive manual use of the WIMS code. The use of AWIMSIOG in this study is for the purpose of efficiently generating WIMS outputs and observing the impact of using different WIMS data libraries for simulations, or incorporating different quantities of neutron absorbers into the BNAF. Accordingly, AWIMSIOG was used to optimize the combination and the quantity of burnable absorbers to be added to the BNAFs. Details of the AWIMSIOG are outlined in Appendix A

5.2.2 Automatic Refuelling and Core Monitoring Program

The automatic Refuelling and Core Monitoring Program (ARCMP) was created by the principle investigator of this study in order to automatically produce generic refuelling and core-following simulation data via RFSP. By automating the process, the generated data are consistent with the programmed method and are not subject to human bias, which is favorable for comparing the performances of the BNAFs to that of the reference NU fuel. The ARCMP provides an easy-access platform incorporating several automated features for RFSP to simplify the process of generating simulation data. The core monitoring function of ARCMP is a set of simple scripts used to text-search RFSP outputs and archive its contents into specifically structured data matrices. The imported data are refined using simple scripts to calculate additional core properties of interest which are not output directly by RFSP. The refined data are automatically archived, and can be recalled by user command in MATLAB via a simple script designed to read the stored data. Additionally, a script designed to manipulate Microsoft Excel can be recalled by user command in MATLAB to print and save the refined data into excel spread sheets with conditional formatting.

The automatic refuelling function of ARCMP is a more complex set of scripts incorporating the principles of fuel management in CANDU reactors to create generic refuelling histories to be used in core-following simulations at the refuelling equilibrium. The generic refuelling histories are created by running the ARCMP from an instantaneous steady state core as the starting point. The ARCMP uses its core monitoring function to obtain instantaneous snapshot data of the core at an interval of every 0.25 FPDs during core following. The instantaneous core snapshot data are analyzed at each interval using the refuelling algorithm to decide if refuelling is necessary, and if so, which channel should best be refuelled to maintain the desired reference flux and power distributions in the core. The interval of 0.25 FPDs was chosen because it is close to the average rate of refuelling required to keep the model core critical using the NU fuel. The ARCMP is therefore capable of ensuring that the model core is consistently kept critical by inquiring the necessity for refuelling at the interval of 0.25 FPDs. The refuelling decisions that are made are logged and stored in simple data matrices as the model core advances forward in reactor time during the core-following simulations. This log is effectively used as the generic refuelling history. The details of the ARCMP are outlined in Appendix B.

In this study, the generic refuelling history is created by running a core-following simulation using the ARCMP for the regular NU fuel. The regular NU fuel is used because it is the reference fuel that the performances of the BNAFs are to be compared with. The generic refuelling history is generated with all LZCs fixed at same positions, such that the flux suppression provided by the LZCs is both constant over time and level across core as refuelling decisions are made. This allows the refuelling algorithm to attempt to flatten the distribution of flux and power in the core via refuelling alone, which is in accordance with the current practice of fuel management for CANDU reactors. When the generic refuelling history is used to conduct the actual core-following simulations, the LZCs are allowed to move for bulk and spatial reactivity control as it best emulates the reality of operating the model core. The starting point of all core-following simulations conducted in the study is from a generic refuelling equilibrium fuelling state. The generic core state is loaded entirely with NU fuels; thus when simulating core-following using the BNAFs, a transition from the initially NU-loaded core is simulated.

Transitioning from the initially identical core state using identical refuelling histories is used as the method of core-following in this study because it ensures all parameters affecting the behavior of the core other than the quantity of added absorbers within fuels are kept constant between simulations.

5.3 Reactor Model

A reference data set (RFSP model) of a CANDU reactor model used in nuclear generating stations (NGS) is used throughout the conduct of the study. The model has 480 fuel channels which fit 13 fuel bundles within each channel. The total core thermal power is 2650 MW_{th}. The reference data set includes incremental cross-section data for all of its reactivity devices, calculated using DRAGON-IST.

5.4 Fuel Models

The fuel models for both BNAF and NU fuel were created in WIMS using the geometries of the conventional 37-element NU fuel outlined previously in Figure 3. The AWIMSIOG was then used in conjunction with WIMS to vary the burnable neutron absorber content in the CANLUB layer and conduct lattice burnup calculations. WIMS-UTIL was then used to generate the fuel tables for each of the BNAFs and the NU fuel.

5.5 Experimental Design

The methods used in the course of this study are developed for the purpose of conducting a generic fuelling study for a CANDU reactor fuelled with BNAFs. To achieve this, the study is divided into five phases. The first phase involves lattice burnup calculations using WIMS to determine the behaviors of the BNAFs and NU fuels at the lattice cell level (a single fuel bundle). The results of the lattice burnup calculations are used to generate fuel tables to be read by RFSP. The second phase involves time-averaged core calculation using RFSP to determine the behavior of the core that is averaged over time when using the BNAFs or NU fuel. The third phase involves the

generation of a patterned age to create an instantaneous snapshot data set of a generic steady-state core. The generic steady-state core is to be used as the starting point for all refuelling and core-following simulations. The fourth phase involves simple refuelling simulations conducted in order to determine the trend in power transients that occurs in the vicinity of fresh fuel (NU or BNAFs). The fifth phase involves core-following using the automatic refuelling program to generate a generic fuelling history over 400 FPDs for the NU fuel, then repeat the core-following using the same history for BNAFs. The flow diagram shown in Figure 4 describes in detail the use of each IST code and developed MATLAB programs in accomplishing each phase of the experiment.

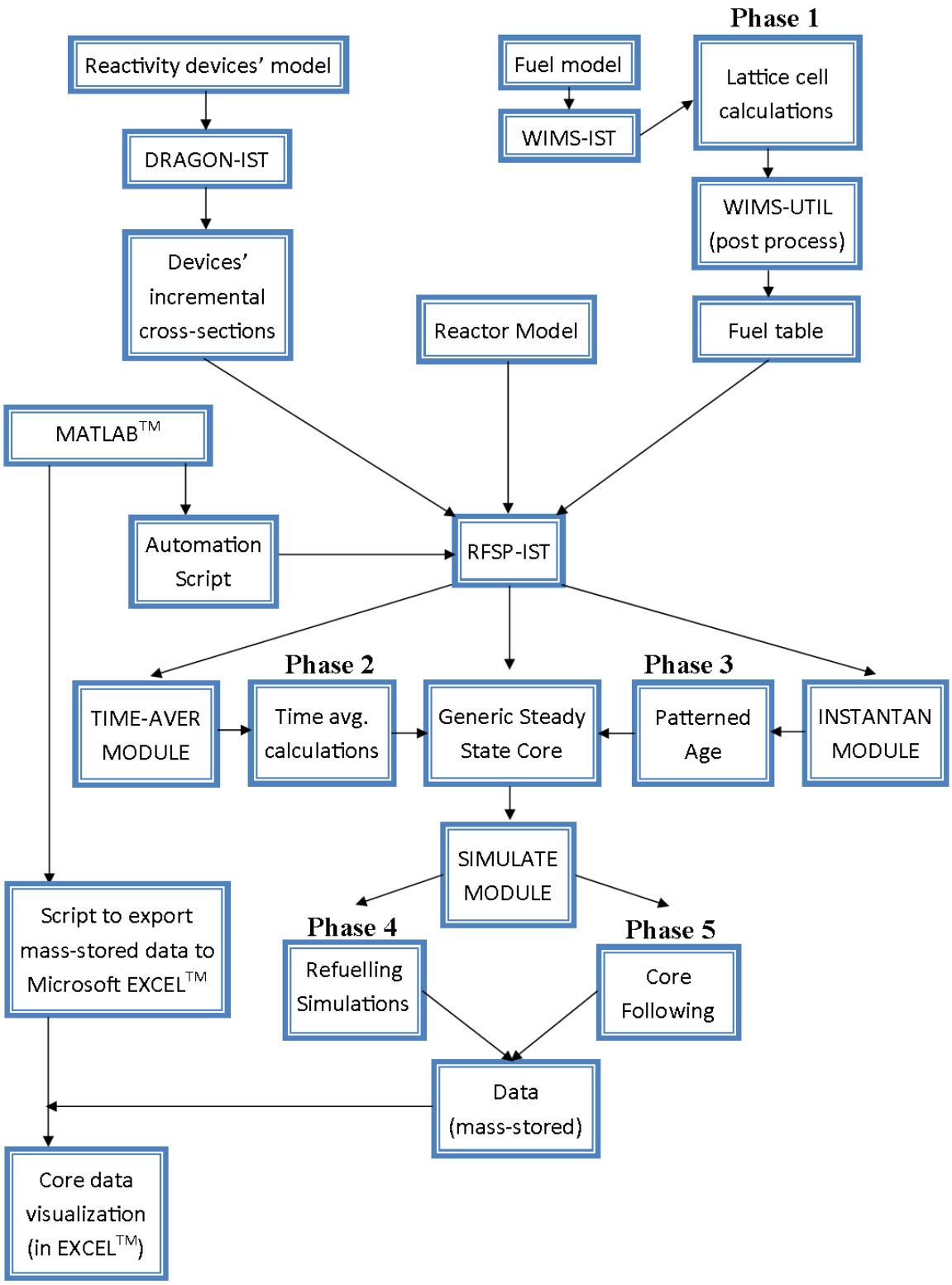


Figure 4. Flow diagram entailing the role of each IST code and other utilities used systematically in the overall method of the research

5.5.1 Phase 1: Lattice Cell Calculation

To determine the effects of added absorbers at the lattice level and generate fuel tables to be read by RFSP, lattice cell calculations are conducted. A stock template of a WIMS input file containing the geometries and the material designation of the current 37-element fuel design is first created. The stock template is then manipulated using AWIMSIOG to conduct lattice burnup calculations for the NU fuel and BNAFs containing user-specified quantities of added Gd_2O_3 and Eu_2O_3 (in CANLUB). Fuel tables are produced at the ends of calculations by the AWIMSIOG. Paquette *et al.* (2014) recommends inclusion of ~120mg (~5ppm in mass fraction of fuel) of Gd_2O_3 and ~300mg (12.5ppm) of Eu_2O_3 to achieve the most ideal shape of the reactivity curve with negligible losses in burnup [2]. In this study, a parametric optimization similar to Paquette *et al.* (2014) was conducted using various combinations of the two absorbers. Figures 5 to 7 below present the results for some of the combinations that were investigated, as well as the reference case for which no poisons (absorbers) were added. The resulting, optimal combination of the two absorbers is 150 mg of Gd_2O_3 and 300 mg of Eu_2O_3, which is similar to the findings of Paquette *et al.* (2014).

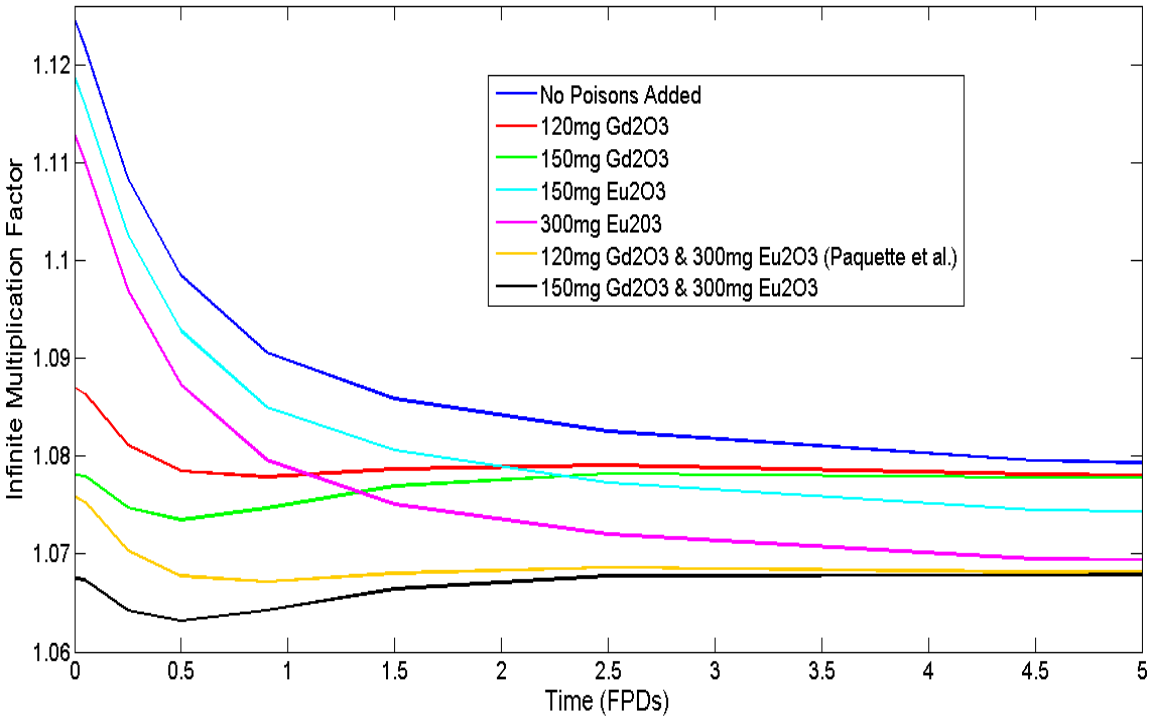


Figure 5. k_{inf} versus time (0-5 FPD) of the model fuel lattice containing varied quantities of Gd_2O_3 and Eu_2O_3

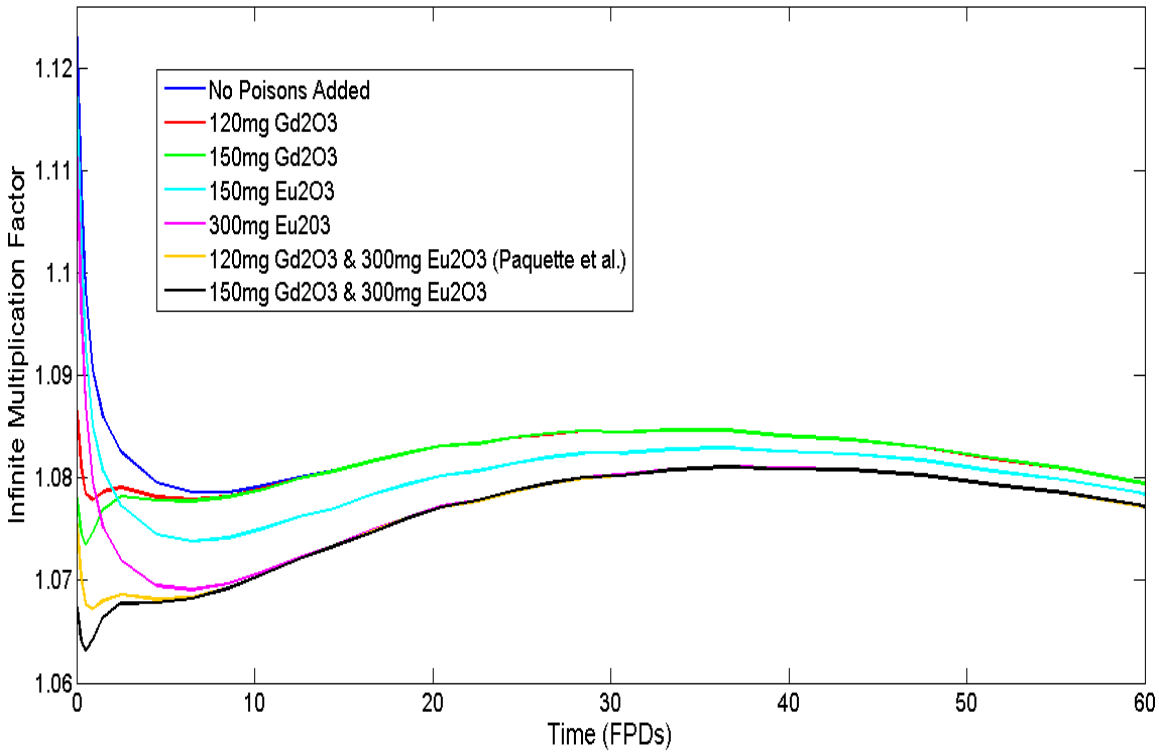


Figure 6. k_{inf} versus. time (0-60 FPD) of the model fuel lattice containing varied quantities of Gd_2O_3 and Eu_2O_3

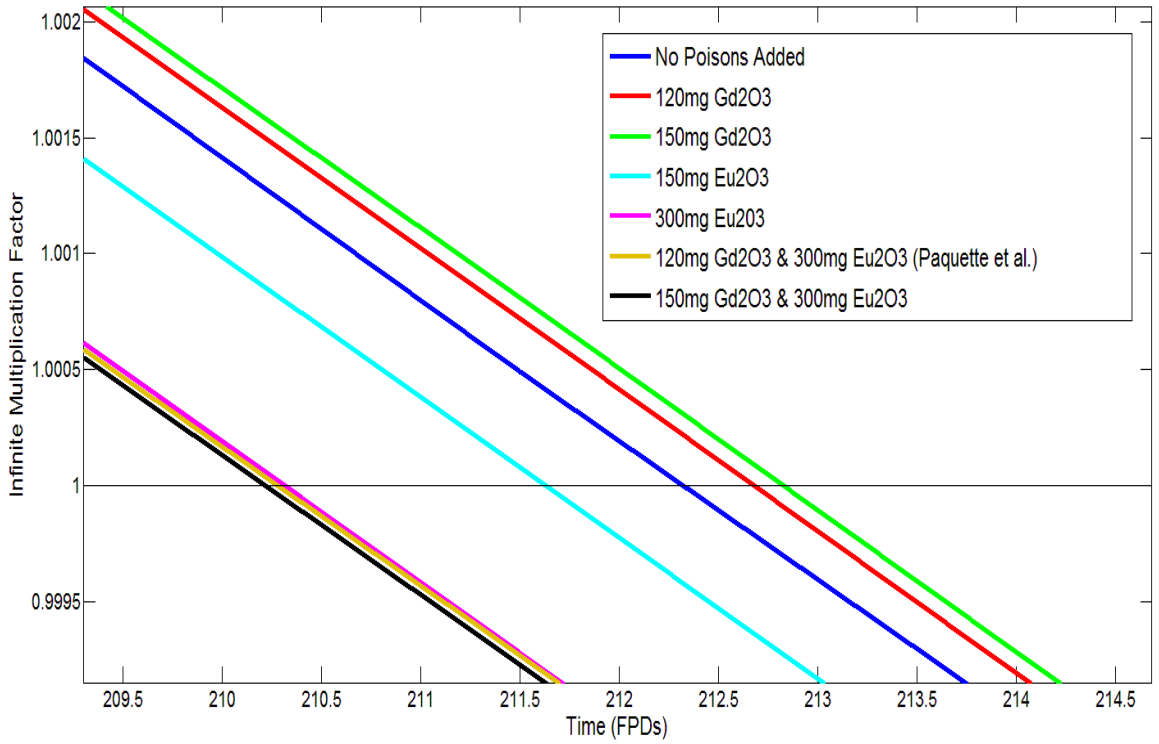


Figure 7. k_{inf} versus. time (209-215FPD) of the model fuel lattice containing varied quantities of Gd_2O_3 and Eu_2O_3

5.5.2 Phase 2: Time-average Core Calculation

A time-average calculation of the reactor core plays an important part in the design and modelling of CANDU reactors. This is because the time-average calculation can be used to establish the necessary target values for the average-over-time irradiations of fuel channels in the core required to achieve the desired average-over-time distribution of core power and the effective neutron multiplication factor (k_{eff}). This can be done for any specified time-average values and positions of reactivity devices in the core. Typically, reasonable positions and values of reactivity devices that are expected as an average over time during an extensive period of core-following are used as the time-average values. A time-average calculation utilizes these values along with the specified time-average irradiation values of fuel channels to determine the time-average values of k_{eff}, burnup, power and flux distribution. It should be noted that time-average calculations do not account for the phenomena of transient powers associated with refuelling because it does not capture the discrete instances of refuelling. This is because time-average calculations are meant to only capture an "average-over-time" picture of the core. For this reason, the time-average core does not account for the effects of refuelling ripples. However, the time-average core provides a good approximation of the target irradiations to which fuel channels should be operated at to achieve the ideal core power distribution.

In this study, the time-average calculation is used exclusively to generate the values of the beginning-of-cycle and the end-of-cycle irradiation values of all fuel bundles in the model core. The two calculated values can then be used by the INSTANTAN module in RFSP to generate an instantaneous snapshot of the core state that is reflective of a patterned distribution of channel "age" that is tailored to the specifications made by the user. The channel "age" used by the INSTANTAN module refers to the time which has passed since a channel has been last refuelled. Based on its "age," a fuel channel is assigned irradiation values between its beginning-of-cycle and the end-of-cycle irradiation values.

The time-average calculation conducted in this study is used to determine the beginning-of-cycle and the end-of-cycle irradiation data for the model core, at the refuelling equilibrium, for refuelling with the reference NU fuel. This is consistent with the overall methodology of the study, which seeks to track and compare the behaviours of the reference core when it is transitioned from being refuelled with reference NU fuel to BNAFs containing different quantities of added absorbers. As such, the starting core state consists entirely of the reference NU fuel. Figures 8 to 10 below respectively outline the reference target irradiations of fuel channels, the resulting time-average core power distribution, and the regionalized resulting time-average core power distribution.

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24
A								2.83	2.83	2.83	2.83	2.83	2.83	2.83	2.83	2.83	2.83							
B						2.93	2.94	2.83	2.76	2.76	2.76	2.76	2.76	2.76	2.76	2.76	2.83	2.99	2.93					
C					2.93	2.93	2.93	2.60	2.60	2.76	2.82	2.82	2.82	2.82	2.82	2.76	2.82	2.93	2.93	2.93				
D				2.93	2.93	2.70	2.90	2.90	3.22	3.39	3.39	3.22	3.22	3.37	3.22	3.22	2.90	2.90	2.70	2.93	2.93			
E			3.03	2.93	2.93	2.70	2.90	3.39	3.39	3.39	3.39	2.68	2.68	3.37	3.22	3.22	2.90	2.90	2.70	2.93	2.93	3.03		
F		3.00	3.00	2.70	2.70	2.70	2.90	2.90	3.39	3.39	3.39	2.68	2.68	3.37	3.26	3.22	2.90	2.90	2.70	2.70	2.67	3.03	3.03	
G		3.00	3.00	2.70	2.70	2.70	2.90	2.90	3.22	3.39	3.22	3.83	3.83	3.22	3.22	3.22	2.90	2.90	2.70	2.70	2.67	3.03	3.03	
H		3.00	3.00	2.70	3.19	2.89	2.89	2.89	3.30	3.30	3.30	3.30	3.30	3.30	3.30	3.30	3.28	3.11	3.03	3.03	2.67	2.70	3.03	
J	3.03	3.00	2.97	2.59	3.48	3.35	3.35	3.35	3.30	3.30	3.30	2.93	2.93	3.30	3.30	3.30	3.28	3.39	3.48	3.03	2.67	2.70	2.70	3.03
K	3.03	2.74	2.74	2.74	3.48	3.35	3.35	3.35	3.30	3.30	3.30	3.30	3.30	3.30	3.30	3.30	3.48	3.48	3.73	3.48	2.70	2.70	2.70	3.03
L	2.92	3.23	3.23	3.23	3.48	3.01	3.01	3.35	3.30	3.30	3.30	3.30	3.30	3.30	3.30	3.30	3.03	3.03	3.03	3.48	2.67	3.23	3.23	3.23
M	2.92	3.27	3.27	3.35	3.53	3.73	3.35	3.41	3.45	3.45	3.45	3.45	3.45	3.45	3.45	3.45	3.43	3.43	3.73	3.52	3.52	3.23	3.23	2.95
N	2.92	3.11	3.11	3.35	3.77	3.43	3.43	3.43	3.49	3.49	3.49	3.45	3.45	3.49	3.49	3.45	3.43	3.43	3.68	3.68	3.52	3.23	3.23	2.95
O	2.92	3.23	3.23	3.35	3.43	3.55	3.43	3.43	3.49	3.49	3.49	3.45	3.45	3.49	3.49	3.45	3.43	3.43	3.43	3.43	3.65	3.23	3.23	2.95
P	2.85	2.77	2.77	3.11	3.72	3.43	3.43	3.38	3.45	3.45	3.45	3.45	3.45	3.45	3.45	3.45	3.43	3.43	3.43	3.43	2.77	2.77	2.77	2.85
Q	2.85	2.88	2.88	2.77	2.80	2.80	2.80	3.43	3.37	3.74	3.74	3.74	3.74	3.74	3.74	3.45	3.43	3.09	3.09	3.06	2.77	2.77	2.77	2.85
R		2.88	3.35	2.79	3.45	3.45	3.45	3.42	3.24	3.29	3.38	3.38	3.38	3.38	3.29	3.29	3.45	3.45	3.45	3.22	2.77	2.91	2.91	
S		2.88	2.88	2.80	2.80	3.45	3.42	3.42	3.06	3.29	3.32	3.32	3.32	3.32	3.29	3.29	2.94	2.94	3.45	2.78	2.77	2.91	2.91	
T		2.88	2.88	2.77	2.77	2.77	3.05	3.05	2.83	3.29	3.39	3.39	3.39	3.39	3.29	3.29	2.94	2.94	2.77	2.78	2.77	2.91	2.91	
U			2.85	2.77	2.77	2.77	2.77	2.77	2.77	2.77	3.18	3.18	2.98	2.98	2.98	2.78	2.78	2.98	2.98	2.77	2.77	2.85		
V				2.77	2.77	2.72	2.75	2.67	2.75	2.90	3.29	3.38	3.38	2.98	2.98	2.44	2.57	2.57	2.57	2.77	2.77			
W					2.77	2.57	2.80	2.80	2.37	2.67	2.67	2.90	2.90	2.67	2.67	2.44	2.78	2.78	2.77	2.77				
X						2.41	2.69	2.69	2.69	2.70	2.70	2.92	2.92	2.70	2.70	2.69	2.69	2.69	2.48					
Y								2.22	2.37	2.37	2.37	2.52	2.52	2.37	2.37	2.37	2.29							

Colour Format:	4.0 n kb^{-1}	3.0 n kb^{-1}	2.0 n kb^{-1}

Figure 8. Time-average channel target irradiations (in n kb^{-1}) required by the model core to achieve its reference power distribution and k_{eff} using the 37-element NU fuel

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24
A								3397	3717	3996	4153	4190	4192	4158	4001	3720	3399							
B						3438	4000	4437	4797	5038	5165	5184	5187	5174	5048	4802	4439	3999	3442					
C					3794	4317	4869	5373	5676	5833	5913	5912	5919	5931	5846	5670	5356	4879	4325	3797				
D				3887	4575	5215	5652	5996	6126	6195	6242	6253	6264	6278	6271	6174	6037	5679	5228	4579	3883			
E			3766	4580	5303	5836	6099	6221	6267	6292	6293	6346	6359	6336	6382	6365	6360	6135	5848	5304	4573	3750		
F		3456	4406	5278	5909	6262	6332	6429	6373	6346	6285	6242	6254	6323	6421	6455	6469	6347	6263	5905	5272	4382	3415	
G		4025	5003	5790	6277	6486	6443	6499	6464	6386	6300	5989	5997	6323	6443	6480	6490	6421	6465	6267	5786	4986	3986	
H		4573	5480	6139	6361	6475	6427	6497	6469	6426	6311	6084	6087	6317	6426	6443	6379	6330	6408	6373	6141	5526	4545	
J	3857	4894	5803	6349	6337	6246	6165	6320	6429	6431	6355	6233	6232	6349	6415	6394	6274	6090	6169	6385	6324	5840	4911	3800
K	4099	5177	6025	6424	6358	6206	6101	6265	6393	6420	6381	6281	6278	6370	6397	6353	6185	6000	6064	6295	6381	5989	5129	4022
L	4281	5226	6005	6358	6371	6292	6175	6245	6350	6376	6350	6275	6271	6337	6352	6317	6249	6103	6203	6286	6275	5923	5138	4144
M	4319	5233	5985	6315	6358	6250	6191	6215	6260	6259	6216	6126	6121	6201	6234	6223	6159	6104	6160	6262	6181	5885	5133	4206
N	4284	5221	5975	6284	6313	6321	6201	6190	6197	6169	6093	5955	5950	6078	6143	6166	6135	6127	6195	6227	6143	5842	5095	4174
O	4162	5111	5885	6235	6336	6291	6186	6164	6157	6111	6011	5818	5813	5994	6084	6123	6109	6120	6236	6258	6096	5789	5017	4066
P	3961	5003	5821	6171	6215	6246	6132	6123	6119	6075	5978	5789	5783	5961	6046	6075	6058	6068	6185	6212	6178	5766	4936	3887
Q	3685	4676	5524	6013	6197	6194	6096	6023	6076	6005	5949	5849	5844	5933	5976	6023	5972	5988	6083	6106	5980	5512	4645	3621
R		4315	5066	5670	5796	5768	5724	5889	6054	6087	6056	5994	5989	6041	6061	6015	5863	5704	5745	5813	5662	5118	4278	
S		3762	4627	5242	5569	5505	5522	5788	6042	6074	6061	6002	5999	6050	6050	5983	5864	5607	5507	5564	5234	4606	3720	
T		3207	4043	4732	5184	5345	5451	5795	6043	6042	5999	5912	5915	5998	6022	5951	5813	5479	5344	5173	4716	4015	3164	
U			3450	4121	4691	5042	5388	5796	5993	6060	5945	5790	5830	5990	6027	5985	5798	5362	5009	4679	4106	3428		
V				3481	4081	4642	5137	5584	5836	5910	5797	5600	5614	5869	5921	5903	5621	5175	4662	4076	3470			
W					3370	3930	4480	4969	5364	5492	5497	5329	5336	5513	5507	5369	4983	4487	3906	3362				
X						3024	3577	4052	4425	4653	4740	4692	4696	4749	4662	4432	4057	3577	3015					
Y								3060	3386	3668	3823	3851	3853	3828	3673	3388	3054							

Colour Format:	7000 kW	5000 kW	3000 kW

Figure 9. The reference (time-average) channel power distribution of the model core using 37-element NU fuel (in kW). The value of k_{eff} is 1.000, max channel power is 6528 kW at channel G-6, max bundle power is 788 kW at channel G-6, bundle #6, and the radial form factor is 1.18

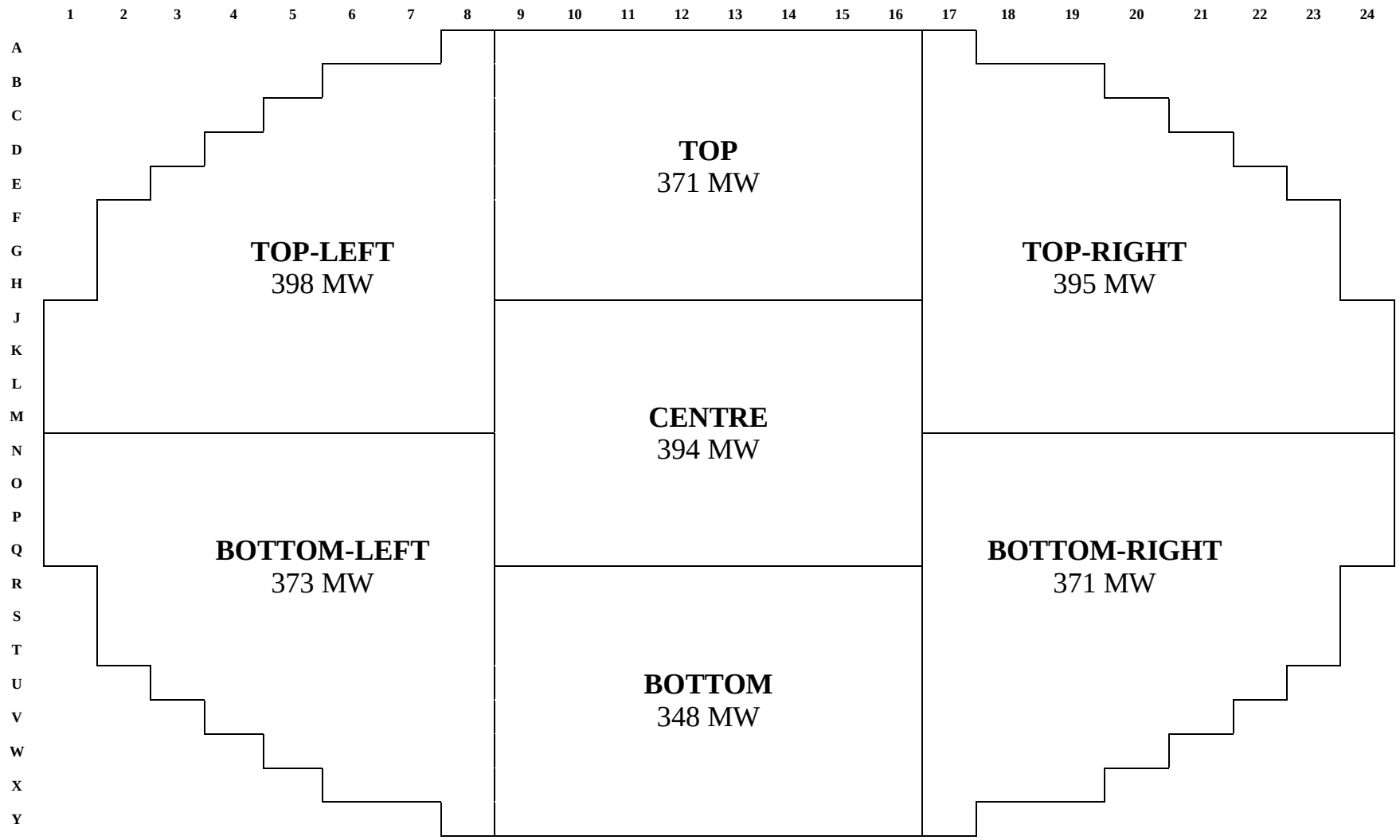


Figure 10. Region-averaged (time-average) power distribution of the model core using 37-element NU fuel (in MW). By the design of the core, there is approximately 5% top to bottom tilt in power

5.5.3 Phase 3: Generation of Equilibrium Refuelling Core Data

A patterned distribution of "age" for each fuel channel in the model core was developed to generate an instantaneous snapshot of the core at the refuelling equilibrium. The "ages" of fuel channels described here define the fluence and the burnup of each individual fuel bundle contained within each channel by using the beginning-of-cycle and end-of-cycle irradiation data obtained using the time-average calculation. The "age" is assigned between zero and one, with zero describing the state of the channel immediately after refuelling (beginning-of-cycle), and one describing the channel at its end-of-cycle irradiation. Typically, to generate an instantaneous snapshot of a core at the refuelling equilibrium state, an interlocking, "checkerboard" type of distribution of high and low channel "ages" is used. The magnitudes of the distributed channel "ages" are chosen specifically to yield an overall rate of reactivity decline in the core that best resembles the known rate of refuelling required for the equilibrium refuelling state. The interlocking pattern is used because in practice, adjacent channels are never refuelled consecutively as it would result in a high power density in their vicinity. The interlocking pattern therefore best represents the distribution of irradiation found in cores at their equilibrium refuelling states. In this study, the channel "age" distribution was developed to yield an equilibrium core with a power distribution best resembling the reference power distribution of the model core. Figure 11 below presents the patterned "age" that is used for the generation of the instantaneous core snapshot. Figures 12 to 14 below respectively illustrate the resulting channel power distribution, the channel power peaking factors (CPPFs) of each channel, and the regional distribution of channel powers.

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24
A								0.20	0.56	0.12	0.58	0.38	0.65	0.23	0.54	0.25	0.45							
B						0.28	0.82	0.16	0.42	0.64	0.86	0.38	0.56	0.22	0.86	0.42	0.08	0.50	0.20					
C					0.20	0.94	0.12	0.70	0.08	0.18	0.32	0.92	0.10	0.68	0.02	0.64	0.18	0.52	0.60	0.20				
D				0.78	0.74	0.40	0.62	0.36	0.90	0.52	0.78	0.50	0.80	0.26	0.48	0.86	0.32	0.78	0.74	0.50	0.40			
E			0.92	0.50	0.98	0.60	0.34	0.72	0.58	0.20	0.74	0.98	0.24	0.66	0.84	0.68	0.92	0.50	0.98	0.60	0.30	0.50		
F		0.00	0.10	0.80	0.24	0.46	0.88	0.06	0.28	0.94	0.40	0.60	0.42	0.74	0.40	0.96	0.50	0.90	0.64	0.46	0.90	0.50	0.90	
G		0.12	0.48	0.24	0.50	0.28	0.50	0.34	0.82	0.22	0.62	0.20	0.35	0.25	0.77	0.42	0.80	0.35	0.70	0.04	0.14	0.60	0.30	
H		0.96	0.02	0.48	0.74	0.30	0.76	0.54	0.16	0.70	0.36	0.72	0.06	0.44	0.50	0.80	0.02	0.43	0.60	0.37	0.76	0.54	0.30	
J	0.16	0.42	0.64	0.86	0.38	0.56	0.22	0.96	0.42	0.28	0.90	0.58	0.28	0.82	0.30	0.45	0.60	0.55	0.30	0.65	0.32	0.96	0.40	0.40
K	0.60	0.08	0.18	0.32	0.92	0.10	0.68	0.02	0.70	0.38	0.52	0.20	0.94	0.19	0.70	0.35	0.18	0.25	0.68	0.40	0.64	0.02	0.80	0.50
L	0.36	0.80	0.48	0.74	0.50	0.60	0.26	0.48	0.36	0.72	0.40	0.60	0.50	0.68	0.40	0.62	0.45	0.60	0.40	0.70	0.40	0.45	0.20	0.32
M	0.72	0.58	0.20	0.74	0.78	0.24	0.56	0.70	0.38	0.80	0.50	0.70	0.60	0.34	0.70	0.54	0.20	0.72	0.83	0.24	0.63	0.84	0.40	0.72
N	0.36	0.90	0.52	0.78	0.50	0.80	0.26	0.48	0.86	0.32	0.78	0.50	0.40	0.62	0.36	0.78	0.50	0.60	0.50	0.76	0.26	0.48	0.95	0.53
O	0.72	0.58	0.20	0.74	0.98	0.24	0.66	0.84	0.38	0.62	0.30	0.68	0.50	0.30	0.72	0.44	0.70	0.50	0.78	0.28	0.66	0.84	0.30	0.70
P	0.06	0.28	0.94	0.40	0.60	0.46	0.04	0.30	0.56	0.10	0.80	0.24	0.46	0.88	0.16	0.48	0.84	0.40	0.60	0.46	0.04	0.30	0.80	0.50
Q	0.44	0.82	0.12	0.65	0.48	0.88	0.14	0.76	0.22	0.68	0.26	0.66	0.14	0.24	0.44	0.82	0.12	0.62	0.34	0.88	0.14	0.76	0.04	0.45
R		0.16	0.70	0.36	0.72	0.06	0.44	0.54	0.96	0.02	0.48	0.70	0.30	0.76	0.54	0.16	0.70	0.36	0.72	0.06	0.44	0.60	0.40	
S		0.42	0.08	0.90	0.58	0.28	0.82	0.16	0.42	0.64	0.86	0.38	0.56	0.22	0.96	0.42	0.08	0.90	0.58	0.28	0.82	0.40	0.70	
T		0.64	0.18	0.52	0.20	0.94	0.12	0.70	0.58	0.18	0.32	0.92	0.10	0.68	0.32	0.64	0.18	0.52	0.20	0.94	0.12	0.10	0.80	
U			0.32	0.78	0.74	0.40	0.62	0.36	0.90	0.52	0.78	0.50	0.80	0.26	0.48	0.86	0.32	0.78	0.74	0.40	0.80	0.40		
V				0.50	0.98	0.60	0.34	0.72	0.58	0.20	0.74	0.98	0.24	0.66	0.84	0.38	0.92	0.50	0.98	0.50	0.30			
W					0.24	0.46	0.88	0.06	0.28	0.94	0.40	0.60	0.46	0.12	0.30	0.56	0.10	0.80	0.40	0.70				
X						0.04	0.14	0.44	0.82	0.12	0.62	0.34	0.88	0.14	0.76	0.22	0.68	0.30	0.80					
Y								0.08	0.01	0.45	0.30	0.62	0.13	0.45	0.18	0.23	0.03							

Colour Format:	1.0 (unitless)	0.5 (unitless)	0.0 (unitless)

Figure 11. The patterned-age used to generate an instantaneous snapshot of the model core at the equilibrium refuelling state. Value of 0 indicates the channel is fresh fuelled and the value of 1.0 indicates the channel has reached its time-average end-of-cycle irradiation

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24
A								3829	3989	4413	4307	4449	4282	4526	4204	4127	3730							
B						3629	3984	4903	5177	5176	5185	5324	5313	5391	4958	5104	4879	4258	3809					
C					3886	4047	5175	5536	5975	6072	6018	5777	6053	5944	6013	5712	5561	4984	4387	4127				
D				3548	4238	5150	5648	6099	5986	6170	6032	6131	6056	6290	6176	5869	5986	5485	5113	4656	4087			
E			3496	4457	4785	5715	6130	6131	6211	6340	6055	6048	6308	6098	5987	5994	5932	5916	5543	5225	4791	3829		
F		3846	4740	5214	5942	6254	6183	6557	6438	6066	6214	6125	6192	6040	6192	5942	6174	5969	6082	5909	5198	4426	3216	
G		4446	5251	5975	6336	6591	6450	6566	6292	6409	6175	6037	6002	6286	6155	6259	6178	6324	6337	6421	5945	4981	4221	
H		4343	5748	6229	6267	6550	6318	6453	6539	6264	6258	5913	6150	6245	6268	6154	6422	6307	6348	6451	6113	5590	4873	
J	4190	5012	5800	6205	6319	6192	6204	6035	6370	6402	6017	6064	6166	6069	6353	6310	6169	6024	6211	6324	6403	5719	5046	4002
K	4092	5316	6127	6386	6011	6230	5974	6325	6202	6310	6202	6184	5893	6279	6164	6313	6242	6027	5889	6256	6329	6144	5093	4097
L	4272	5004	5885	6028	6126	6109	6139	6144	6235	6071	6156	6000	6024	6047	6193	6110	6146	5925	6067	6045	6210	5902	5231	4209
M	4164	5031	5897	5901	5920	6119	6001	5938	6085	5872	5948	5781	5845	6036	5926	5990	6055	5769	5747	6143	5945	5568	5049	4063
N	4234	4897	5763	5831	5966	5938	6085	5987	5811	5997	5737	5726	5785	5834	5967	5797	5880	5807	5904	5873	6035	5641	4739	4033
O	4100	4998	5833	5862	5825	6180	5970	5852	6036	5912	5916	5596	5672	5907	5805	5914	5765	5859	5867	6153	5863	5472	4921	3895
P	4250	5122	5621	6077	6005	6169	6239	6152	6044	6124	5778	5798	5729	5699	6032	5912	5710	5909	5977	6133	6266	5775	4765	3840
Q	3932	4740	5728	5992	6172	6051	6204	5905	6146	5911	6010	5760	5970	6019	5941	5797	5983	5860	6051	5922	6099	5432	4672	3671
R		4811	5214	5836	5760	5969	5810	5886	5870	6238	6075	5904	6083	5932	6033	6114	5772	5707	5632	5978	5767	5142	4405	
S		4138	5137	5286	5650	5667	5472	5971	6140	6067	5913	6055	6034	6187	5877	6069	6023	5483	5473	5684	5220	4804	3630	
T		3391	4534	4893	5348	5295	5652	5844	6134	6240	6112	5765	6111	6032	6180	5982	6004	5518	5446	5082	4885	4324	3046	
U			3779	4053	4581	5151	5506	6005	6006	6210	5915	5857	5860	6249	6204	6001	5959	5308	4940	4801	4014	3617		
V				3557	3844	4724	5373	5757	6068	6234	5851	5526	5926	6079	6035	6163	5642	5227	4548	4130	3663			
W					3670	4191	4497	5402	5792	5700	5854	5615	5725	6053	5990	5747	5316	4421	4011	3301				
X						3380	4082	4520	4662	5296	5170	5157	5023	5323	4990	5050	4303	3871	2989					
Y								3566	3943	4234	4431	4293	4503	4432	4333	3999	3484							

Colour Format:	7000 kW	5000 kW	3000 kW

Figure 12. Instantaneous channel power distribution obtained using the patterned-age shown in Figure 11 (in kW). The value of k_{eff} is 0.983, max channel power is 6591 kW at channel G-6, max bundle power is 810 kW at channel G-6, bundle #6, and the radial form factor is 1.19

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24
A								1.11	1.06	1.09	1.02	1.04	1.00	1.07	1.03	1.09	1.08							
B						1.04	0.98	1.09	1.06	1.01	0.99	1.01	1.01	1.03	0.97	1.05	1.08	1.05	1.09					
C					1.01	0.93	1.05	1.02	1.04	1.03	1.01	0.97	1.01	0.99	1.02	0.99	1.02	1.01	1.00	1.07				
D				0.90	0.92	0.97	0.99	1.01	0.97	0.99	0.96	0.97	0.96	1.00	0.98	0.94	0.98	0.95	0.96	1.00	1.04			
E			0.92	0.96	0.89	0.97	1.00	0.98	0.99	1.01	0.96	0.95	0.99	0.96	0.93	0.94	0.92	0.95	0.93	0.97	1.03	1.01		
F		1.11	1.07	0.98	1.00	0.99	0.97	1.02	1.01	0.96	0.99	0.98	0.99	0.96	0.97	0.92	0.95	0.93	0.96	0.99	0.97	1.00	0.93	
G		1.10	1.05	1.03	1.00	1.01	1.00	1.01	0.98	1.01	0.99	1.02	1.01	1.00	0.96	0.97	0.95	0.98	0.97	1.01	1.02	0.99	1.05	
H		0.95	1.05	1.01	0.99	1.01	0.98	1.00	1.02	0.99	1.00	0.98	1.02	1.00	0.99	0.96	1.01	1.00	0.99	1.01	0.99	1.00	1.07	
J	1.09	1.03	1.00	0.98	1.00	1.00	1.02	0.97	1.01	1.01	0.96	0.99	1.00	0.97	1.01	1.00	0.99	1.00	1.01	0.99	1.01	0.97	1.02	1.05
K	1.00	1.03	1.02	1.00	0.96	1.02	0.99	1.03	0.99	1.00	0.99	1.01	0.96	1.01	0.98	1.01	1.03	1.02	0.98	1.00	0.99	1.03	0.99	1.02
L	1.00	0.97	0.99	0.96	0.98	0.99	1.01	1.00	1.00	0.97	0.99	0.98	0.98	0.98	1.00	0.99	1.00	0.99	0.99	0.97	1.00	1.00	1.02	1.02
M	0.97	0.97	1.00	0.95	0.95	1.00	0.99	0.98	1.00	0.96	0.98	0.97	0.98	1.00	0.97	0.99	1.00	0.96	0.95	1.00	0.98	0.96	0.99	0.97
N	1.00	0.95	0.98	0.94	0.96	0.96	1.00	0.99	0.96	1.00	0.97	0.99	1.00	0.99	1.00	0.96	0.98	0.97	0.97	0.96	1.00	0.98	0.94	0.97
O	0.99	0.99	1.00	0.95	0.94	1.00	0.99	0.97	1.00	0.99	1.01	0.99	1.00	1.01	0.98	0.99	0.96	0.98	0.96	1.00	0.97	0.95	0.99	0.96
P	1.08	1.03	0.97	0.99	0.98	1.00	1.04	1.02	1.01	1.03	0.99	1.03	1.02	0.98	1.02	0.99	0.96	0.99	0.98	1.00	1.02	1.00	0.97	0.99
Q	1.07	1.02	1.04	1.00	1.00	0.98	1.03	0.99	1.03	1.00	1.03	1.01	1.04	1.04	1.01	0.98	1.02	0.99	1.00	0.97	1.02	0.98	1.00	1.01
R		1.11	1.03	1.03	1.00	1.04	1.02	1.01	0.98	1.04	1.02	1.00	1.03	0.99	1.01	1.02	0.99	1.00	0.98	1.03	1.01	1.00	1.02	
S		1.09	1.10	1.00	1.01	1.03	0.99	1.03	1.02	1.00	0.98	1.01	1.01	1.03	0.97	1.01	1.02	0.97	0.99	1.01	0.99	1.03	0.97	
T		1.05	1.11	1.02	1.02	0.98	1.03	1.00	1.00	1.03	1.01	0.97	1.03	1.00	1.02	1.00	1.02	0.99	1.00	0.97	1.02	1.06	0.95	
U			1.08	0.97	0.96	1.00	1.00	1.01	0.98	1.01	0.98	1.00	0.99	1.03	1.01	0.98	1.00	0.97	0.96	1.00	0.96	1.04		
V				1.00	0.92	0.99	1.02	1.00	1.01	1.03	0.99	0.97	1.03	1.01	0.99	1.01	0.97	0.98	0.95	0.99	1.03			
W					1.06	1.03	0.97	1.05	1.04	1.00	1.03	1.02	1.04	1.06	1.05	1.03	1.03	0.95	0.99	0.95				
X						1.08	1.10	1.07	1.01	1.10	1.05	1.06	1.03	1.08	1.03	1.09	1.02	1.04	0.95					
Y								1.11	1.11	1.10	1.11	1.07	1.12	1.11	1.13	1.13	1.09							

Colour Format:	1.3 (unitless)	1.0 (unitless)	0.7 (unitless)

Figure 13. CPPFs of each fuel channel of the instantaneous snapshot core. CPPFs displayed here serve the purpose of comparing the instantaneous channel powers generated using the patterned-age shown in Figure 11, to that of the reference power distribution

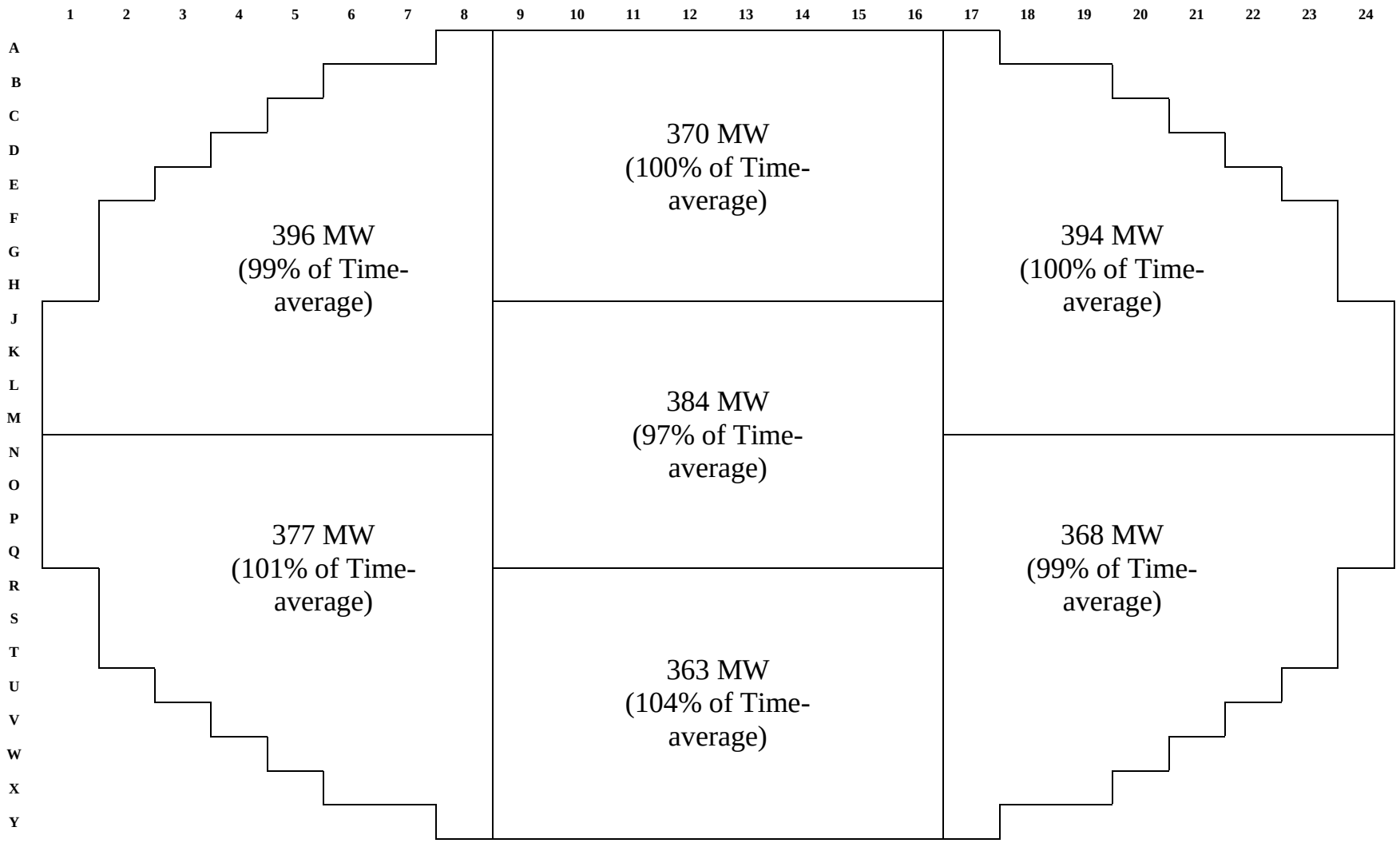


Figure 14. Regionalized instantaneous power distribution of the core snapshot generated using the patterned-age shown in Figure 11 (in MW). Regional power peaking factors (comparison to reference regional powers) are also displayed

The instantaneous core resulting from the patterned age shown in Figure 11 takes into account the impact of refuelling ripples. Channels at very low ages (near zero) operate at powers greater than their time-average values. This is because a low value of channel age near zero is representative of a channel containing fresh fuels which experience the effects of refuelling transients. For this reason, channels at very low ages typically possess values of CPPFs greater than one.

In CANDU reactors, the "equilibrium refuelling state" is defined as the state of operation when the rate of refuelling the reactor is constant [1]. This is opposed to the state of "approach to refuelling equilibrium" which exists when a fresh core first comes online. For a fresh core, there is an initial period of about 6 months when no refuelling is carried out and the excess reactivity of the core is compensated by the presence of boron (a type of BNA) that is injected into the moderator [1]. As the excess reactivity in the core declines, the moderator poison is gradually used up or removed and refuelling begins at an initially high rate, which declines progressively until an equilibrium rate is achieved [1]. The instantaneous snapshot of the model core (shown in Figure 12) resulting from the patterned-age used in this study is a constructed core state that is, by design, essentially at the equilibrium refuelling state. To achieve a core state that is as close as possible to the refuelling equilibrium, the constructed core is refuelled from its initial state for an extensive period of time (~400 FPDs) using the automatic refuelling algorithm to ensure that the rate of refuelling has reached an equilibrium. The resulting core state is a generic set of irradiation data representing the model core at the equilibrium refuelling state. In this study, all refuelling and core-following simulations are carried out starting from the above core state, because it provides a good representation of the normal operating conditions of the modelled CANDU reactor. Also, by consistently using the same starting core state for simulations, parameters other than the quantity of BNAs within the fuel, which can also affect the performance of fuels, are kept constant between simulations. Figures 15 to 21 shown below outline the instantaneous core snapshot data of the generic equilibrium refuelling core state. These instantaneous core data are collected in particular because they are used in core-following and refuelling simulations to decide which channels should be selected for refuelling at

each instance of refuelling. Figure 15 presents the distribution of channel powers and provides the values of several other important parameters including the k_{eff}, the maximum channel and bundle powers, the radial form factor, and the rate of bulk reactivity decline. Figure 16 outlines the CPPFs of each fuel channel. Figure 17 outlines the regionalized distribution of channel powers. Figure 18 presents the highest powered fuel bundle contained within each fuel channel. Figure 19 outlines the average discharge burnup of fuels that would result for each fuel channel in the instance that they are refuelled. Figure 20 presents the axial tilts in power of each fuel channel. Figure 21 outlines the increase in the bulk reactivity of the core that would result, for each fuel channel, in the instance that they are refuelled.

Based on comparison with typical, known values of channel powers, CPPFs, the radial form factor, and the distribution of powers for the model core during normal operation, the generated equilibrium refuelling core state data is a good representation of the model core at its refuelling equilibrium. It is therefore expected that the core-following simulations involving transition-refuelling (using BNAFs) from the constructed core state will provide a good case study for demonstrating a gain in margin for the model core when refuelling using BNAFs during normal operation.

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24
A								3463	3620	3955	4227	4040	4243	4070	3849	3786	3421							
B						3727	4222	4351	4926	5045	5257	5024	4955	5263	5206	4987	4353	3892	3630					
C					4108	4524	5134	5424	5712	5822	5938	5964	5841	5906	5988	5708	5295	5082	4593	3967				
D				4329	4951	5532	5791	6127	6306	6262	6196	6400	6300	6554	6447	6459	6178	5701	5358	4770	3791			
E			4100	4970	5599	5858	6057	6135	6247	6068	6240	6468	6464	6487	6416	6385	6435	6083	5879	5280	4292	3535		
F		3673	4429	5273	5905	6021	5943	5955	6294	6104	5991	6209	6265	6416	6547	6364	6412	6174	6003	5757	5092	4345	3048	
G		4081	5160	5786	5955	5926	6046	5895	6128	6027	6206	5983	5921	6096	6438	6413	6298	6310	6414	6135	5679	4986	3789	
H		4485	5293	5864	6121	6164	5963	5779	6233	6234	6267	5990	5798	5865	6252	6216	6098	5980	6187	6287	6103	5404	4193	
J	3910	4818	5602	6162	6295	5763	5732	5882	5934	6101	6131	5674	5836	6050	6016	5843	6174	6130	6036	6174	6066	5731	4645	3558
K	4261	5055	5888	6151	6191	5899	5500	5733	6089	5987	6314	6059	6030	5919	6079	6103	5908	5964	6123	6028	6001	5812	4983	3823
L	4396	5247	6045	6189	6416	6192	5791	5730	6030	6212	6186	6291	6206	6090	5985	6101	5834	6017	6007	6276	6286	5855	5114	4224
M	4292	5020	6108	6279	6413	6195	5858	6063	5995	6238	6357	6296	5906	6052	6140	5871	5883	6070	6232	6194	6077	5722	5031	4134
N	4204	4872	6007	6391	6250	6391	6171	6023	5861	6056	6251	6195	5988	6130	6026	5717	5799	5795	6094	6351	6120	5889	4984	3866
O	3724	4889	5670	6250	6361	6330	6346	6242	5947	6146	6030	6106	6052	6010	6086	5908	6054	5907	6188	6178	6123	5702	4964	3654
P	3773	4823	5526	6108	6376	6129	6185	6113	5902	5977	6137	5921	5797	6061	6025	6118	5788	5975	6261	5983	6005	5763	4882	3606
Q	3543	4649	5281	5876	6226	6257	6109	6225	6155	6198	5993	6036	5893	6039	6227	6162	6059	6195	6344	6069	5887	5333	4468	3429
R		4141	4918	5842	6075	6039	6272	6330	6244	6084	5874	6113	5960	6078	5796	5991	6186	6163	6116	6076	5659	4817	3924	
S		3750	4889	5596	5933	6095	6083	6389	6401	6193	6126	5912	6114	5986	6070	6200	6177	5858	5812	5841	5298	4674	3394	
T		3221	4191	4927	5409	5706	5837	6169	6221	6065	6207	6074	5979	6210	6298	6162	6129	5540	5439	5361	4874	4073	2954	
U			3709	4452	5046	5614	5926	6222	6075	6169	6073	6028	6003	6251	6203	5957	6059	5582	5337	5027	4342	3542		
V				3718	4519	5217	5759	5951	6060	6215	5971	5866	5760	6027	5919	5920	5896	5476	4962	4097	3663			
W					3751	4136	4800	5456	5745	5740	5931	5732	5767	5857	5795	5528	5197	4711	3992	3313				
X						3335	3897	4240	4808	5058	5127	4971	4739	4892	4664	4464	4268	3768	2976					
Y								3272	3612	3779	3891	3723	3795	3698	3617	3459	3210							

Colour Format:	7000 kW	5000 kW	3000 kW

Figure 15. Instantaneous channel power distribution of the generic equilibrium core (in kW). The value of k_{eff} is 1.000, max channel power is 6554 kW at D-14, max bundle power is 799 kW at D-14, the radial form factor is 1.19, and the rate of reactivity decline is -0.4167 milli-k FPD^{-1}

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24
A								1.02	0.97	0.99	1.02	0.96	1.01	0.98	0.96	1.02	1.01							
B						1.08	1.06	0.98	1.03	1.00	1.02	0.97	0.96	1.02	1.03	1.04	0.98	0.97	1.05					
C					1.08	1.05	1.05	1.01	1.01	1.00	1.00	1.01	0.99	1.00	1.02	1.01	0.99	1.04	1.06	1.04				
D				1.11	1.08	1.06	1.02	1.02	1.03	1.01	0.99	1.02	1.01	1.04	1.03	1.05	1.02	1.00	1.02	1.04	0.98			
E			1.09	1.09	1.06	1.00	0.99	0.99	1.00	0.96	0.99	1.02	1.02	1.02	1.01	1.00	1.01	0.99	1.01	1.00	0.94	0.94		
F		1.06	1.01	1.00	1.00	0.96	0.94	0.93	0.99	0.96	0.95	0.99	1.00	1.01	1.02	0.99	0.99	0.97	0.96	0.97	0.97	0.99	0.89	
G		1.01	1.03	1.00	0.95	0.91	0.94	0.91	0.95	0.94	0.99	1.00	0.99	0.96	1.00	0.99	0.97	0.98	0.99	0.98	0.98	1.00	0.95	
H		0.98	0.97	0.96	0.96	0.95	0.93	0.89	0.96	0.97	0.99	0.98	0.95	0.93	0.97	0.96	0.96	0.94	0.97	0.99	0.99	0.98	0.92	
J	1.01	0.98	0.97	0.97	0.99	0.92	0.93	0.93	0.92	0.95	0.96	0.91	0.94	0.95	0.94	0.91	0.98	1.01	0.98	0.97	0.96	0.98	0.95	0.94
K	1.04	0.98	0.98	0.96	0.97	0.95	0.90	0.92	0.95	0.93	0.99	0.96	0.96	0.93	0.95	0.96	0.96	0.99	1.01	0.96	0.94	0.97	0.97	0.95
L	1.03	1.00	1.01	0.97	1.01	0.98	0.94	0.92	0.95	0.97	0.97	1.00	0.99	0.96	0.94	0.97	0.93	0.99	0.97	1.00	1.00	0.99	1.00	1.02
M	0.99	0.96	1.02	0.99	1.01	0.99	0.95	0.98	0.96	1.00	1.02	1.03	0.96	0.98	0.98	0.94	0.96	0.99	1.01	0.99	0.98	0.97	0.98	0.98
N	0.98	0.93	1.01	1.02	0.99	1.01	1.00	0.97	0.95	0.98	1.03	1.04	1.01	1.01	0.98	0.93	0.95	0.95	0.98	1.02	1.00	1.01	0.98	0.93
O	0.89	0.96	0.96	1.00	1.00	1.01	1.03	1.01	0.97	1.01	1.00	1.05	1.04	1.00	1.00	0.96	0.99	0.97	0.99	0.99	1.00	0.98	0.99	0.90
P	0.95	0.96	0.95	0.99	1.03	0.98	1.01	1.00	0.96	0.98	1.03	1.02	1.00	1.02	1.00	1.01	0.96	0.98	1.01	0.96	0.97	1.00	0.99	0.93
Q	0.96	0.99	0.96	0.98	1.00	1.01	1.00	1.03	1.01	1.03	1.01	1.03	1.01	1.02	1.04	1.02	1.01	1.03	1.04	0.99	0.98	0.97	0.96	0.95
R		0.96	0.97	1.03	1.05	1.05	1.10	1.07	1.03	1.00	0.97	1.02	1.00	1.01	0.96	1.00	1.06	1.08	1.06	1.05	1.00	0.94	0.92	
S		1.00	1.06	1.07	1.07	1.11	1.10	1.10	1.06	1.02	1.01	0.99	1.02	0.99	1.00	1.04	1.05	1.04	1.06	1.05	1.01	1.01	0.91	
T		1.00	1.04	1.04	1.04	1.07	1.07	1.06	1.03	1.00	1.03	1.03	1.01	1.04	1.05	1.04	1.05	1.01	1.02	1.04	1.03	1.01	0.93	
U			1.08	1.08	1.08	1.11	1.10	1.07	1.01	1.02	1.02	1.04	1.03	1.04	1.03	1.00	1.05	1.04	1.07	1.07	1.06	1.03		
V				1.07	1.11	1.12	1.12	1.07	1.04	1.05	1.03	1.05	1.03	1.03	1.00	1.00	1.05	1.06	1.06	1.01	1.06			
W					1.11	1.05	1.07	1.10	1.07	1.05	1.08	1.08	1.08	1.06	1.05	1.03	1.04	1.05	1.02	0.99				
X						1.10	1.09	1.05	1.09	1.09	1.08	1.06	1.01	1.03	1.00	1.01	1.05	1.05	0.99					
Y								1.07	1.07	1.03	1.02	0.97	0.98	0.97	0.98	1.02	1.05							

Colour Format:	1.3 (unitless)	1.0 (unitless)	0.7 (unitless)

Figure 16. The CPPFs of each channel of the generic equilibrium core

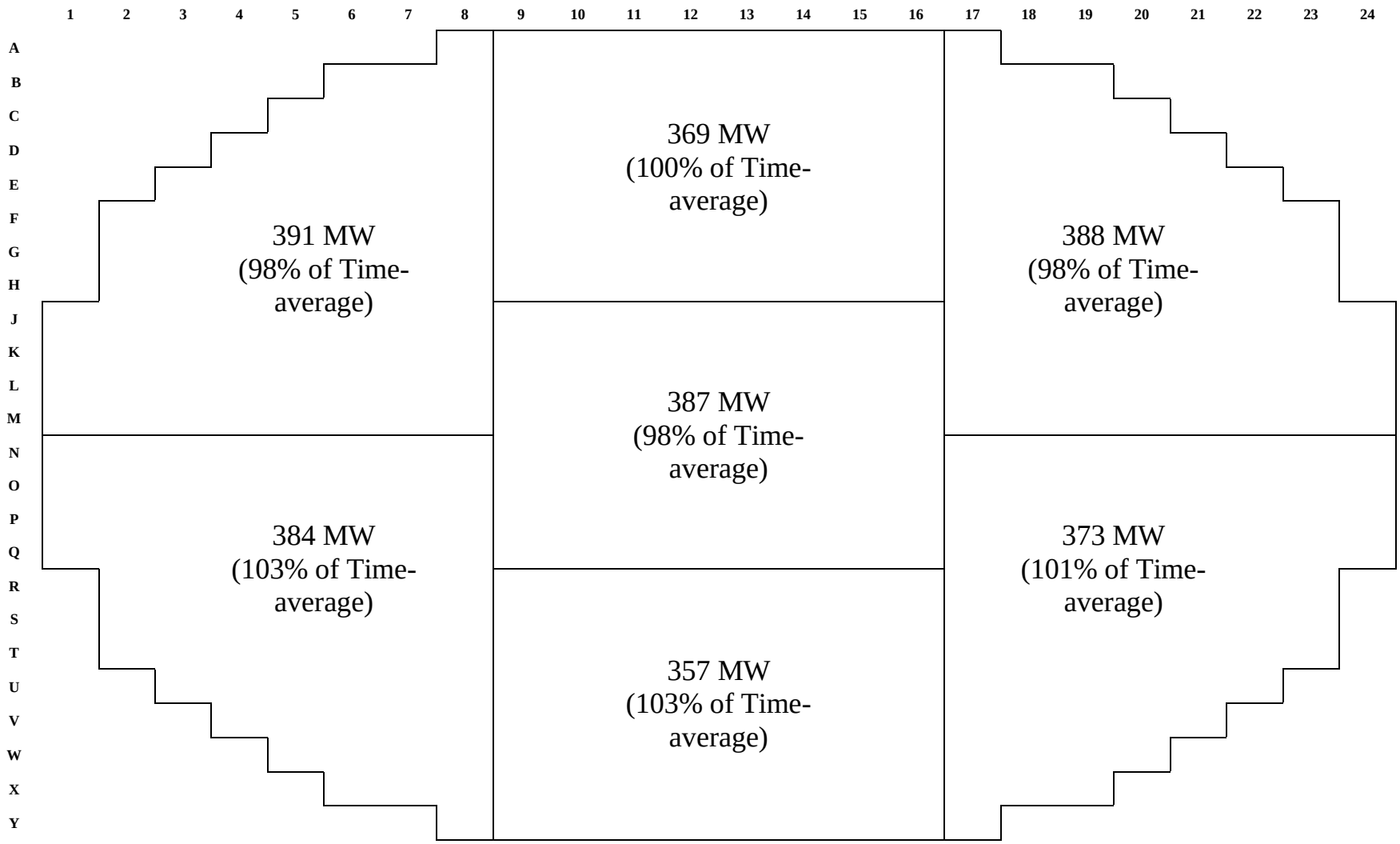


Figure 17. Regionalized instantaneous power distribution of the generic equilibrium core (in MW). Regional power peaking factors (comparison to reference regional powers) are also displayed

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24
A								440	450	492	530	503	540	506	475	477	433							
B						485	538	533	612	621	649	618	608	661	642	621	535	483	470					
C					530	572	642	673	694	704	735	739	719	714	740	691	653	637	586	508				
D				566	631	703	727	741	772	753	734	791	757	799	769	798	752	713	674	604	476			
E			536	639	700	721	729	746	726	711	742	787	782	773	787	760	794	726	736	652	526	442		
F		478	557	674	734	717	702	670	742	679	670	749	742	748	776	775	752	740	714	712	630	545	378	
G		519	654	721	713	701	691	670	679	688	720	693	683	683	769	746	738	746	786	745	719	627	473	
H		563	662	710	752	731	688	621	712	680	709	706	653	657	715	761	687	706	738	781	767	695	518	
J	501	613	692	774	762	688	660	657	638	692	695	619	651	658	722	661	729	746	752	743	756	724	589	448
K	551	638	736	738	757	703	629	625	699	645	700	692	656	666	683	743	670	728	756	736	724	733	639	482
L	575	675	758	762	773	752	653	644	659	712	703	698	680	651	719	711	679	713	733	756	787	727	652	545
M	554	625	774	759	786	717	666	667	703	682	690	710	641	686	710	722	658	719	736	750	742	715	633	527
N	546	601	748	792	733	768	686	695	649	705	694	682	656	661	721	657	702	633	727	772	766	736	637	486
O	457	608	706	749	777	724	736	693	708	681	668	694	687	659	687	726	696	684	716	760	743	719	633	454
P	472	614	670	750	760	720	690	708	656	708	666	653	630	660	713	701	680	669	760	700	744	721	624	450
Q	445	587	657	705	769	735	713	699	727	695	658	673	663	667	703	738	670	731	766	737	717	680	564	432
R		509	599	736	737	745	763	753	699	715	645	668	645	682	653	661	721	747	759	743	717	589	482	
S		469	612	701	753	777	756	766	759	689	685	657	695	662	670	739	722	719	717	734	662	587	419	
T		407	526	629	674	736	717	738	719	717	714	687	671	721	746	715	742	671	689	675	616	514	370	
U			476	568	641	734	763	759	730	720	709	730	731	736	721	712	730	722	701	648	558	455		
V				476	583	680	734	746	725	762	712	728	706	733	714	705	737	698	640	518	475			
W					485	521	600	683	715	702	748	746	752	747	728	686	653	596	506	419				
X						427	492	525	603	633	652	648	600	626	581	554	536	478	374					
Y								411	453	472	488	464	478	461	452	435	409							

Colour Format:	840 kW	750 kW	500 kW

Figure 18. Instantaneous distribution of maximum bundle powers (highest power bundle in channel) of the generic equilibrium core (in kW)

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24
A								114	151	136	59	152	90	139	163	111	121							
B						97	112	155	92	130	113	172	176	135	53	89	148	163	96					
C					114	138	62	166	152	165	171	155	171	169	160	166	184	97	77	112				
D				80	112	135	169	163	155	193	196	175	195	163	175	151	164	179	153	79	140			
E			98	82	61	159	173	197	185	209	193	141	168	197	209	188	164	178	150	123	158	153		
F		96	148	188	155	172	186	194	153	199	213	168	168	186	170	204	175	187	181	154	169	66	188	
G		126	94	149	171	179	168	206	192	211	175	203	218	205	179	176	184	164	131	164	158	107	129	
H		141	189	173	186	164	172	211	155	187	172	168	196	231	173	201	213	213	193	176	136	159	156	
J	119	183	177	164	164	221	209	182	230	200	194	214	199	189	216	232	180	162	213	192	177	147	168	145
K	65	182	168	177	193	210	215	229	192	215	164	197	192	216	181	196	217	197	177	209	177	163	163	151
L	150	182	173	206	177	167	195	218	209	195	207	182	160	188	212	179	205	163	207	189	172	174	183	140
M	172	198	160	191	191	201	218	179	199	185	182	167	223	208	173	228	201	168	176	200	210	182	190	157
N	152	198	165	189	225	191	201	214	232	216	182	166	213	183	199	236	220	240	213	188	216	169	199	205
O	173	194	204	194	193	215	182	181	217	181	205	173	168	203	186	220	188	198	199	216	186	187	182	184
P	123	178	180	189	187	217	204	204	228	215	186	196	212	185	205	174	226	215	208	216	183	146	147	145
Q	131	141	187	174	195	183	200	198	196	191	217	183	224	209	165	181	201	180	154	181	161	181	173	128
R		146	209	170	179	198	155	164	192	202	211	188	203	197	219	205	178	160	177	163	172	183	157	
S		133	56	113	164	146	178	151	161	194	190	213	184	215	192	182	151	176	175	146	170	66	170	
T		142	133	182	180	176	196	162	180	208	169	186	204	185	165	177	158	189	187	160	156	123	163	
U			107	107	129	128	154	154	180	169	191	173	189	160	169	183	150	181	147	71	104	115		
V				134	111	125	123	171	161	155	181	184	197	173	187	162	135	140	127	156	101			
W					98	149	139	117	144	155	144	152	132	139	143	150	169	117	132	154				
X						87	107	147	104	54	130	167	193	163	143	140	60	105	148					
Y								99	100	131	124	161	129	139	124	103	79							

Colour Format:	270 MW h kgU^{-1}	170 MW h kgU^{-1}	40 MW h kgU^{-1}

Figure 19. The average discharge burnup of fuels for each channel of the generic equilibrium core, which results if the channel is refuelled (in MW h kgU^{-1})

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24
A								4	-3	1	-3	0	-5	-1	-7	-2	-7							
B						2	6	0	2	-4	-4	-9	0	-6	-6	-7	-2	-7	-1					
C					1	7	5	9	-4	3	-9	0	-10	1	-10	0	-12	-3	-4	0				
D				1	5	3	11	-1	6	-8	4	-10	3	-9	0	-11	-2	-13	0	-3	1			
E			-2	2	3	8	-1	12	-3	9	-11	-4	-7	2	-19	-4	-12	1	-7	-1	-4	1		
F		-1	3	-6	7	-3	10	-3	7	-6	4	-11	2	-12	-7	-20	-5	-14	1	-10	2	-4	0	
G		4	0	5	-4	9	-2	12	-4	10	-10	1	-12	2	-18	-9	-18	-5	-9	-2	-12	-4	-8	
H		-1	11	-4	11	-1	9	-6	6	-6	-1	-12	3	-16	-8	-24	-4	-19	-2	-14	-7	-14	-5	
J	-1	10	-5	8	0	11	-6	8	-8	9	-13	1	-13	-1	-23	-7	-19	-10	-21	-2	-16	-6	-15	-4
K	1	-6	9	-2	11	-6	9	-6	10	-6	-2	-16	2	-16	-7	-24	-7	-21	-8	-19	-4	-14	-3	-11
L	-1	9	-3	12	-1	7	-5	13	-3	10	-12	-2	-10	0	-23	-12	-23	-10	-21	-4	-14	-1	-14	-7
M	9	-4	6	-5	10	-3	13	1	16	-2	3	-9	5	-17	-11	-27	-8	-17	-6	-17	0	-16	-2	-11
N	2	10	-1	9	-6	9	-2	16	1	15	-5	1	-10	0	-22	-9	-26	-1	-19	-7	-19	-5	-16	2
O	9	-2	14	-2	13	-3	10	3	19	2	11	-5	2	-12	-6	-26	-10	-19	-4	-20	-4	-17	-3	-13
P	4	13	-1	13	1	15	0	17	2	19	-3	8	-8	3	-20	-8	-22	-3	-17	-2	-17	-6	-11	-4
Q	10	5	15	2	16	2	18	3	18	2	11	-6	11	-11	-5	-20	-3	-16	-7	-15	-4	-16	0	-9
R		9	-1	13	3	17	9	16	3	17	-5	6	-7	7	-12	-1	-15	-7	-14	-3	-13	0	-8	
S		5	9	10	15	9	16	9	16	2	11	-7	9	-9	2	-15	-5	-14	0	-8	2	-4	-1	
T		11	7	17	4	17	2	14	4	17	-1	8	-5	8	-8	-1	-10	0	-12	1	-8	-1	-7	
U			12	8	12	9	13	6	16	3	11	-3	10	-1	3	-12	0	-11	-2	-4	-1	-6		
V				12	6	10	8	15	2	11	-4	9	-4	7	-9	4	-4	1	-4	0	-5			
W					10	5	11	8	10	0	6	-2	2	-1	2	-5	5	-3	1	-5				
X						9	5	9	3	4	2	5	-7	5	-3	3	0	2	-4					
Y								3	6	1	4	-1	3	-2	3	-1	2							

Colour Format:	20 %	0 %	-20 %

Figure 20. The instantaneous axial tilts in power of each channel of the generic equilibrium core (in percentages)

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24
A								0.03	0.06	0.05	0.02	0.07	0.02	0.06	0.08	0.03	0.03							
B						0.02	0.03	0.09	0.03	0.07	0.01	0.06	0.07	0.03	0.02	0.02	0.08	0.08	0.02					
C					0.04	0.07	0.03	0.06	0.06	0.06	0.07	0.05	0.08	0.08	0.06	0.07	0.08	0.03	0.01	0.03				
D				0.01	0.05	0.02	0.07	0.06	0.05	0.08	0.11	0.06	0.10	0.06	0.09	0.05	0.06	0.08	0.04	0.02	0.06			
E			0.02	0.02	0.03	0.06	0.06	0.10	0.08	0.12	0.09	0.03	0.06	0.09	0.11	0.10	0.07	0.09	0.04	0.07	0.10	0.06		
F		0.02	0.09	0.07	0.05	0.08	0.09	0.10	0.05	0.10	0.12	0.06	0.06	0.08	0.07	0.11	0.07	0.09	0.10	0.06	0.06	0.02	0.07	
G		0.05	0.03	0.04	0.09	0.10	0.06	0.10	0.09	0.11	0.07	0.08	0.09	0.11	0.07	0.08	0.09	0.06	0.04	0.07	0.05	0.04	0.04	
H		0.07	0.08	0.08	0.09	0.05	0.07	0.11	0.05	0.08	0.06	0.06	0.09	0.11	0.07	0.09	0.11	0.11	0.09	0.07	0.04	0.05	0.09	
J	0.04	0.05	0.08	0.06	0.05	0.11	0.08	0.07	0.11	0.09	0.08	0.11	0.08	0.08	0.09	0.12	0.05	0.04	0.10	0.09	0.08	0.04	0.05	0.05
K	0.01	0.07	0.07	0.09	0.10	0.09	0.10	0.09	0.07	0.11	0.05	0.09	0.08	0.11	0.09	0.07	0.10	0.07	0.06	0.12	0.10	0.06	0.04	0.06
L	0.02	0.05	0.07	0.12	0.07	0.05	0.08	0.11	0.09	0.08	0.10	0.06	0.07	0.09	0.10	0.07	0.10	0.05	0.10	0.08	0.06	0.08	0.05	0.01
M	0.03	0.08	0.05	0.11	0.08	0.09	0.11	0.07	0.09	0.08	0.07	0.06	0.12	0.09	0.06	0.10	0.09	0.05	0.07	0.10	0.12	0.10	0.06	0.03
N	0.02	0.08	0.06	0.07	0.14	0.07	0.09	0.10	0.11	0.10	0.07	0.06	0.09	0.07	0.08	0.11	0.11	0.13	0.11	0.06	0.10	0.05	0.06	0.05
O	0.06	0.06	0.10	0.09	0.11	0.11	0.06	0.07	0.11	0.08	0.11	0.05	0.05	0.09	0.08	0.10	0.06	0.10	0.09	0.11	0.08	0.08	0.04	0.05
P	0.04	0.04	0.08	0.08	0.08	0.13	0.10	0.09	0.12	0.11	0.07	0.08	0.09	0.08	0.09	0.08	0.12	0.10	0.07	0.12	0.08	0.04	0.03	0.05
Q	0.04	0.02	0.08	0.09	0.08	0.09	0.10	0.08	0.09	0.08	0.10	0.07	0.10	0.09	0.06	0.08	0.10	0.06	0.05	0.09	0.06	0.06	0.04	0.03
R		0.07	0.09	0.05	0.08	0.10	0.04	0.07	0.10	0.10	0.12	0.07	0.10	0.08	0.11	0.11	0.06	0.04	0.07	0.06	0.06	0.07	0.08	
S		0.05	0.01	0.02	0.06	0.04	0.06	0.03	0.06	0.09	0.09	0.11	0.07	0.11	0.10	0.07	0.06	0.07	0.07	0.04	0.06	0.02	0.07	
T		0.04	0.06	0.06	0.08	0.08	0.09	0.07	0.09	0.11	0.07	0.08	0.10	0.08	0.05	0.08	0.04	0.08	0.08	0.06	0.04	0.04	0.05	
U			0.02	0.04	0.07	0.02	0.04	0.05	0.10	0.08	0.09	0.06	0.08	0.05	0.07	0.10	0.05	0.07	0.03	0.02	0.03	0.03		
V				0.05	0.04	0.01	0.01	0.07	0.08	0.06	0.10	0.08	0.09	0.08	0.10	0.08	0.03	0.03	0.02	0.08	0.02			
W					0.02	0.07	0.08	0.01	0.04	0.07	0.03	0.04	0.02	0.02	0.02	0.04	0.05	0.05	0.05	0.05				
X						0.01	0.03	0.07	0.04	0.02	0.02	0.03	0.07	0.03	0.08	0.07	0.02	0.02	0.04					
Y								0.02	0.02	0.05	0.04	0.07	0.04	0.05	0.04	0.02	0.01							

Colour Format:	0.15 milli-k	0.065 milli-k	0 milli-k

Figure 21. The increase in the bulk reactivity of the core for each channel of the generic equilibrium core, which results if the channel is refuelled (in milli-k)

5.5.4 Phase 4: Refuelling Simulations

The refuelling simulations are conducted to determine the impact of added absorbers on the transient powers of fresh fuelled channels. Simulations in RFSP capture the effects of refuelling in three dimensions, with the smallest unit of calculated flux, power, or any other parameters of interest being determined at the fuel bundle level. Refuelling simulations using RFSP can therefore account for the transient power of a volume of the reactor core in the vicinity of the fresh inserted fuels. In corollary, the impact of added absorbers is also accounted for in three dimensions, such that the reduction in transient powers caused by the added absorbers is reflected in adjacent fuel bundles and channels as well. The impact of added absorbers observed from the refuelling simulations are to be compared with the impact observed in WIMS lattice calculations, which only illustrate the effects within the single fuel bundle lattice.

In this part of the study, the transient powers of fresh fuelled channels are observed from immediately after refuelling (0 FPD) to 50 FPDs of in-core irradiation. This is done in order to calculate the transient powers of the fresh-fuelled channels during the period of in-core irradiation within which the fuelling transient and the plutonium peak should occur within the fresh fuels. The transient powers of the fresh-fuelled channels are determined for the regular NU fuel and BNAFs containing the varied quantities of added absorbers. The reduction in the transient powers of fresh-fuelled channels caused by the addition of the select neutron absorbers can then be determined.

In this study, refuelling simulations are conducted using the reference fuelling scheme of the model core. Figure 22 below outlines the details of the reference fuelling scheme. In addition to the reference fuelling scheme, a refuelling schematic consistent entirely of the 8-bundle-shift mode, which is the typical fuelling mode used in CANDU6 reactors, will also be investigated. This is done to determine the feasibility of refuelling all 480 channels of the model core using the 8-bundle-shift fuelling mode. As shown in Figure 22, the reference fuelling scheme incorporates a combination of the 4-bundle-shift mode in the central region of the core and the 8-bundle-shift mode in the periphery of the

core for refuelling. This is done because the high-flux central region of the core typically operates at significantly higher powers than the periphery of the core, and it also experiences significantly larger power transients when channels within the region are refuelled. The reference fuelling scheme for the model core therefore utilizes the 4-bundle-shift mode to refuel the central region of the core as it imparts smaller, more manageable transient powers than the 8-bundle-shift mode. By doing so, the reference fuelling scheme facilitates a refuelling practice which helps to ensure that all fuel channels are kept near their desired reference powers, and within the limits of their operating envelopes.

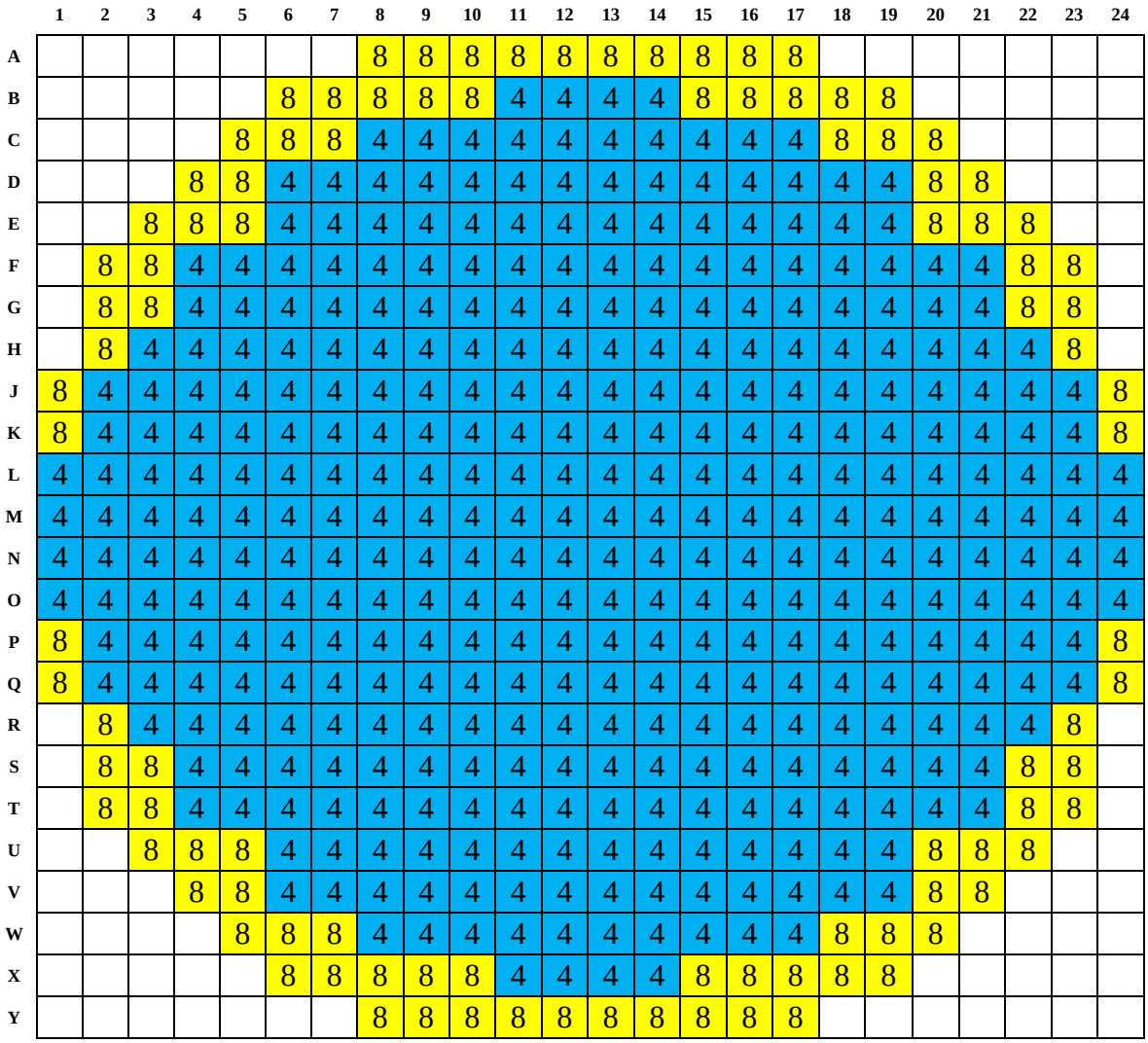


Figure 22. The reference fuelling scheme of the model CANDU core. Regions coloured in blue are refuelled using 4-bundle-shift mode and regions coloured in yellow are refuelled using 8-bundle-shift mode.

The difference between the resulting transient powers when refuelling with the 4-bundle-shift mode and the 8-bundle-shift mode is in part due to the fact four more fuel bundles are inserted when refuelling with the latter. However, the axial positions along the fuel channels, into which each of the fuelling modes insert fresh fuel bundles, also contributes to the differences in the transient powers. This is because the axial flux profile of fuel channels typically consists of a cosine shape along the length of the fuel channel from its inlet to the outlet. This causes fuel bundles located near the centers of the fuel channels, in axial bundle positions 5 to 8, to operate at higher powers. When refuelling using the 4-bundle-shift mode, the fuel bundles that are shifted into the high-flux axial positions are irradiated fuels that have already passed their refuelling transients. The 8-bundle-shift fuelling mode, however, inserts fresh fuel bundles directly into bundle positions 1 to 8, which includes the high-flux central positions along the axis of the fuel channel. For this reason, the impact of using added absorbers to reduce the magnitudes of the refuelling transients is greater when refuelling with the 8-bundle-shift fuelling mode than the 4-bundle-shift mode.

It is interesting to investigate refuelling the model core using only the 8-bundle-shift fuelling mode, because this inserts larger quantities of reactivity than the 4-bundle-shift mode at each refuelling, and so allows refuelling operations to occur at lower frequencies. This is desired in practice, because it reduces the strain on refuelling machines to perform their daily tasks of refuelling, thus improving the reliability of these sophisticated machines.

5.5.5 Phase 5: Core-following Simulations

As the final phase of the study, core-following simulations are conducted for the generic, refuelling equilibrium state core using the reference NU fuel and the BNAFs. Simulations for all fuels depart from the same instantaneous snapshot of the generic refuelling equilibrium core using the same refuelling history and the same refuelling scheme. Two types of simulations are conducted for each fuel type. The first simulation is conducted using the reference fuelling scheme shown in Figure 22. The second simulation is conducted using only the 8-bundle-shift refuelling mode. This is done to assess the feasibility of using reductions in the magnitudes of refuelling ripples, obtained via the addition of BNAs, to refuel all of the fuel channels in the model core using the 8-bundle-shift fuelling mode. Over the course of the core-following simulations, instantaneous core data are collected at every 0.25 FPDs. The collected data include the same information that are outlined in Figures 15 to 21, which are used for channel selection by the refuelling algorithm. In addition, however, the locations of the channels that are selected for refuelling are also logged, with the actual average burnup of the discharged fuels from refuelled channels being collected as well. Also, the individual and the average fill of LZCs, the k_{eff}, the radial form factor, and the rate of bulk reactivity decline in the core are also collected.

The collected data are analyzed to determine if there are reductions in the maximum instantaneous powers and the peaking factors of fuel bundles and fuel channels, which would indicate improvements for the operating margins of fuels and fuel channels as well as the process of core flattening. The collected data for the instantaneous fills of LZCs are analyzed to determine reductions in the average fills of LZCS as well as mitigation of high fills in select LZCs, which would indicate an improvement for the economy of neutrons as well as conserving the added margin that is provided by the LZCs. The collected data for the average discharge burnup of fuels are analyzed to determine if there are any significant reductions in the average discharge burnup of fuels, which would indicate if the added absorbers are imparting any negative effects to the useful burnup of fuels.

Chapter 6: Results

In this section, the results of the refuelling and the core-following simulations for the regular NU fuel and each of the BNAFs being investigated in this study are presented. The simulations were conducted for both the reference fuelling scheme of the model core and a scheme that is entirely consistent of the 8-bundle-shift mode (F8 fuelling scheme). These results are further discussed in detail in Chapter 7.

6.1 Refuelling Simulations

The resulting reductions in the post-refuelling, transient powers of fuel channels, in comparison to refuelling with the regular NU fuel, are outlined in Figures 23 to 28. The resulting reductions in the transient powers of the highest powered fuel bundles within each fuel channel are outlined in Figures 29 to 34. In all of the figures, both the average reductions (spherical markers) in transient powers as well as the minimum and the maximum range of reductions (dash markers) across the entire core are presented. The reductions in the transient powers are presented as absolute differences from the post-refuelling powers resulting for the regular NU fuel, which is the reference case of the study. Results for both the reference fuelling scheme and the entirely 8-bundle-shift fuelling mode are presented. Overall, the results indicated significant reductions in the immediate post-refuelling power ramps in the vicinity of fresh fuels when refuelling with added absorbers. The reductions in power were shown to dissipate quickly and become negligible after approximately 7 FPDs of irradiation for fuels containing only gadolinium, and 50 FPDs for fuels containing europium. The magnitudes of the reductions in power were shown to be greater when refuelling with larger quantities of added absorbers, and they are comparable to the projections of the WIMS lattice calculations.

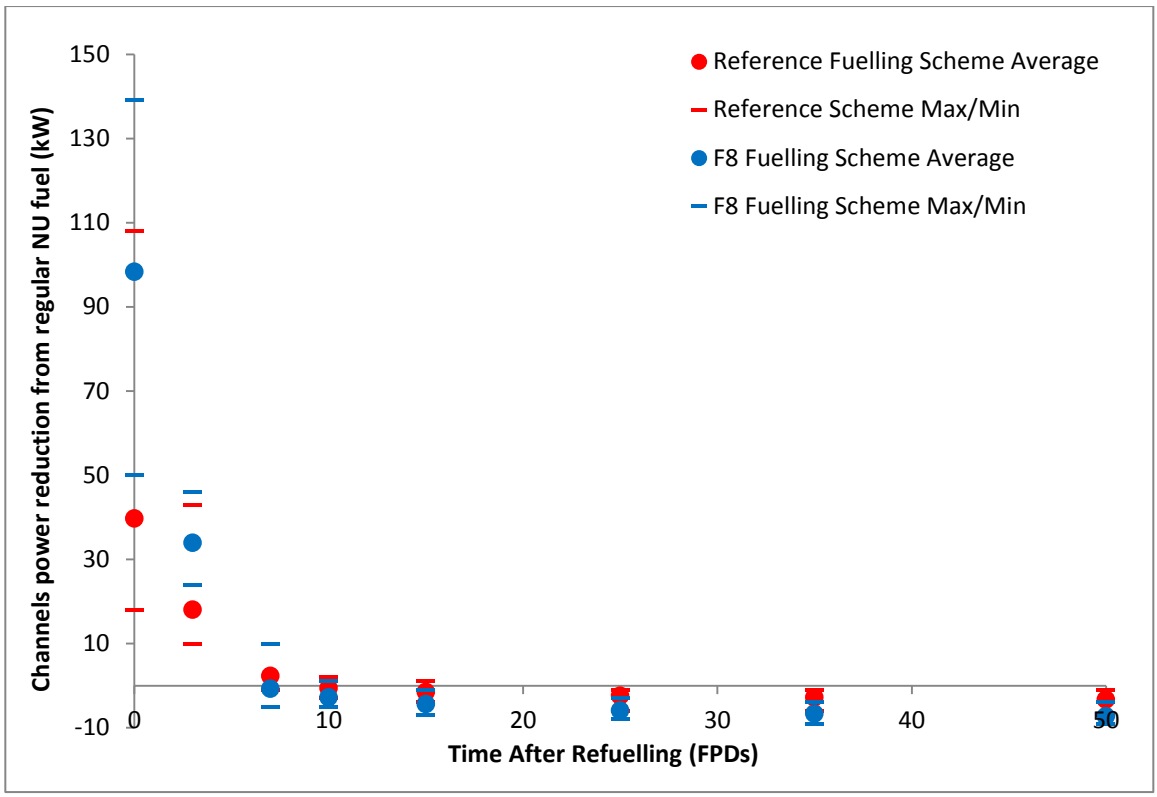


Figure 23. Average and min/max range (between all 480 channels) of reductions in post-refuelling channels powers incurred by refuelling with added absorbers (120 mg Gd_2O_3)

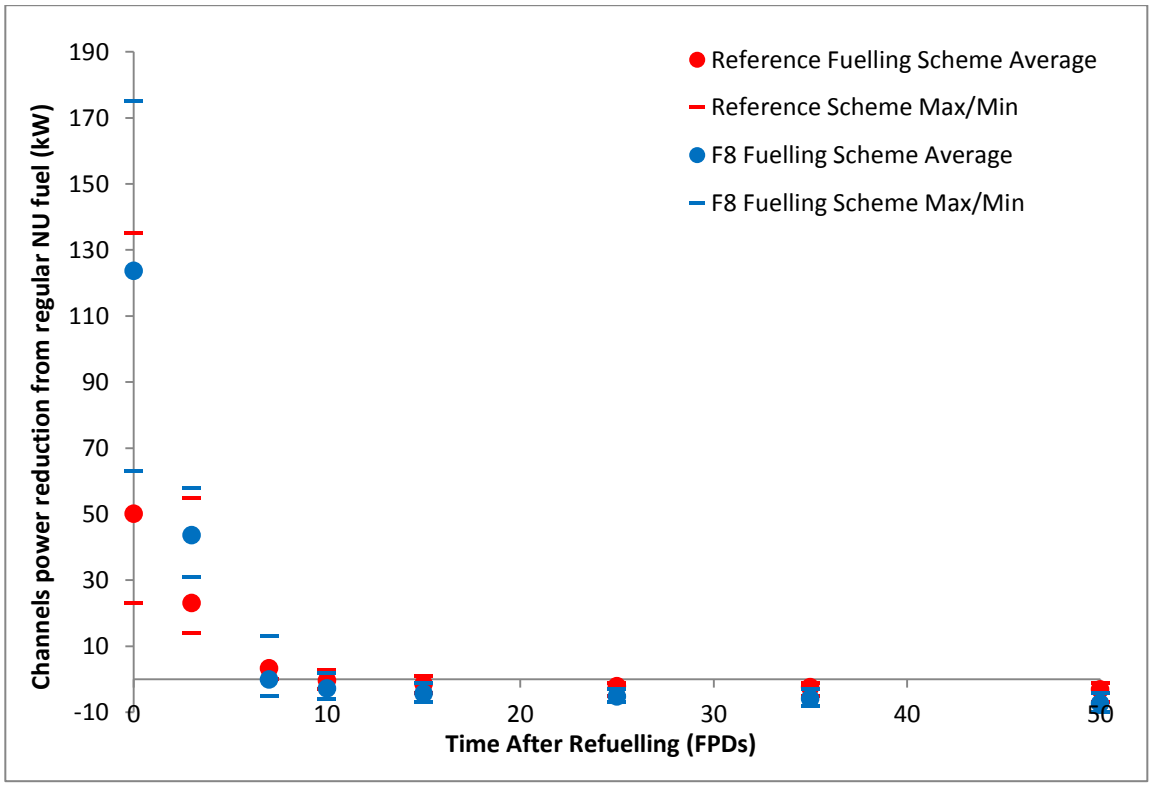


Figure 24. Average and min/max range (between all 480 channels) of reductions in channels powers incurred by refuelling with added absorbers (150 mg Gd_2O_3)

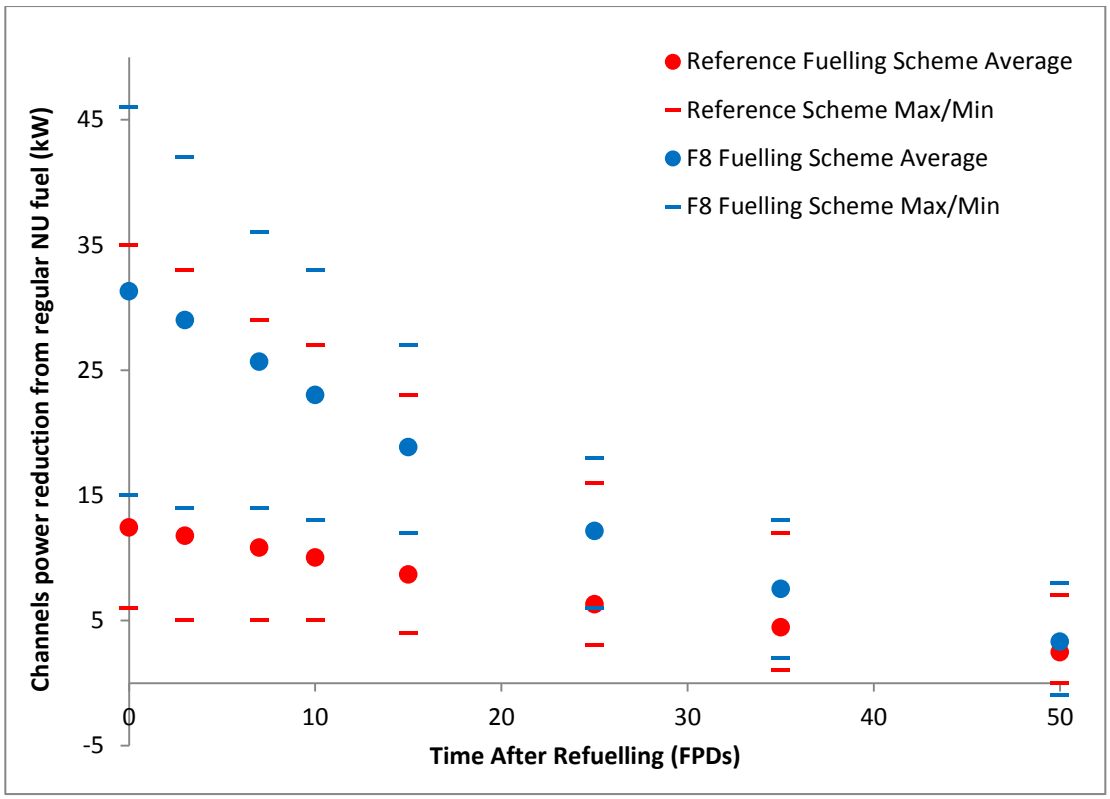


Figure 25. Average and min/max range (between all 480 channels) of reductions in channels powers incurred by refuelling with added absorbers (150 mg Eu_2O_3)

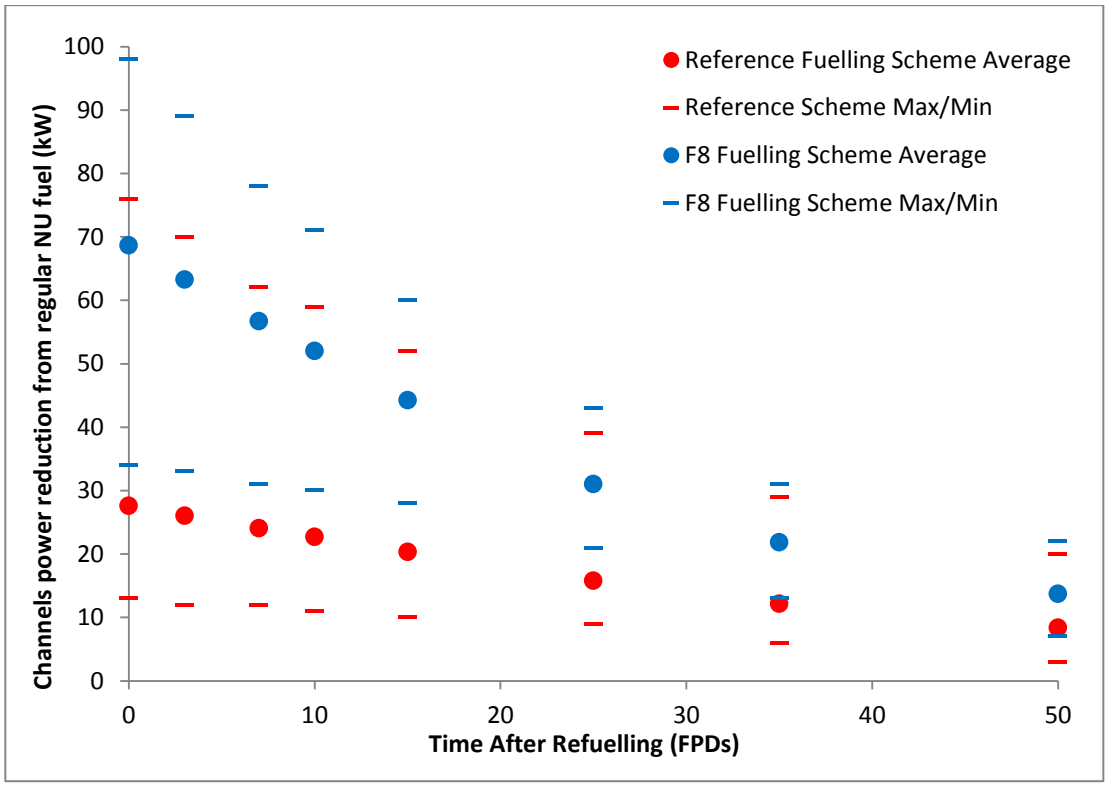


Figure 26. Average and min/max range (between all 480 channels) of reductions in channels powers incurred by refuelling with added absorbers (300 mg Eu_2O_3)

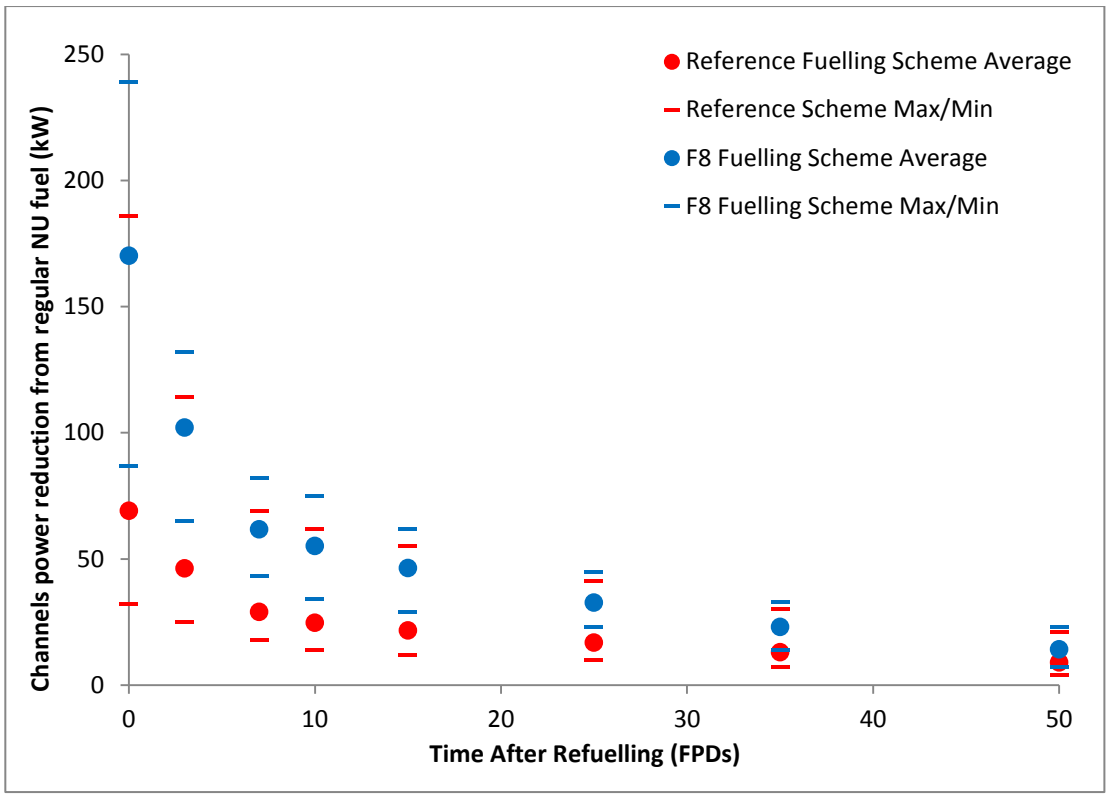


Figure 27. Average and min/max range (between all 480 channels) of reductions in channels powers incurred by refuelling with added absorbers (120 mg Gd_2O_3 + 300 mg Eu_2O_3)

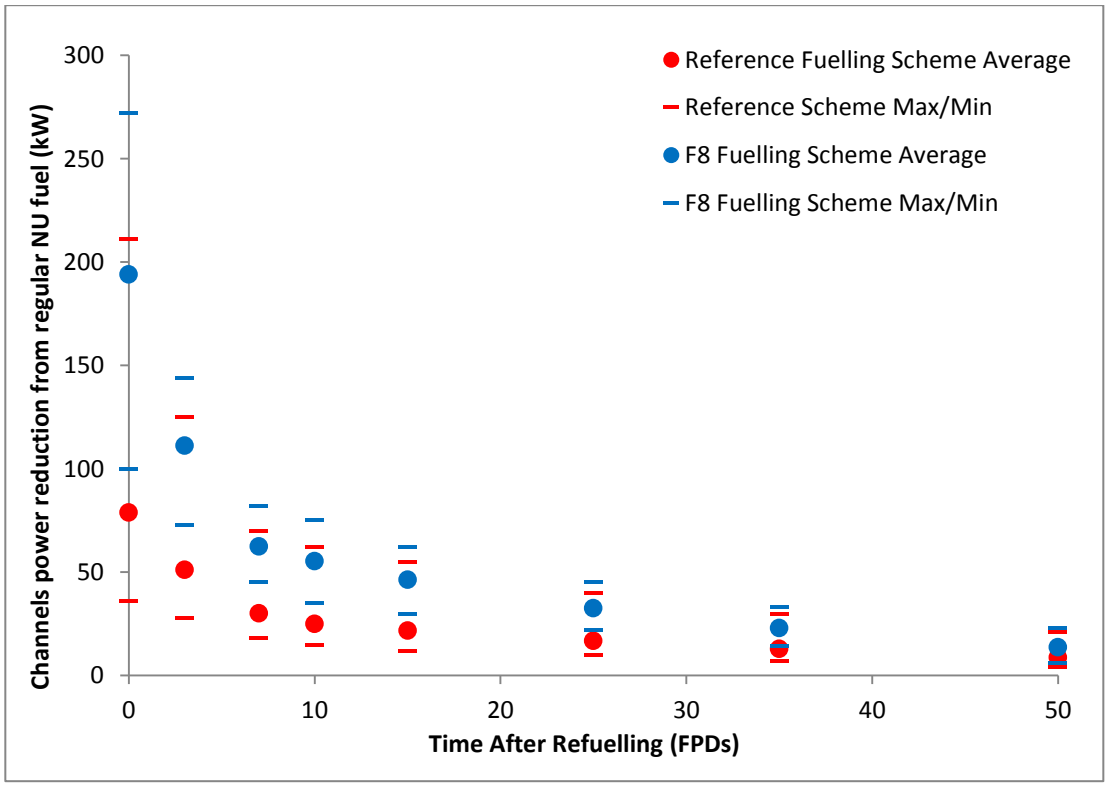


Figure 28. Average and min/max range (between all 480 channels) of reductions in channels powers incurred by refuelling with added absorbers (150 mg Gd_2O_3 + 300 mg Eu_2O_3)

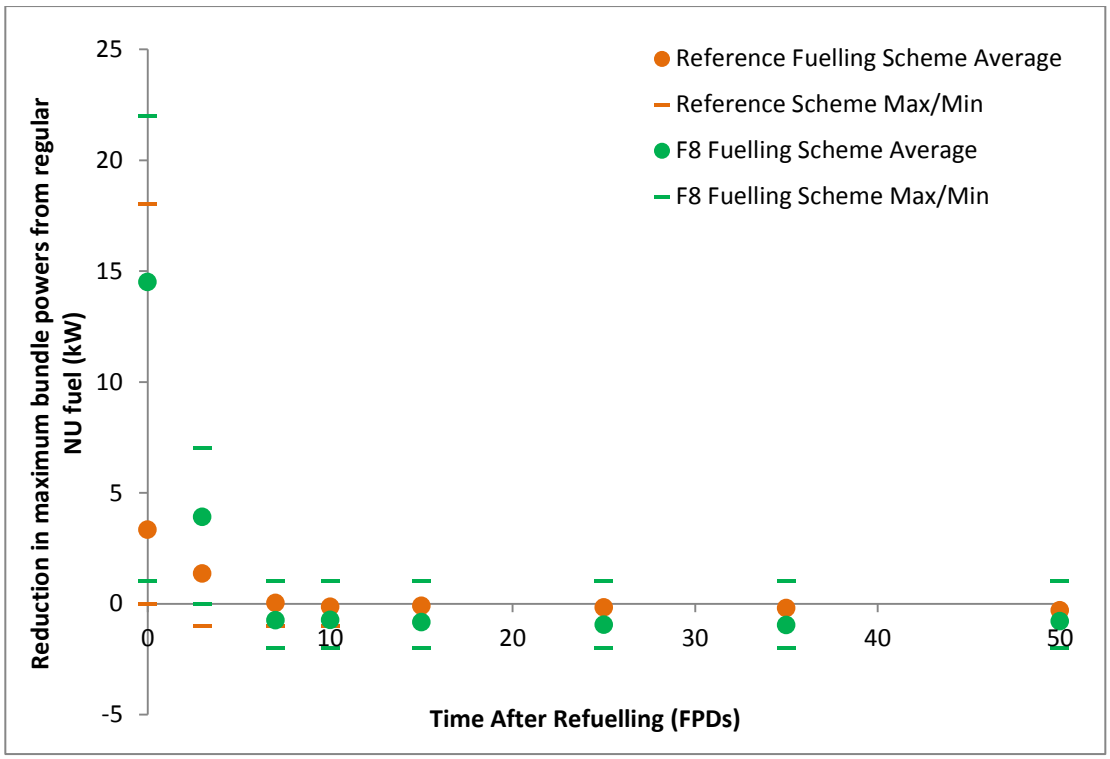


Figure 29. Average and min/max range (between all 480 channels) of reductions in the powers of the highest-powered fuel bundles within each fuel channel incurred by refuelling with added absorbers (120 mg Gd_2O_3)

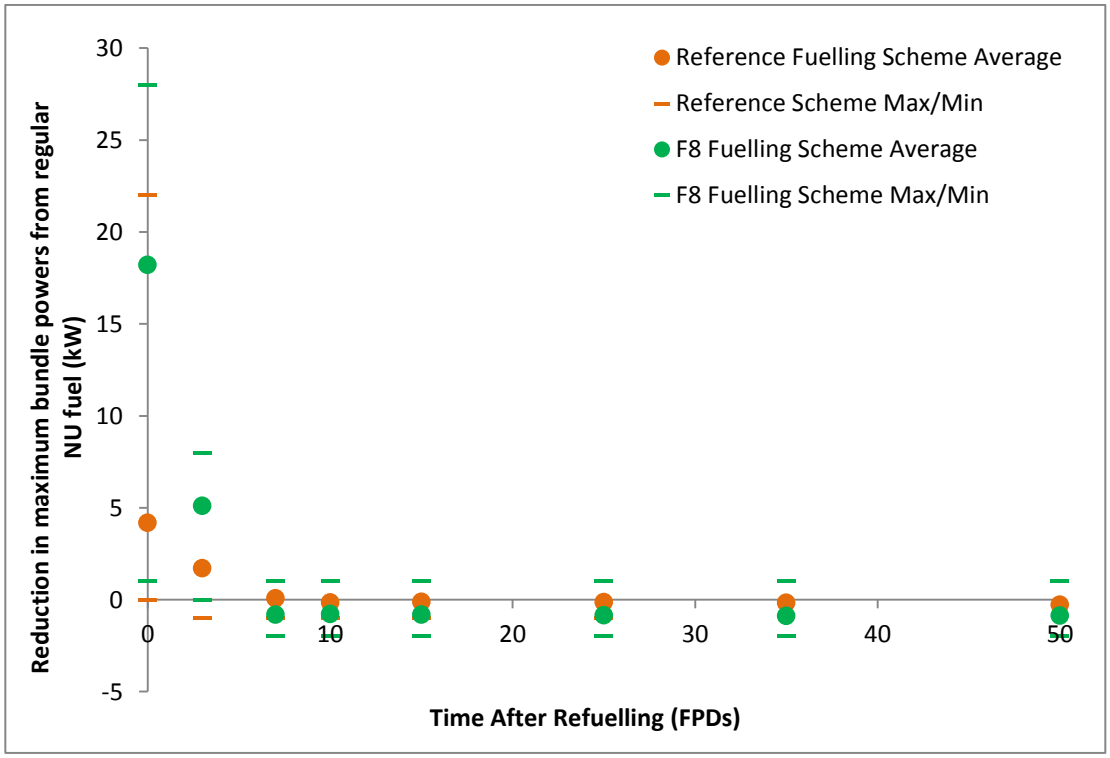


Figure 30. Average and min/max range (between all 480 channels) of reductions in the powers of the highest-powered fuel bundles within each fuel channel incurred by refuelling with added absorbers (150 mg Gd_2O_3)

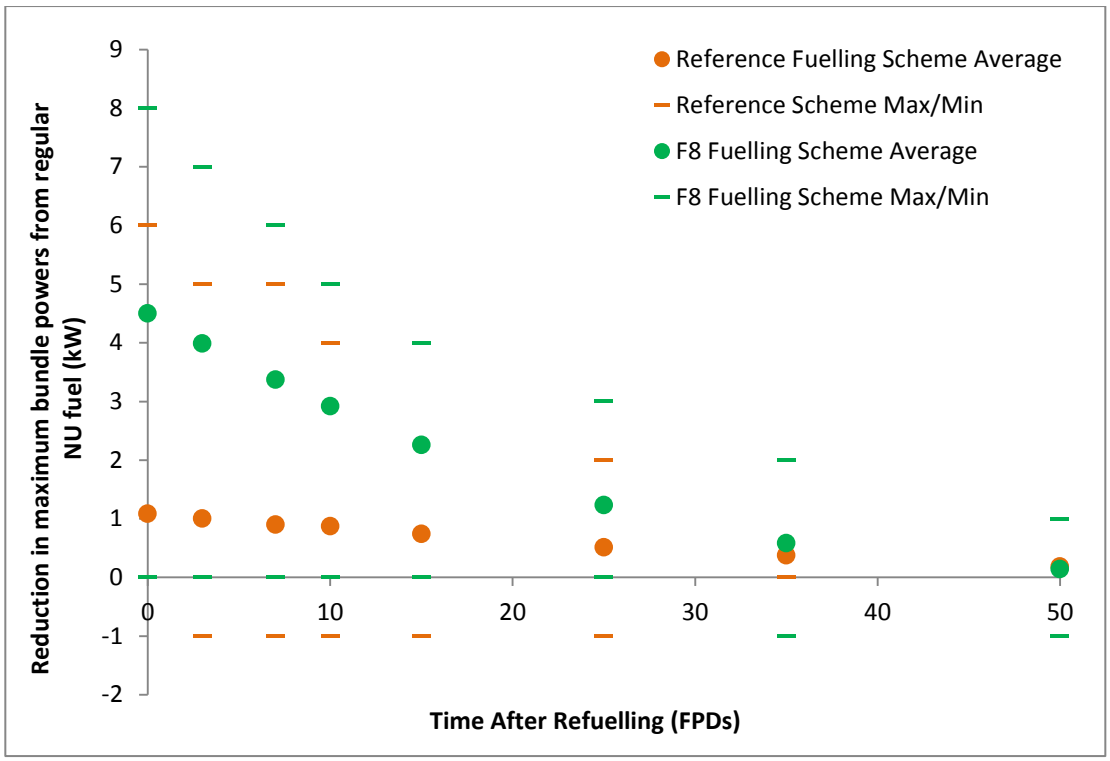


Figure 31. Average and min/max range (between all 480 channels) of reductions in the powers of the highest-powered fuel bundles within each fuel channel incurred by refuelling with added absorbers (150 mg Eu_2O_3)

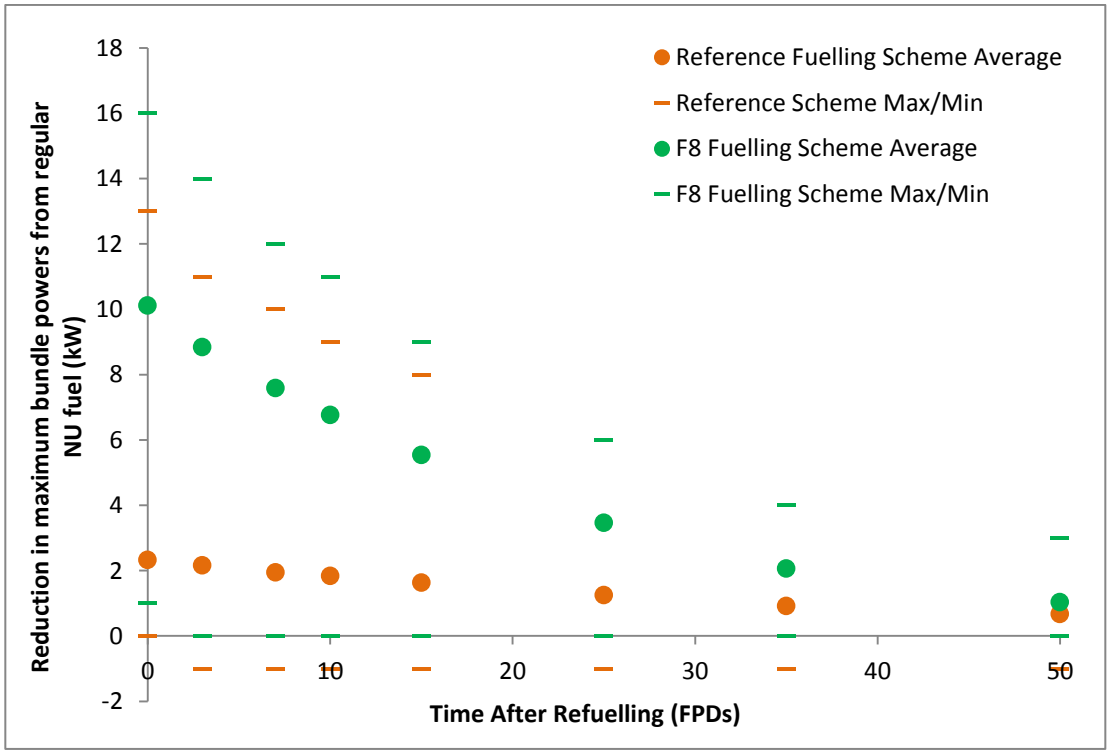


Figure 32. Average and min/max range (between all 480 channels) of reductions in the powers of the highest-powered fuel bundles within each fuel channel incurred by refuelling with added absorbers (300 mg Eu_2O_3)

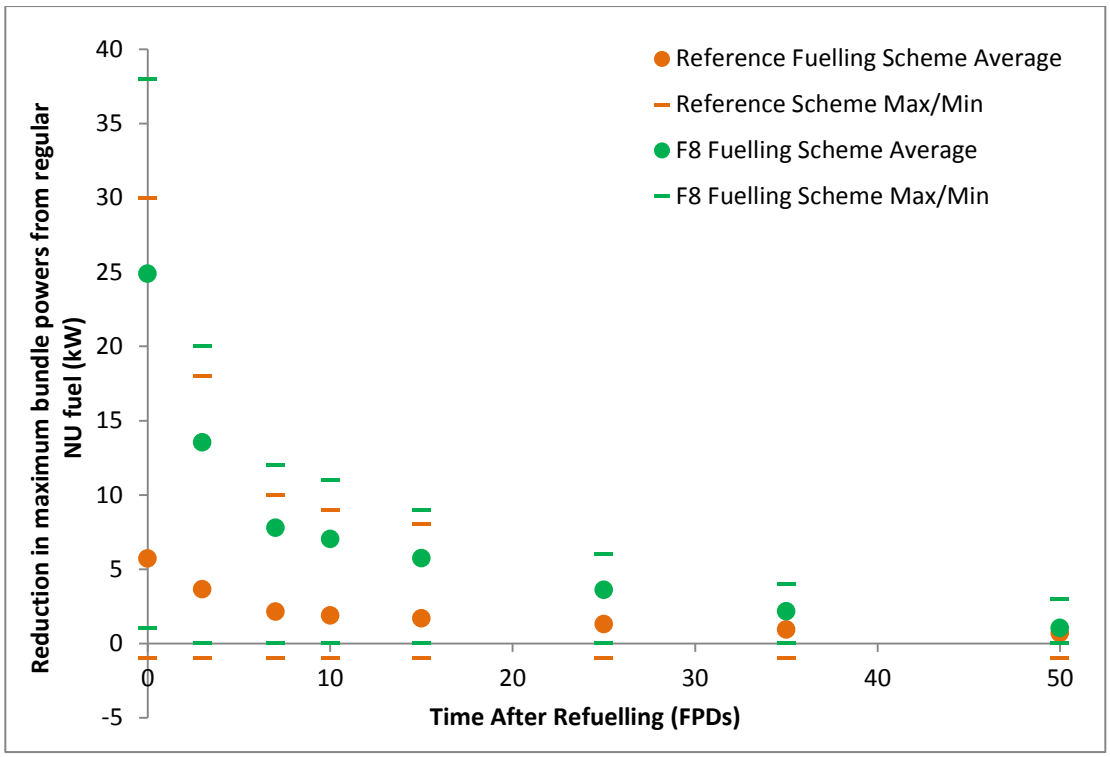


Figure 33. Average and min/max range (between all 480 channels) of reductions in the powers of the highest-powered fuel bundles within each fuel channel incurred by refuelling with added absorbers (120 mg Gd_2O_3 & 300 mg Eu_2O_3)

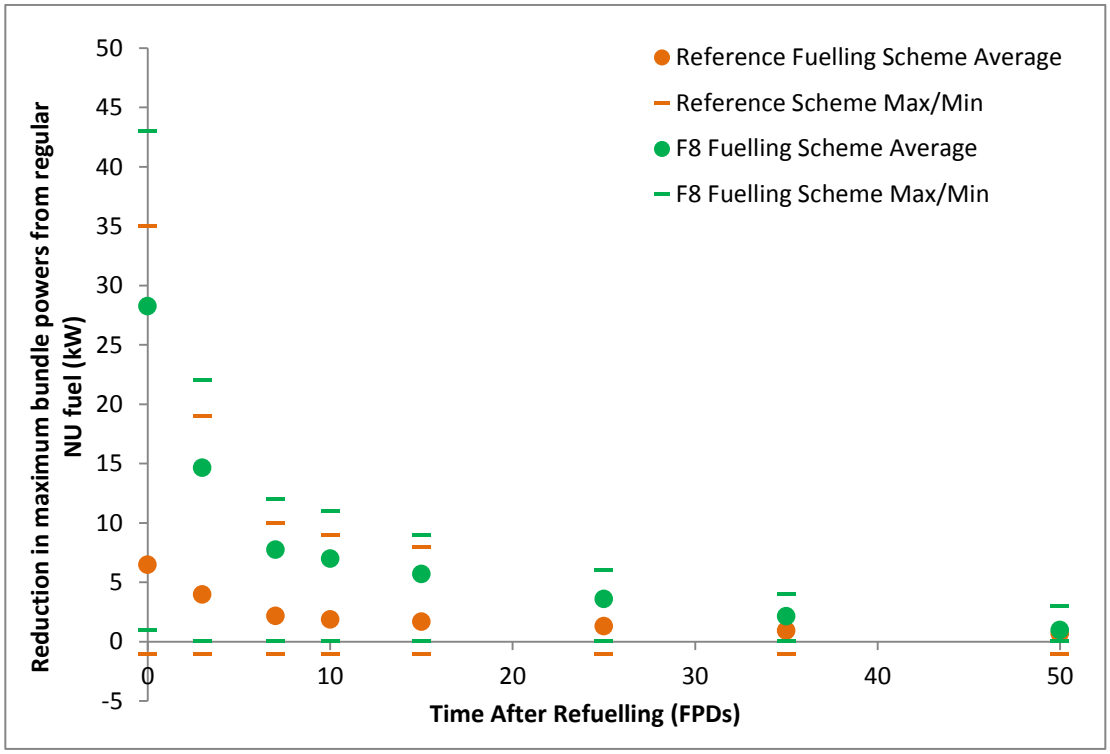


Figure 34. Average and min/max range (between all 480 channels) of reductions in the powers of the highest-powered fuel bundles within each fuel channel incurred by refuelling with added absorbers (150 mg Gd_2O_3 & 300 mg Eu_2O_3)

6.2 Core-Following Simulations

The results of the core-following simulations for the regular NU fuel and the BNAF containing the proposed quantities of burnable absorbers are outlined in Figures 35 to 58. The results for both the reference fuelling scheme and the entirely 8-bundle-shift fuelling mode are presented, which include the rate of bulk reactivity decline, the average fill levels of LZCs, the maximum channel power in the core, the maximum bundle power in the core, the radial form factor of the core, and the highest channel power peaking factor in the core. Each of the figures are presented in order to illustrate the changes in the values of respective performance parameters as the core transitions from being refuelled with regular NU fuels to the BNAF containing the proposed quantity of burnable absorbers. The refuelling algorithm ensured the values of k_{eff} during all simulations remained at approximately 1.000.

Figures 35 and 36 respectively present the values of the instantaneous rate of reactivity decline of the core during simulations for the regular NU fuel and the BNAF containing the proposed quantities of burnable absorbers. Figures 37 and 38 present the same results for the entirely 8-bundle-shift fuelling mode.

Figures 39 and 40 present the values of the average liquid zone controllers' percentage fill levels in the core during simulations for the regular NU fuel and the BNAF containing the proposed quantities of burnable absorbers. Figures 41 and 42 present the same results for the entirely 8-bundle-shift fuelling mode.

Figures 43 and 44 present the values of the maximum channel power in the core during simulations for the regular NU fuel and the BNAF containing the proposed quantities of burnable absorbers. Figures 45 and 46 present the same results for the entirely 8-bundle-shift fuelling mode.

Figures 47 and 48 present the values of the maximum bundle power in the core during simulations for the regular NU fuel and the BNAF containing the proposed

quantities of burnable absorbers. Figures 49 and 50 present the same results for the entirely 8-bundle-shift fuelling mode.

Figures 51 and 52 present the values of the radial form factor of the core during simulations for the regular NU fuel and the BNAF containing the proposed quantities of burnable absorbers. Figures 53 and 54 present the same results for the entirely 8-bundle-shift fuelling mode.

Figures 55 and 56 present the values of the maximum CPPF in the core during simulations for the regular NU fuel and the BNAF containing the proposed quantities of burnable absorbers. Figures 57 and 58 present the same results for the entirely 8-bundle-shift fuelling mode.

In addition to the plots of performance parameters shown in Figures 35 to 58 (for only the NU fuel and the proposed BNAF), the average (of 400 FPDs) values of each performance parameter and their range of values (for channel and bundle powers) for all BNAFs simulated in the study are presented in Tables 2 to 7. The tables also present the resulting average discharge burnup of each fuel type. The indicated results are used to determine and compare the average reductions in the peak powers in the core and improvements for the shape of the distribution of flux/power in the core incurred by transitioning the core from regular NU fuel to fuels with added absorbers.

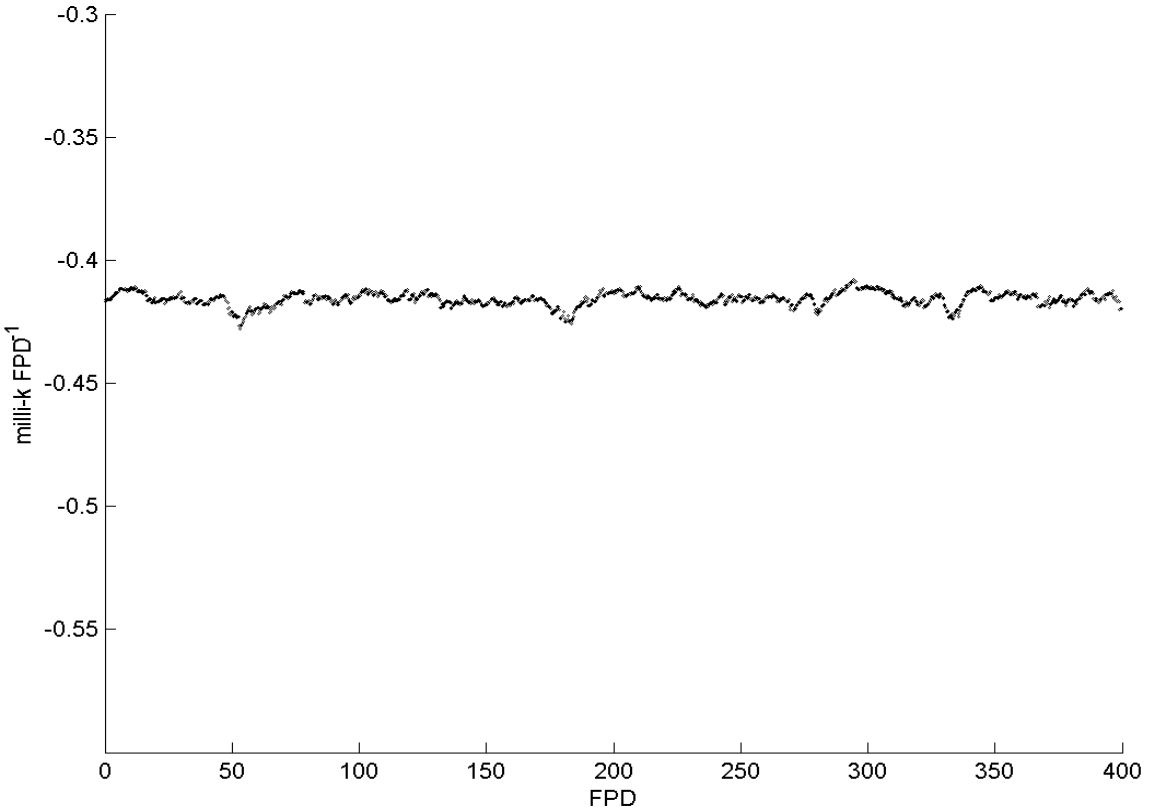


Figure 35. Rate of reactivity decline of the core during core-following using the regular NU fuel, for the reference fuelling scheme

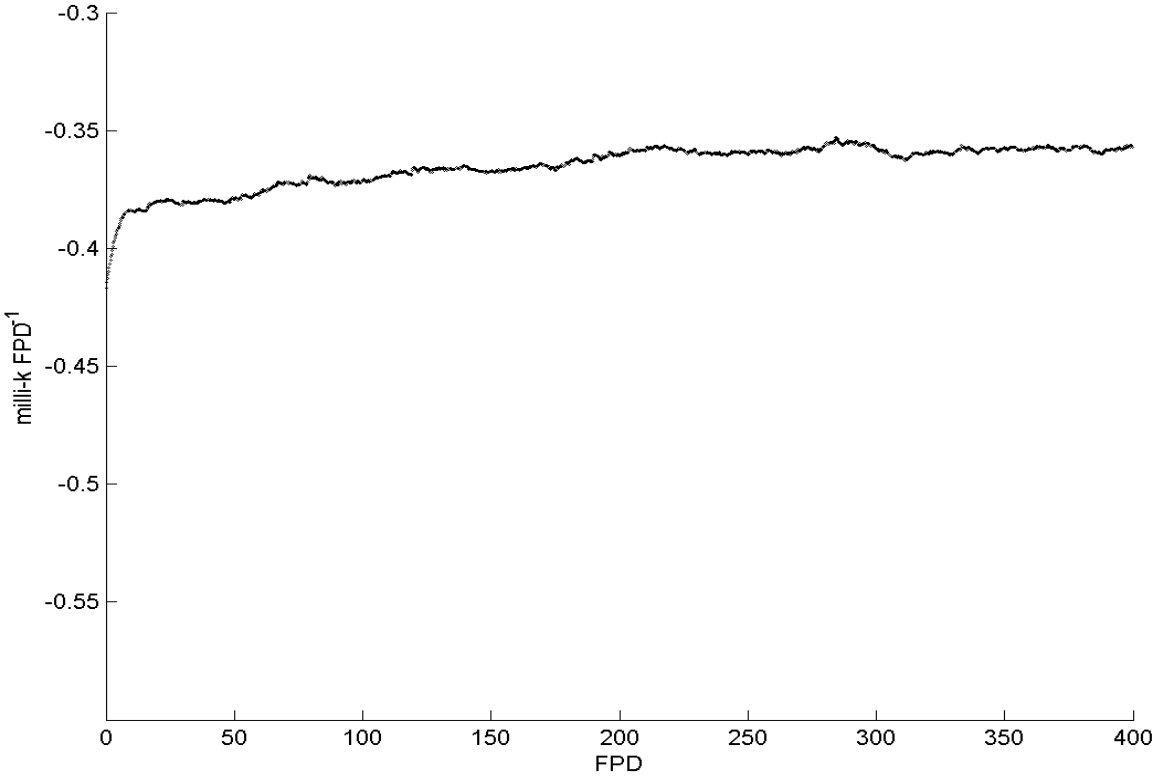


Figure 36. Rate of reactivity decline of the core during core-following using BNAF containing 150 mg Gd_2O_3 & 300 mg Eu_2O_3, for the reference fuelling scheme

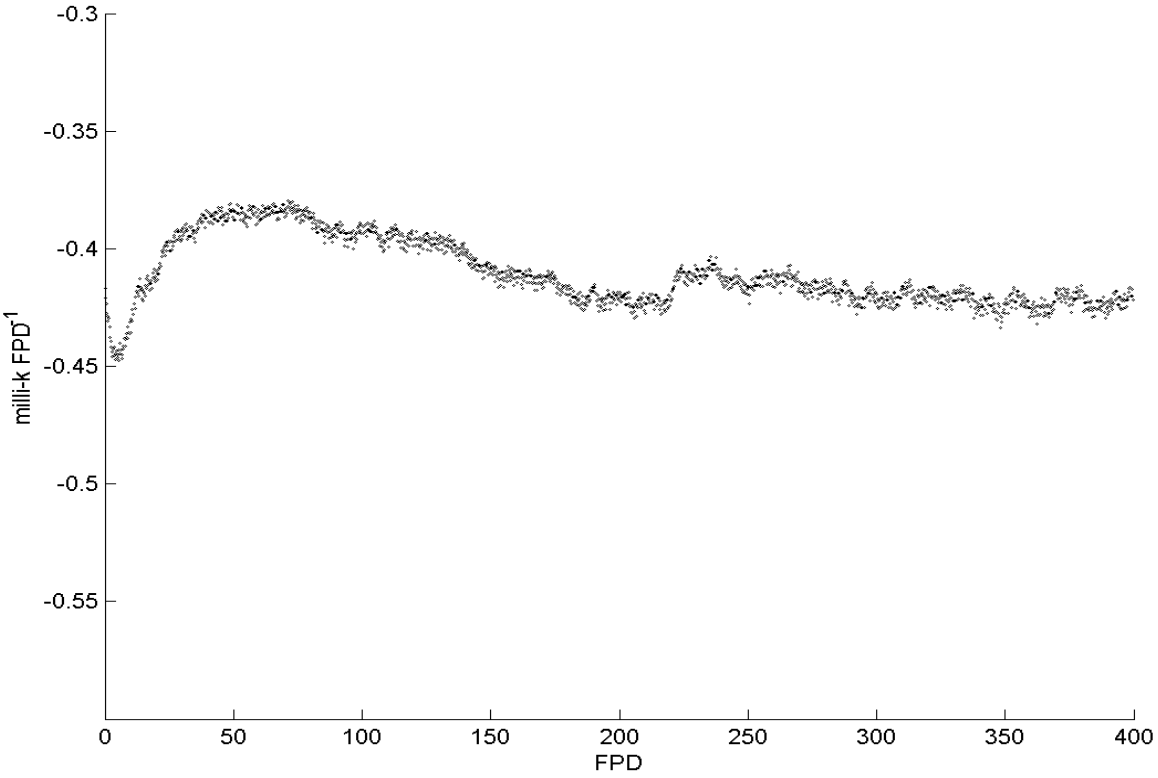


Figure 37. Rate of reactivity decline of the core during core-following using the regular NU fuel, for the F8 fuelling scheme

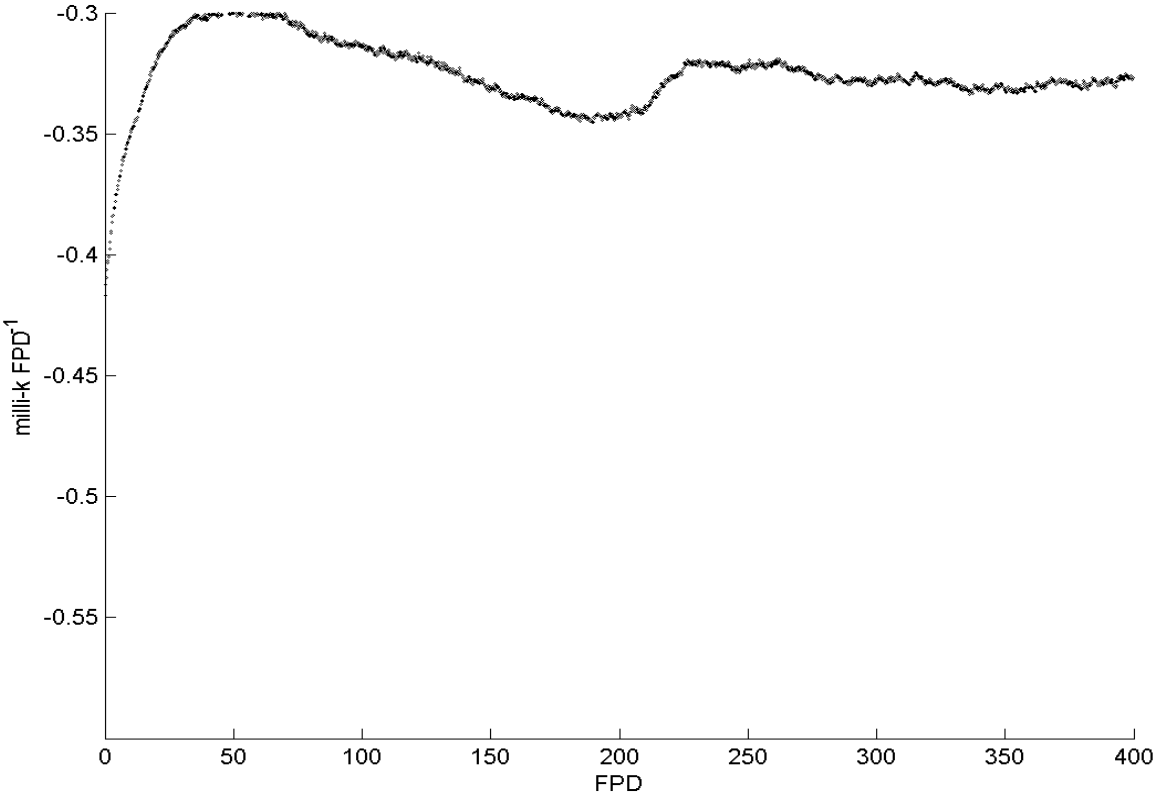


Figure 38. Rate of reactivity decline of the core during core-following using BNAF containing 150 mg Gd_2O_3 & 300 mg Eu_2O_3, for the F8 fuelling scheme

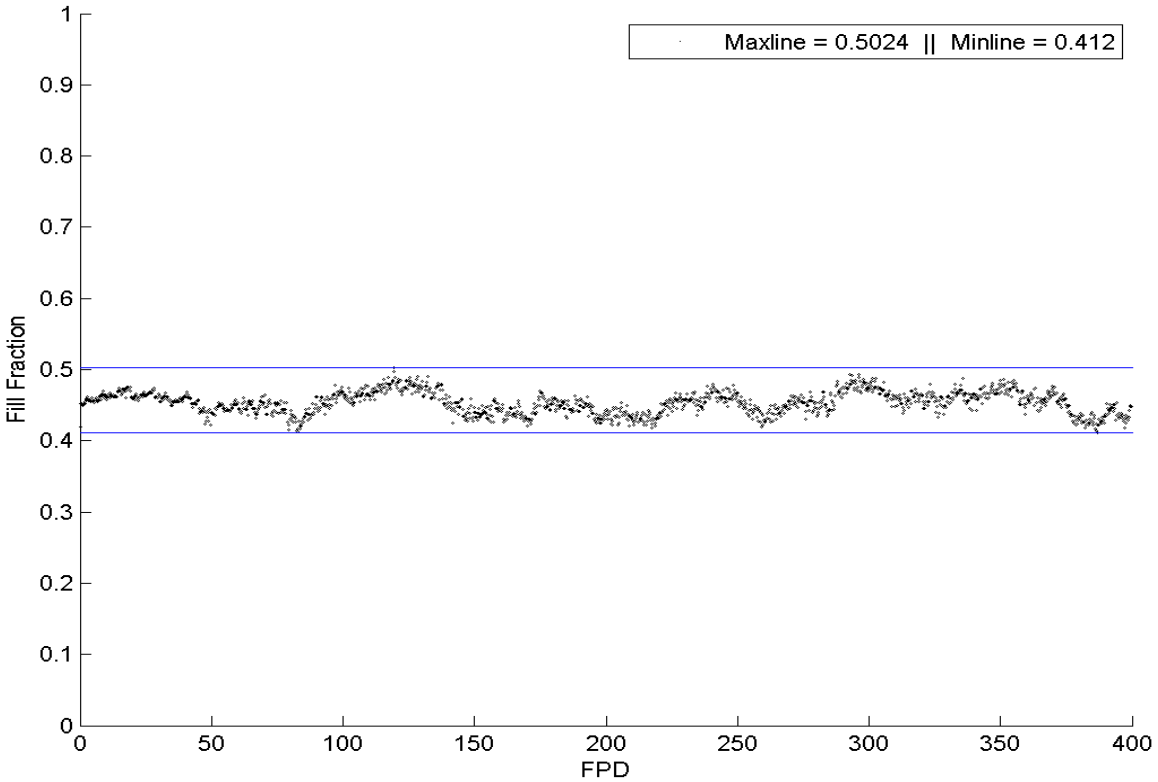


Figure 39. Average fill level of liquid zone controllers in the core during core-following using the regular NU fuel, for the reference fuelling scheme

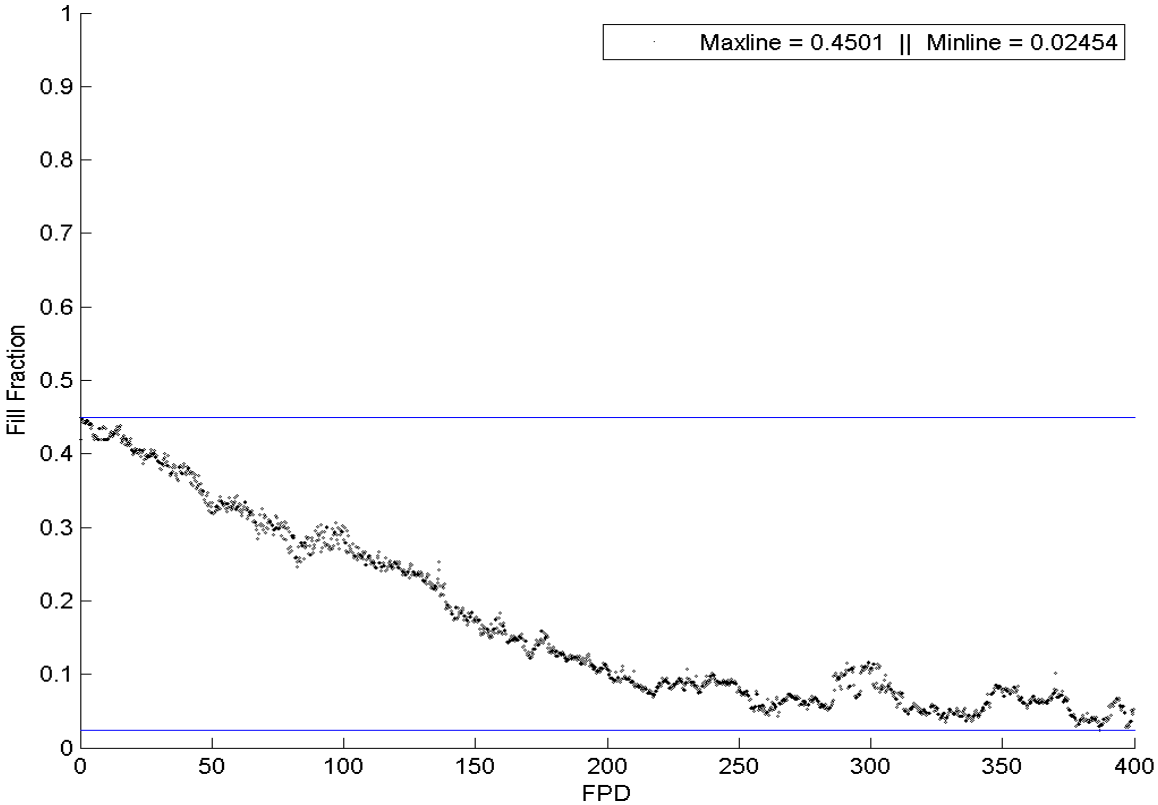


Figure 40. Average fill level of liquid zone controllers in the core during core-following using BNAF containing 150 mg Gd_2O_3 & 300 mg Eu_2O_3, for the reference fuelling scheme

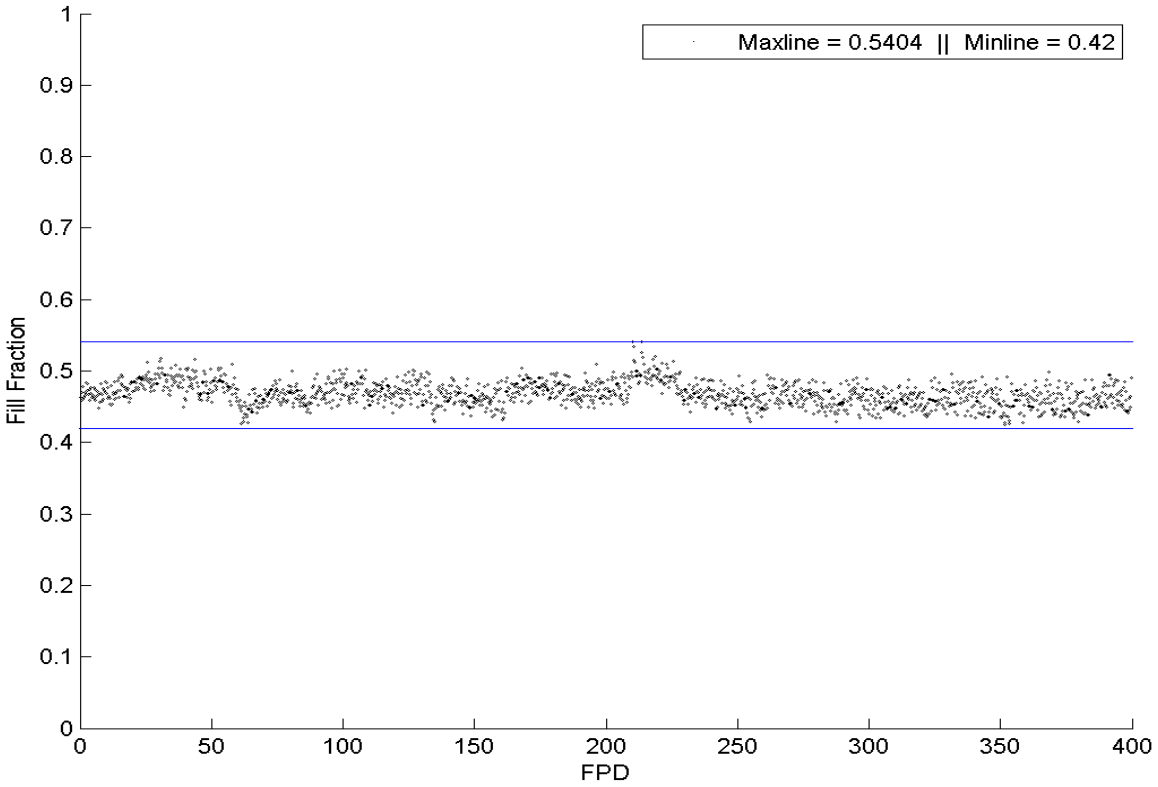


Figure 41. Average fill level of liquid zone controllers in the core during core-following using the regular NU fuel, for the F8 fuelling scheme

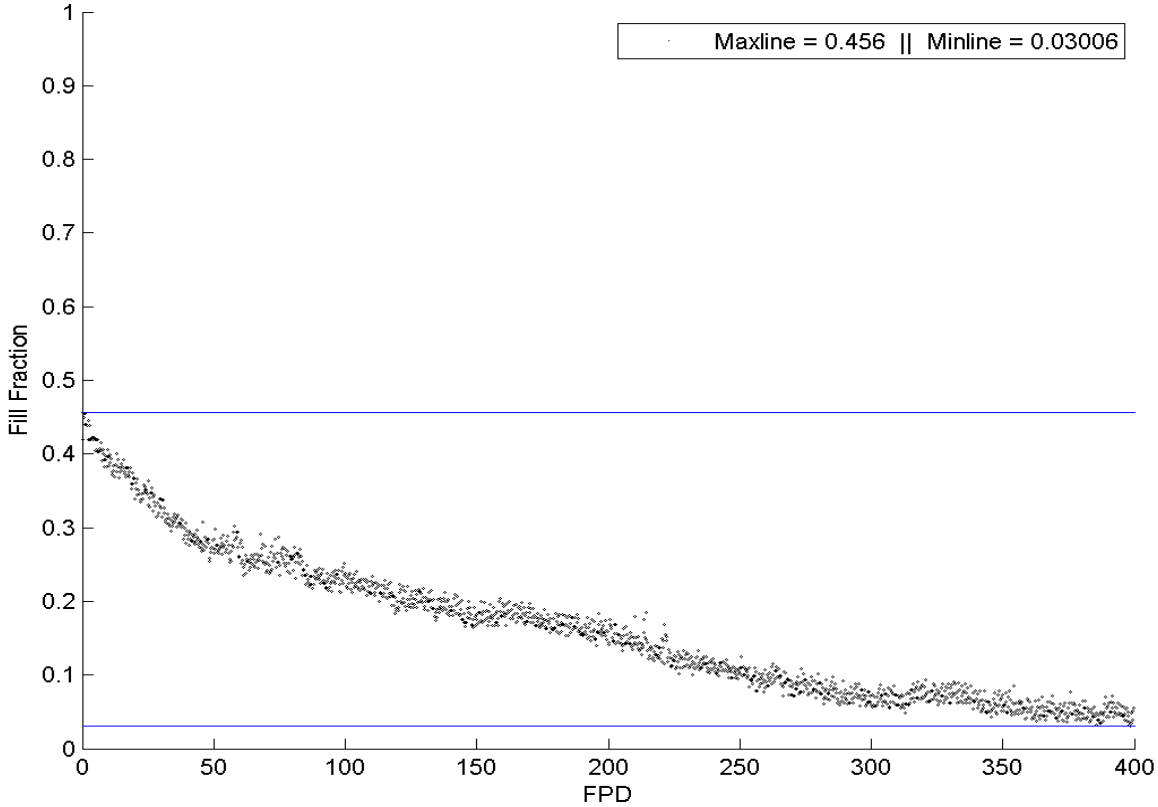


Figure 42. Average fill level of liquid zone controllers in the core during core-following using BNAF containing 150 mg Gd_2O_3 & 300 mg Eu_2O_3, for the F8 fuelling scheme

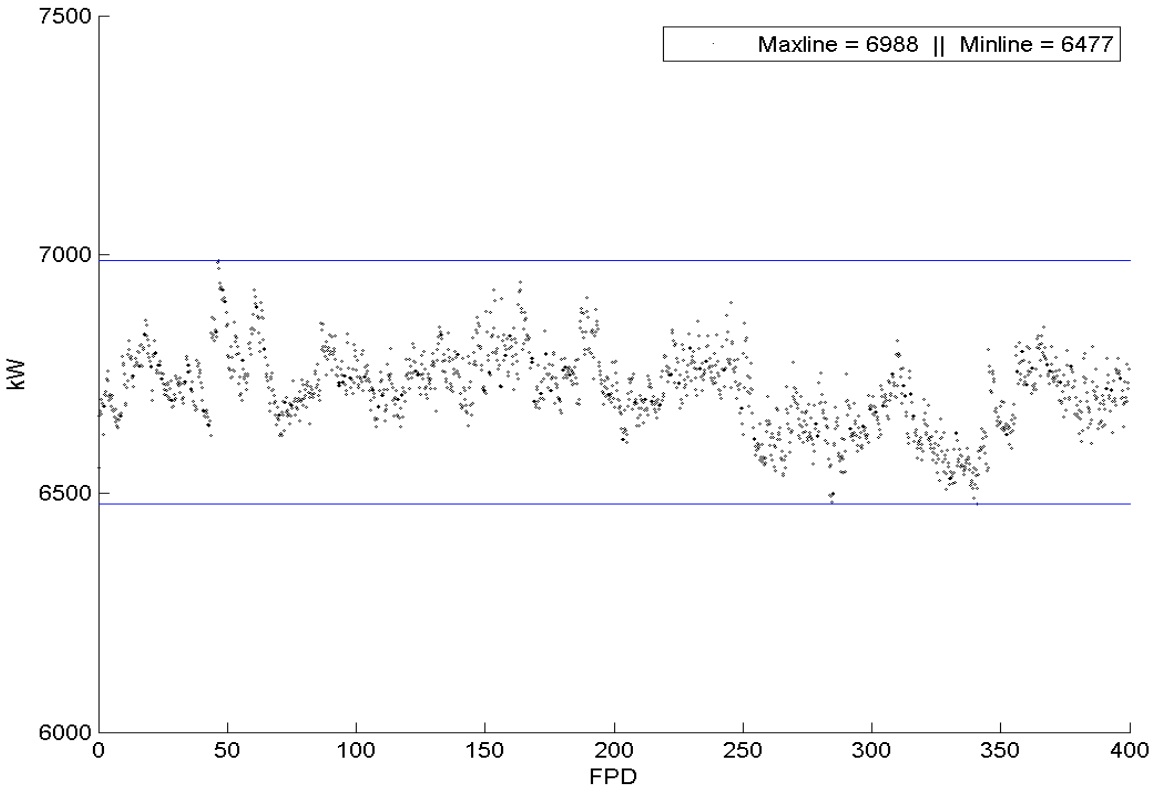


Figure 43. Maximum channel power in the core during core-following using the regular NU fuel, for the reference fuelling scheme

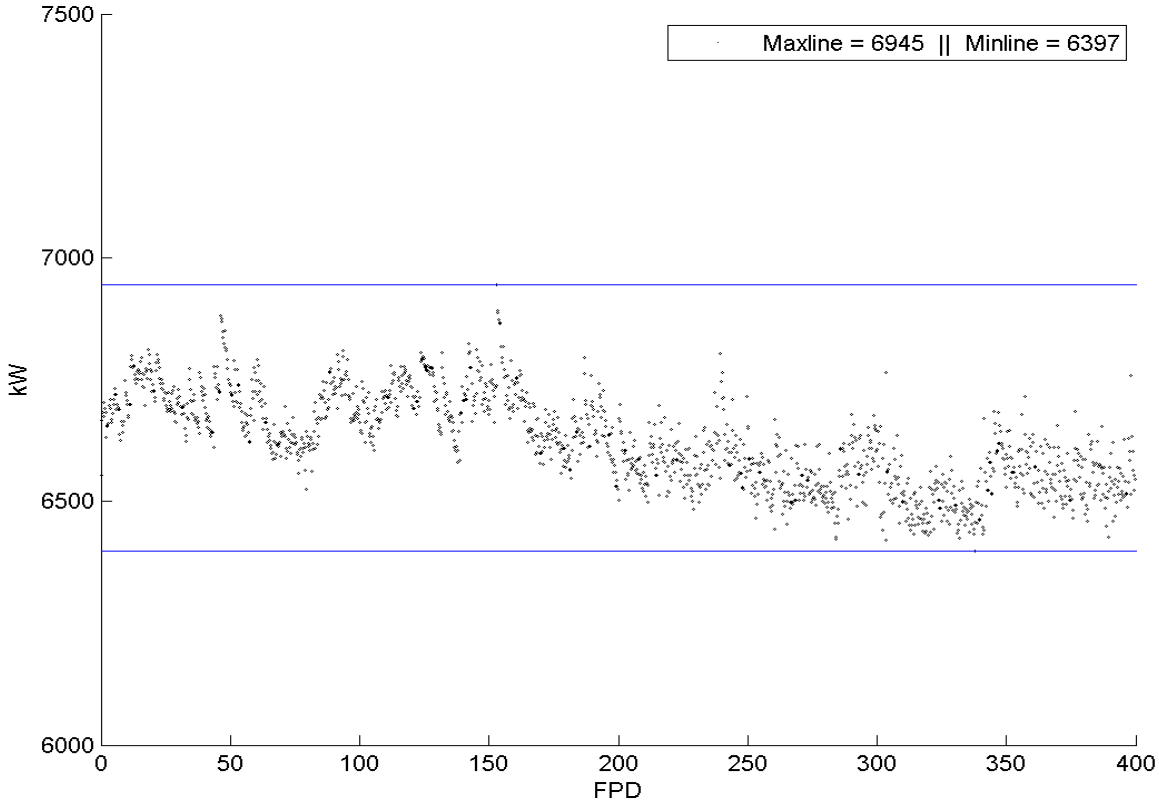


Figure 44. Maximum channel power in the core during core-following using BNAF containing 150 mg Gd_2O_3 & 300 mg Eu_2O_3, for the reference fuelling scheme

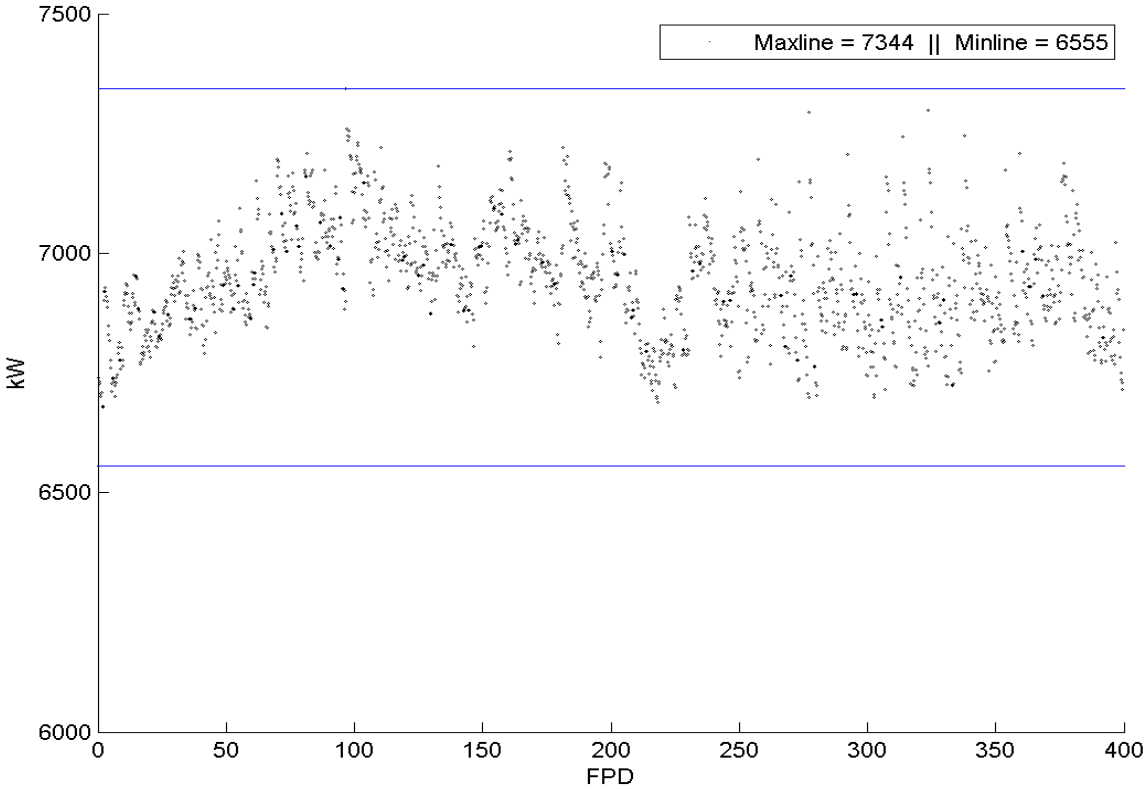


Figure 45. Maximum channel power in the core during core-following using the regular NU fuel, for the F8 fuelling scheme

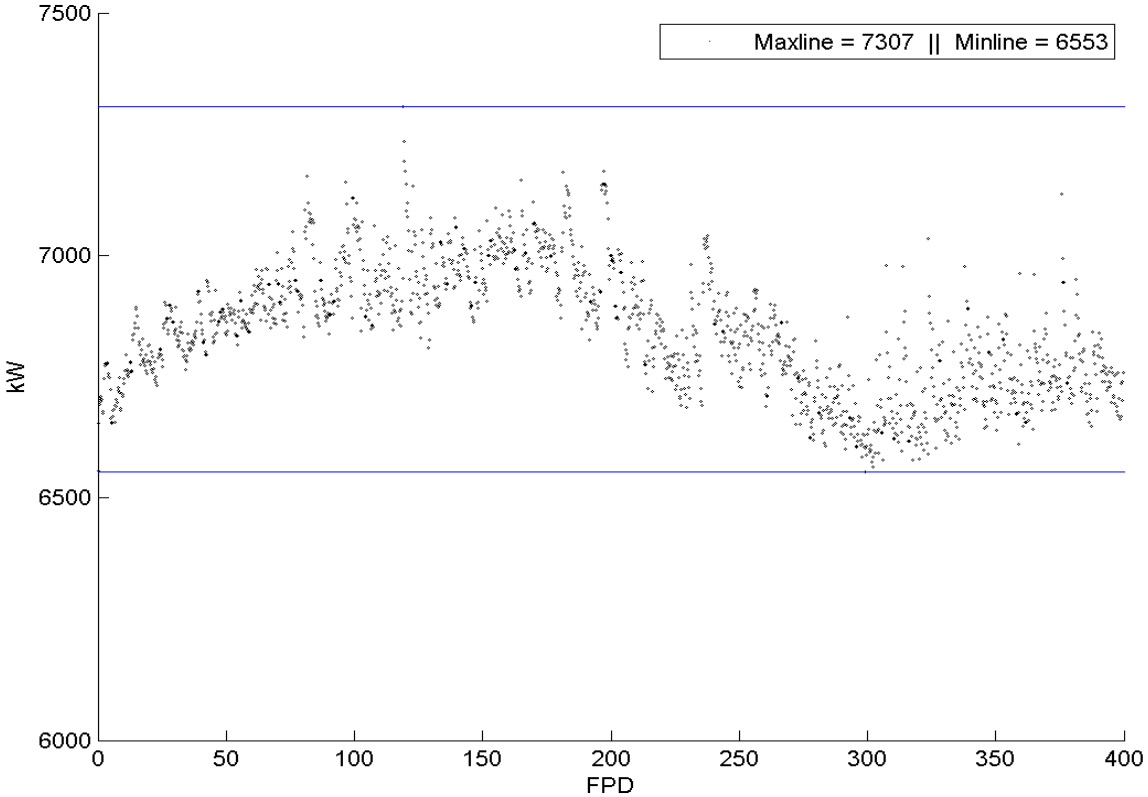


Figure 46. Maximum channel power in the core during core-following using BNAF containing 150 mg Gd_2O_3 & 300 mg Eu_2O_3, for the F8 fuelling scheme

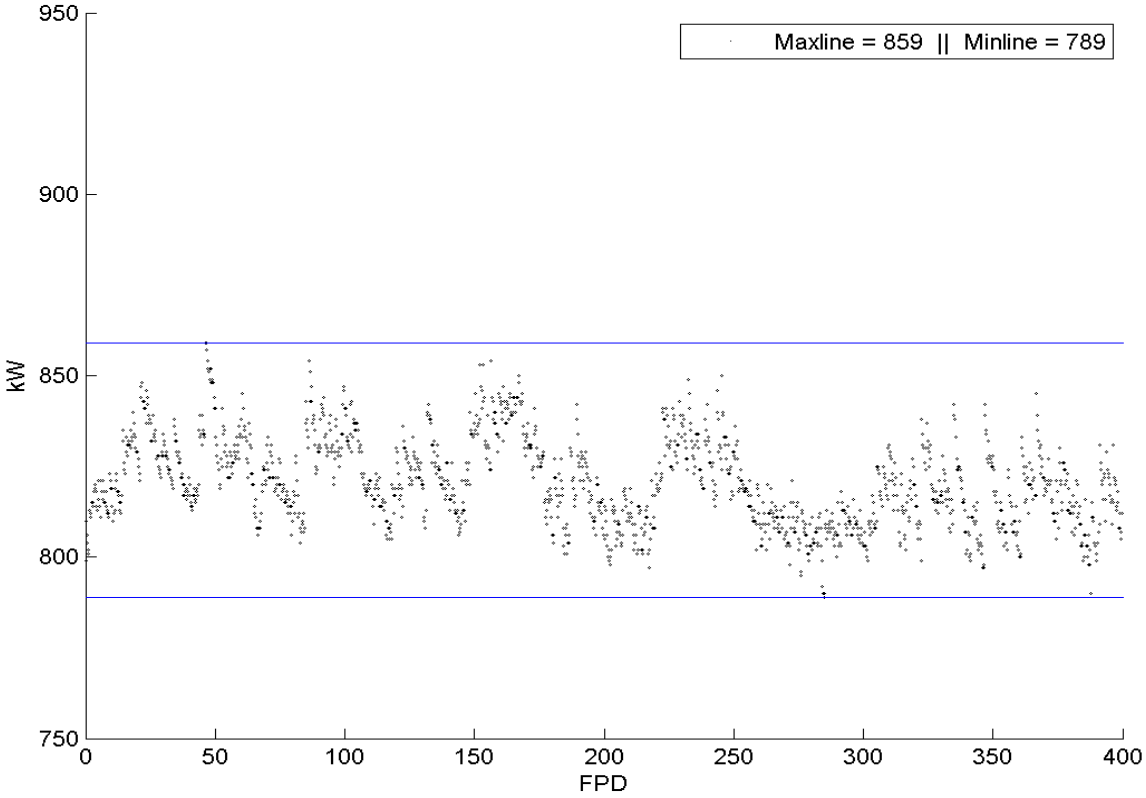


Figure 47. Maximum bundle power in the core during core-following using the regular NU fuel, for the reference fuelling scheme

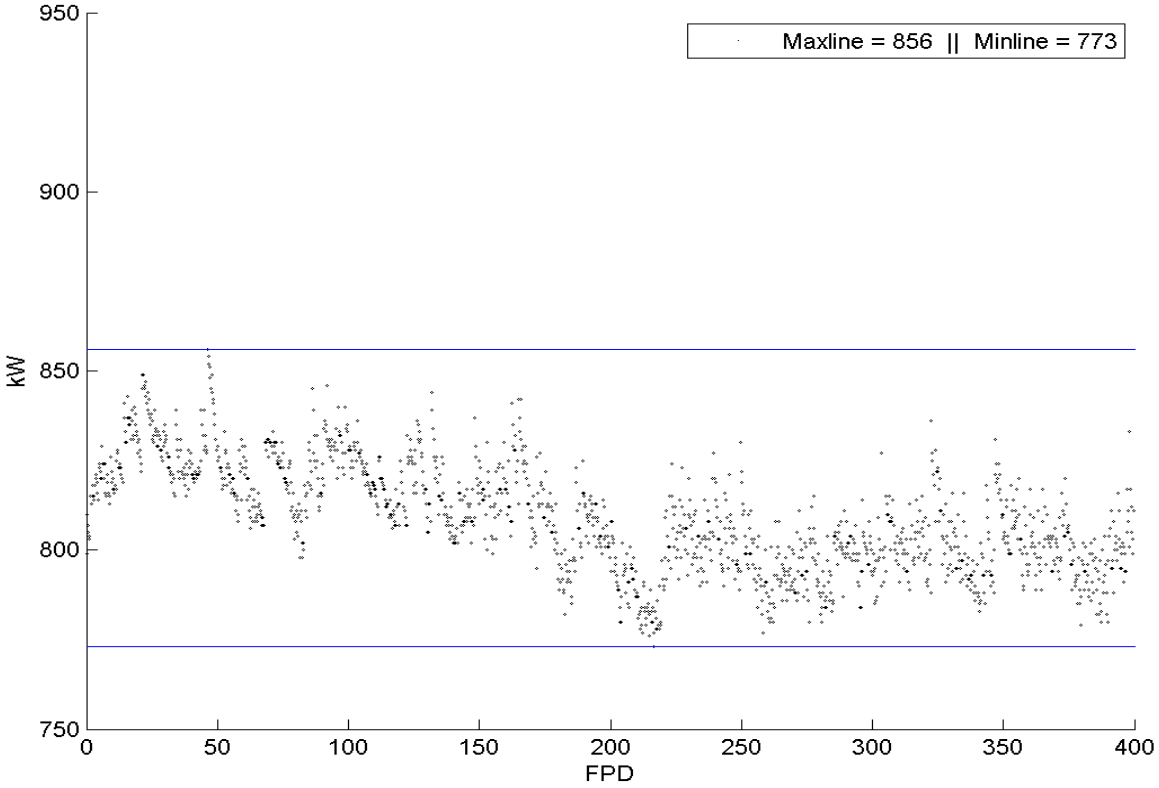


Figure 48. Maximum bundle power in the core during core-following using BNAF containing 150 mg Gd_2O_3 & 300 mg Eu_2O_3, for the reference fuelling scheme

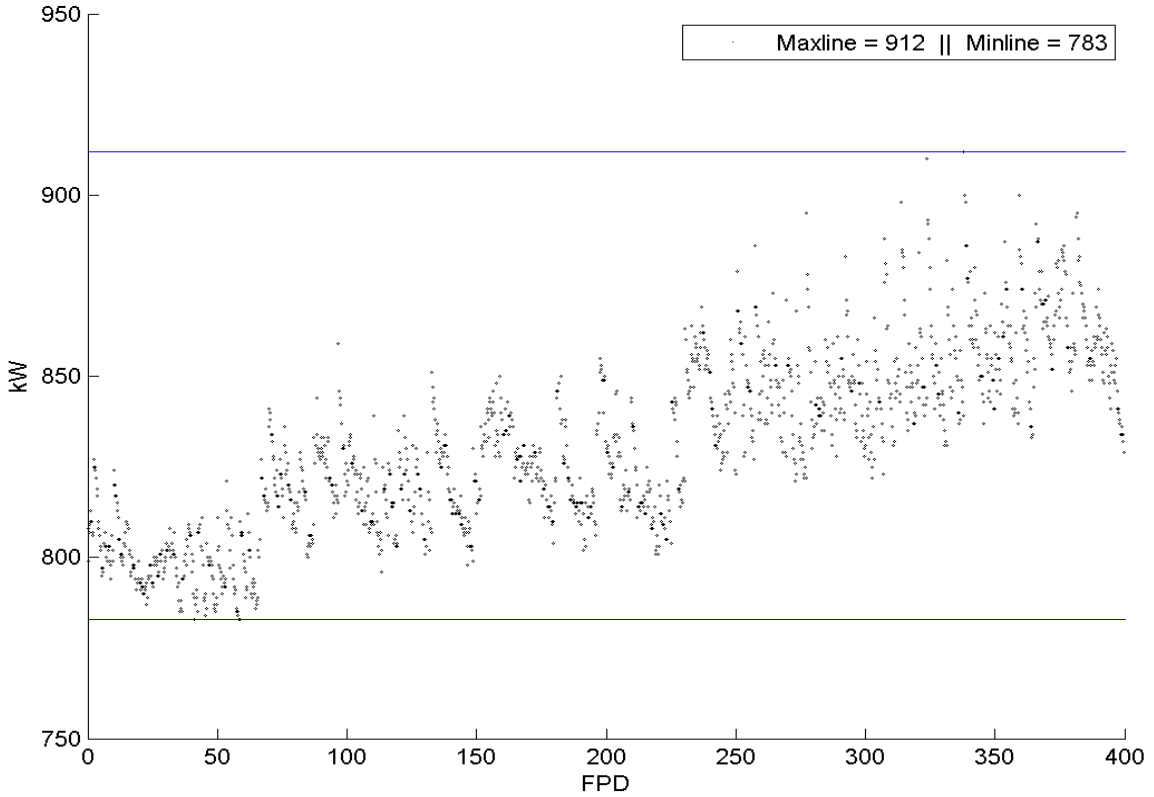


Figure 49. Maximum bundle power in the core during core-following using the regular NU fuel, for the F8 fuelling scheme

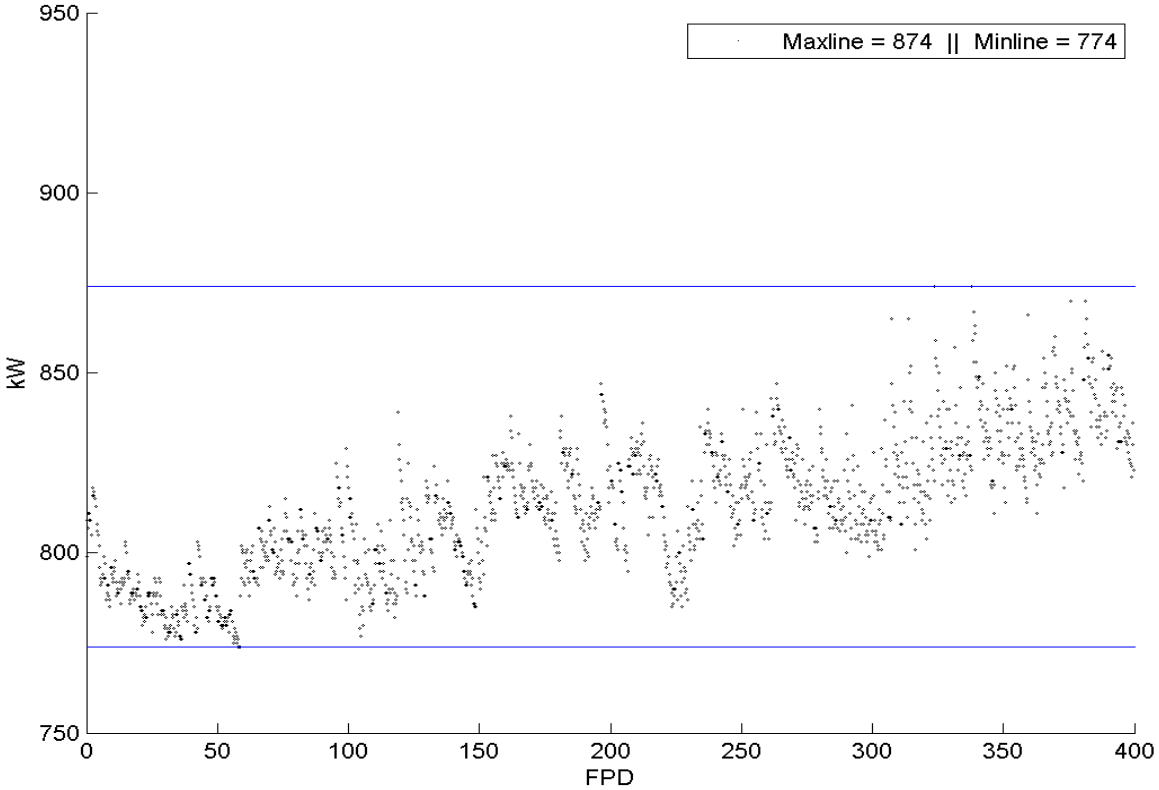


Figure 50. Maximum bundle power in the core during core-following using BNAF containing 150 mg Gd_2O_3 & 300 mg Eu_2O_3, for the F8 fuelling scheme

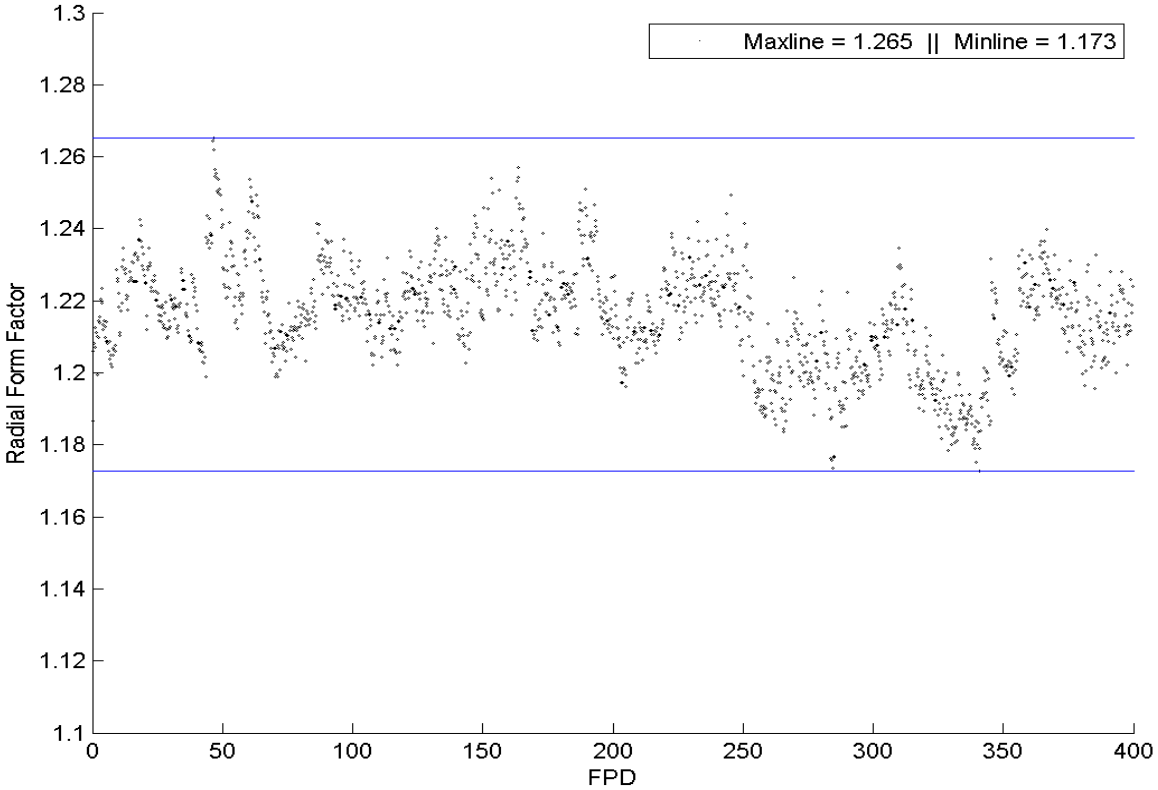


Figure 51. Radial form factor of the core during core-following using the regular NU fuel, for the reference fuelling scheme

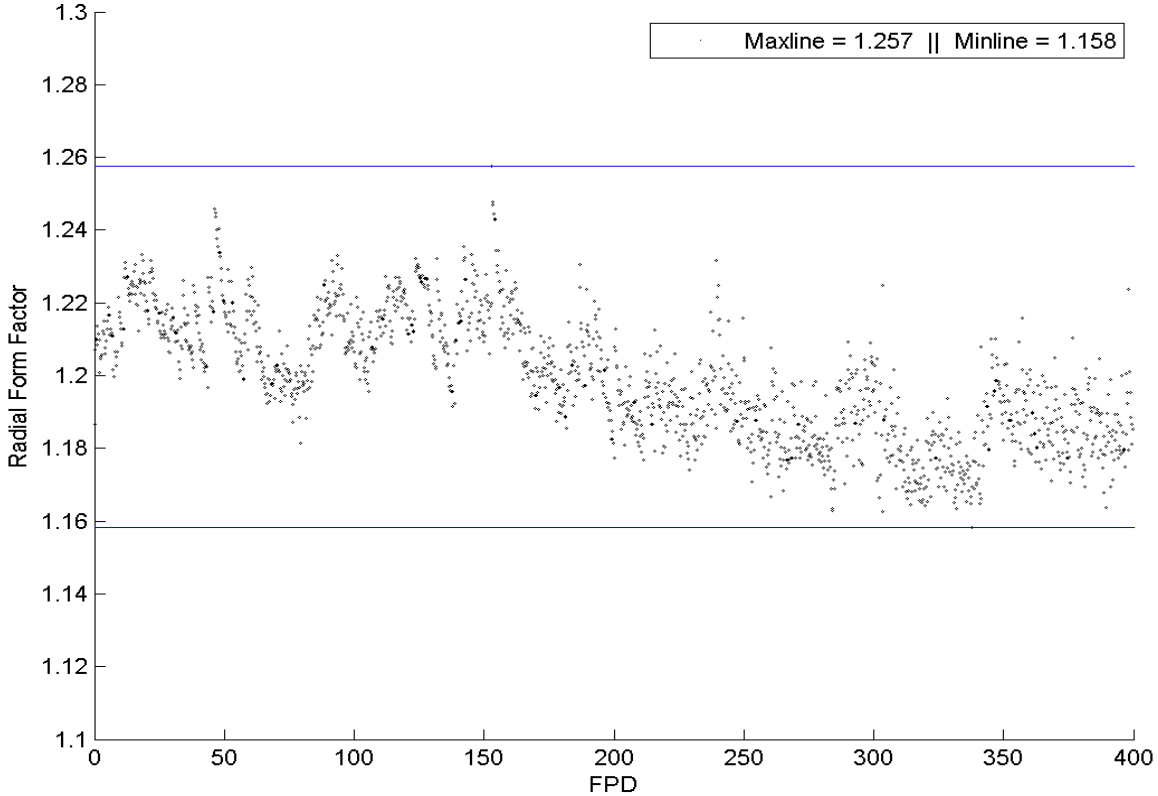


Figure 52. Radial form factor of the core during core-following using BNAF containing 150 mg Gd_2O_3 & 300 mg Eu_2O_3, for the reference fuelling scheme

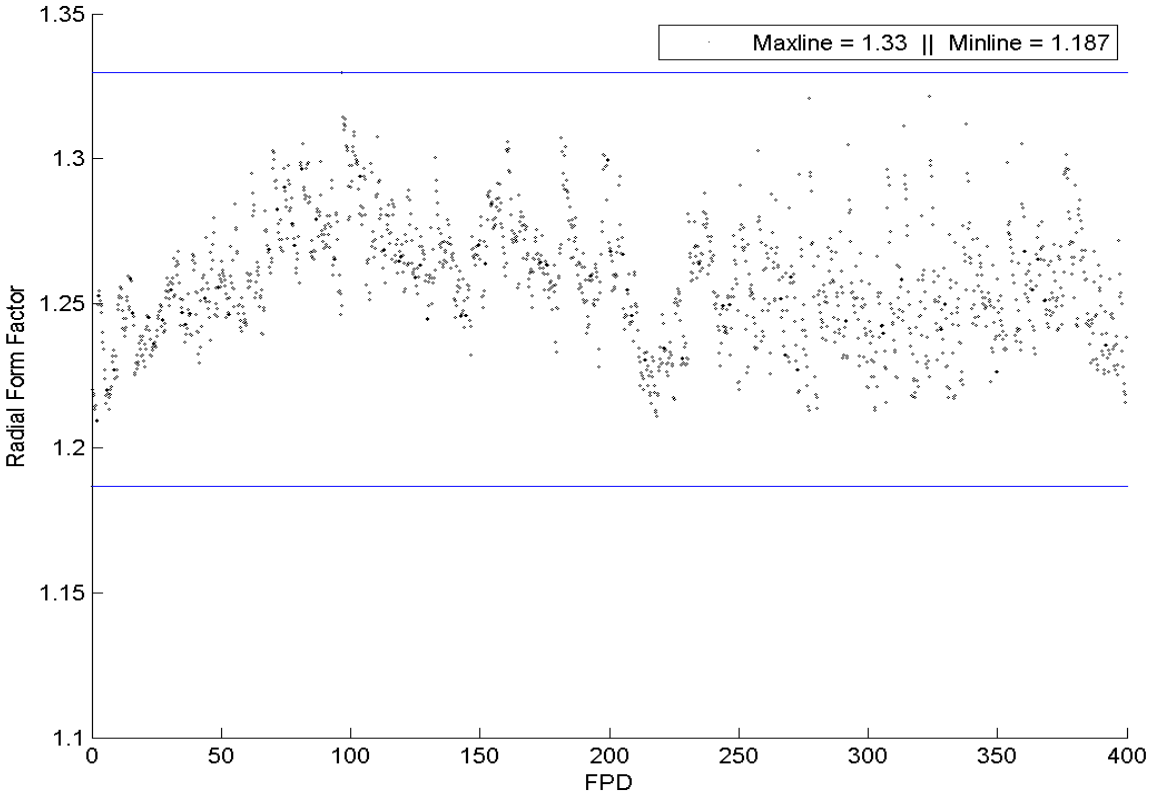


Figure 53. Radial form factor of the core during core-following using the regular NU fuel, for the F8 fuelling scheme

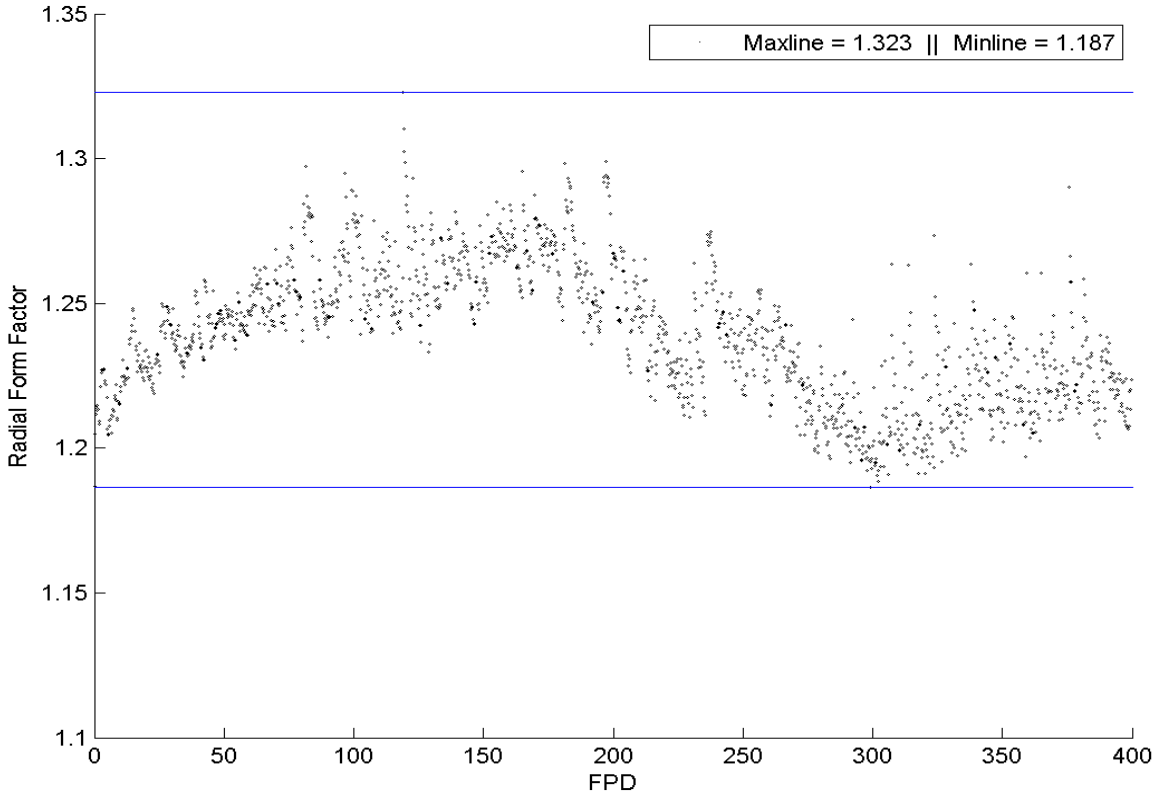


Figure 54. Radial form factor of the core during core-following using BNAF containing 150 mg Gd_2O_3 & 300 mg Eu_2O_3, for the F8 fuelling scheme

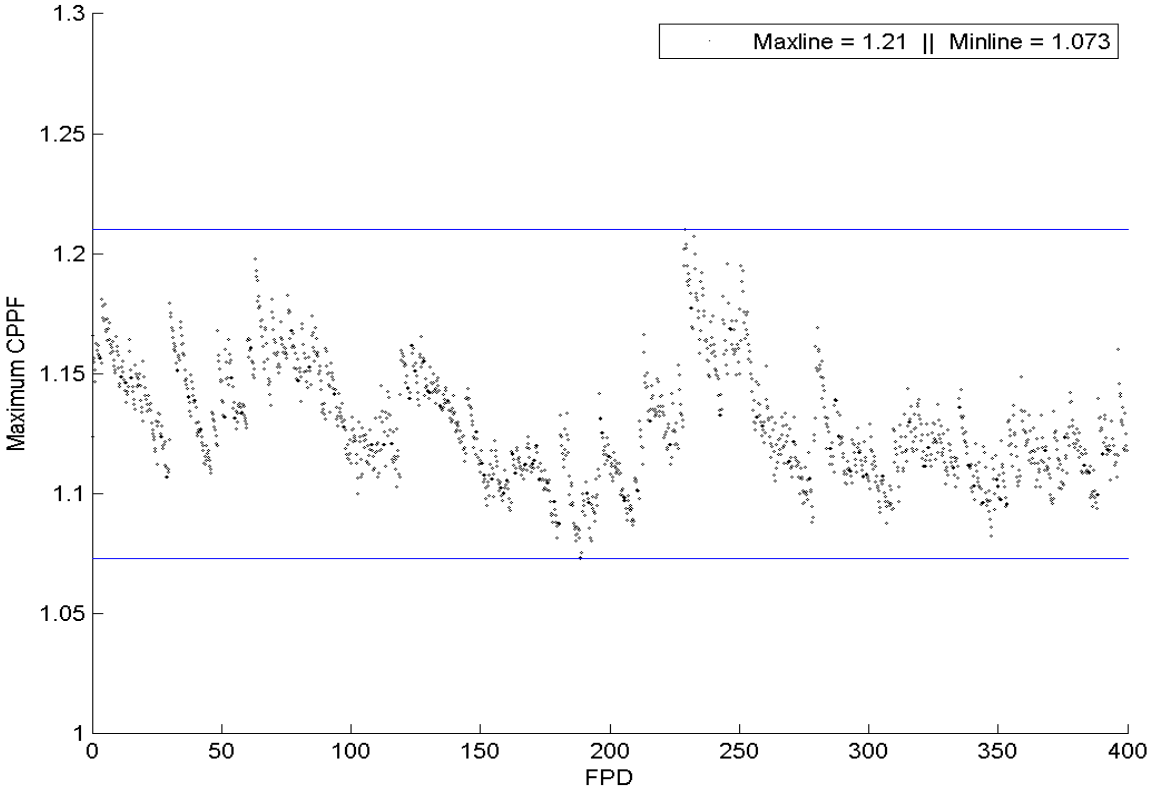


Figure 55. Highest channel power peaking factor in the core during core-following using the regular NU fuel, for the reference fuelling scheme

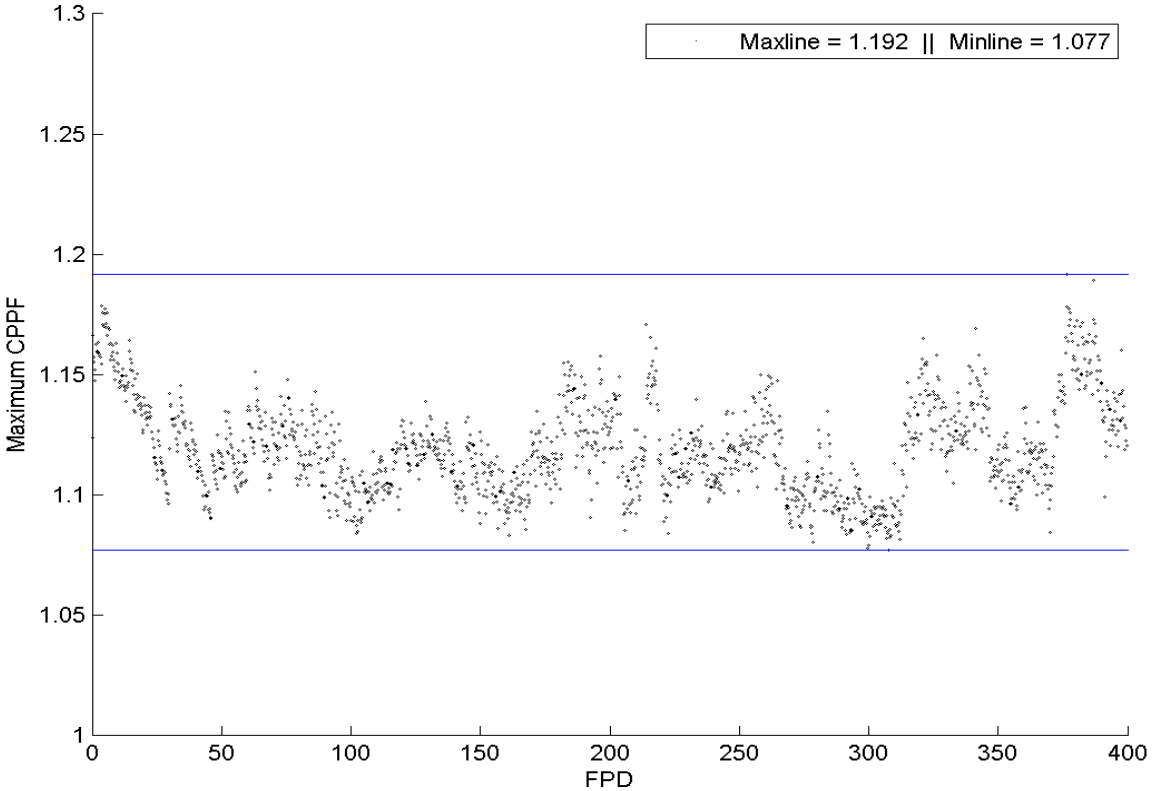


Figure 56. Highest channel power peaking factor in the core during core-following using BNAF containing 150 mg Gd_2O_3 & 300 mg Eu_2O_3, for the reference fuelling scheme

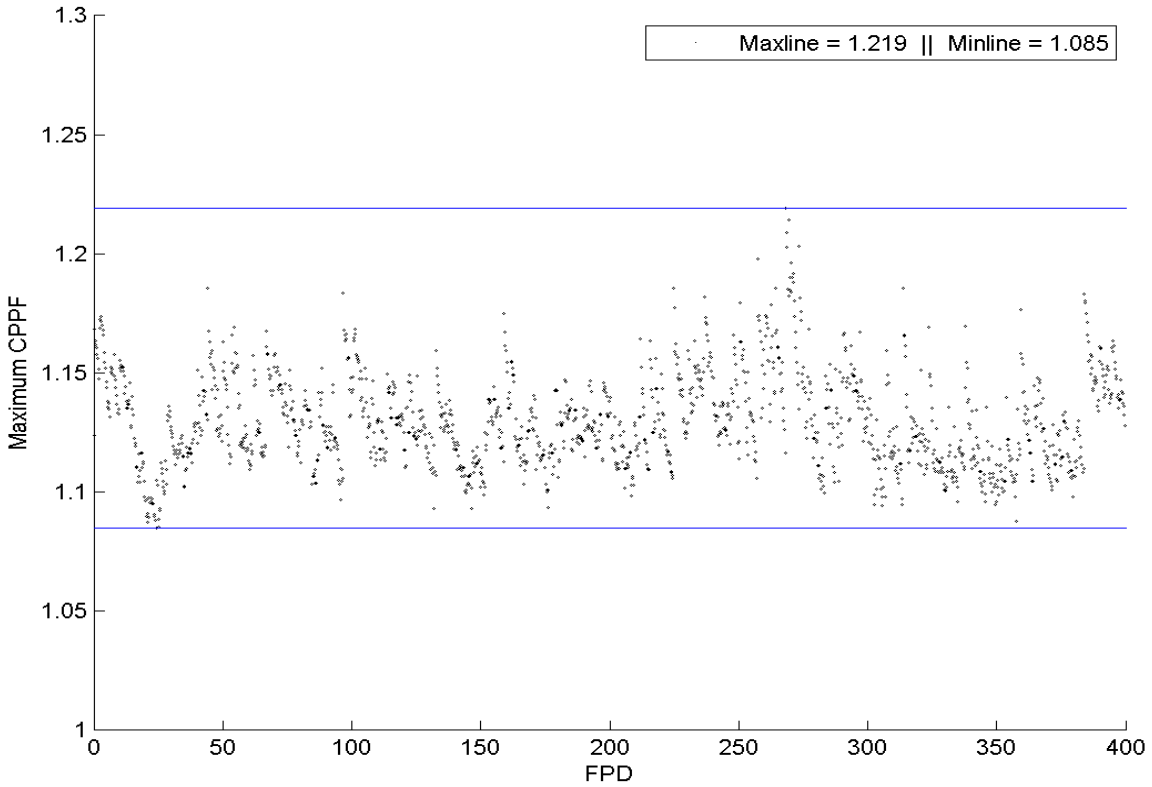


Figure 57. Highest channel power peaking factor in the core during core-following using the regular NU fuel, for the F8 fuelling scheme

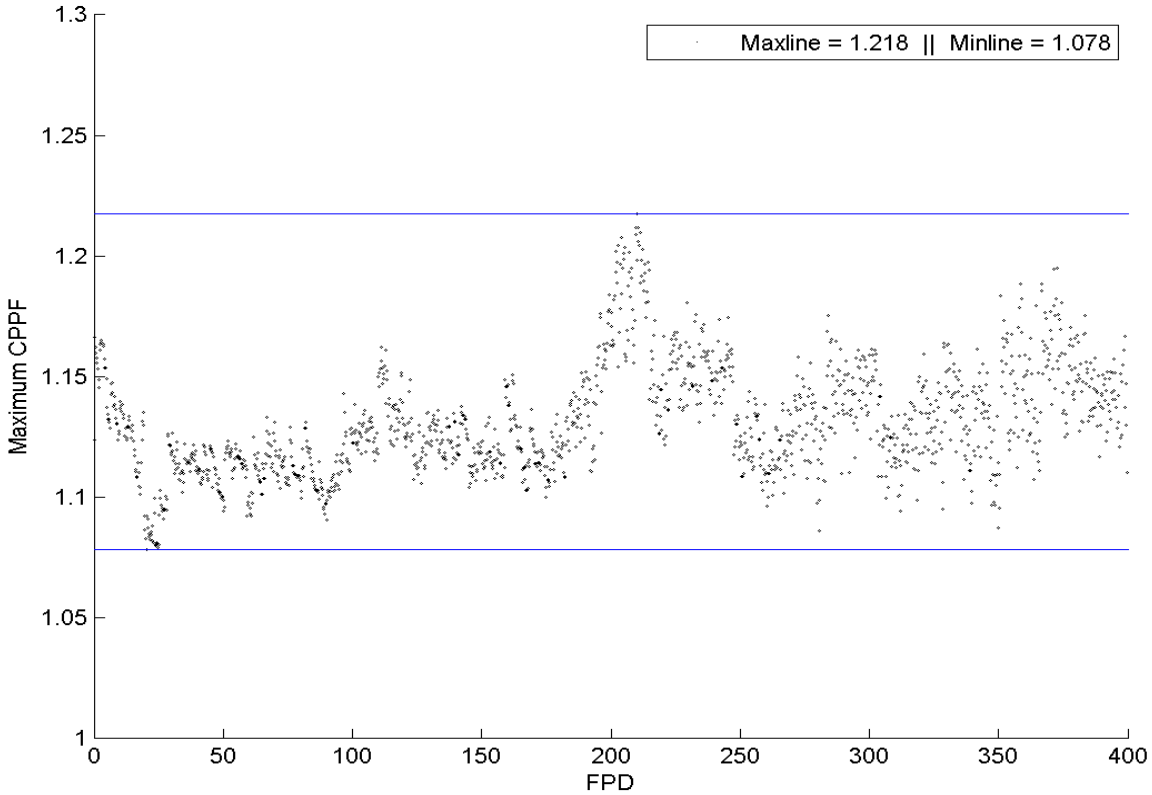


Figure 58. Highest channel power peaking factor in the core during core-following using BNAF containing 150 mg Gd_2O_3 & 300 mg Eu_2O_3, for the F8 fuelling scheme

Table 2. Comparative mean values of the average exit burnup of fuels resulting from the core followings using the regular NU fuel and all BNAFs.

			Regular NU	120 mg Gd2O3	150 mg Gd2O3	150 mg Eu2O3	300 mg Eu2O3	120 mg Gd2O3 & 300 mg Eu2O3	150 mg Gd2O3 & 300 mg Eu2O3
Mean average discharge burnup	**Reference fuelling scheme**	MW h kgU^{-1}	169	169	169	169	169	169	169
		MW d $TonneU^{-1}$	7042	7042	7042	7042	7042	7042	7042
	F8 fuelling scheme	MW h kgU^{-1}	132	132	132	132	132	132	132
		MW d $TonneU^{-1}$	5500	5500	5500	5500	5500	5500	5500

Table 3. Comparative mean values of the average LZC fill and the maximum LZC fill during the core followings using the regular NU fuel and all BNAFs

		Regular NU	120 mg Gd2O3	150 mg Gd2O3	150 mg Eu2O3	300 mg Eu2O3	120 mg Gd2O3 & 300 mg Eu2O3	150 mg Gd2O3 & 300 mg Eu2O3
Reference fuelling scheme	**Mean average LZC fill level**	45%	40%	40%	30%	19%	17%	17%
	Mean maximum LZC fill level	92%	88%	88%	78%	53%	49%	49%
F8 fuelling scheme	**Mean average LZC fill level**	47%	41%	41%	31%	19%	17%	16%
	Mean maximum LZC fill level	97%	95%	94%	82%	60%	55%	55%

Table 4. Comparative minimum, mean, and the maximum values of the peak (highest) channel power in the core during the core followings using the regular NU fuels and all BNAFs

		Regular NU	120 mg Gd2O3	150 mg Gd2O3	150 mg Eu2O3	300 mg Eu2O3	120 mg Gd2O3 & 300 mg Eu2O3	150 mg Gd2O3 & 300 mg Eu2O3
Reference fuelling scheme	Maximum peak channel power (kW)	6988	6946	6943	6954	6961	6948	6945
	Mean peak channel power (kW)	6717	6707	6704	6673	6632	6624	6623
	Minimum peak channel power (kW)	6477	6477	6476	6434	6430	6397	6397
F8 fuelling scheme	Maximum peak channel power (kW)	7344	7301	7296	7319	7364	7321	7307
	Mean peak channel power (kW)	6944	6925	6923	6902	6862	6847	6844
	Minimum peak channel power (kW)	6555	6555	6555	6555	6555	6555	6553

Table 5. Comparative mean values of the radial form factor during the core followings using the regular NU fuel and all BNAFs

		Regular NU	120 mg Gd2O3	150 mg Gd2O3	150 mg Eu2O3	300 mg Eu2O3	120 mg Gd2O3 & 300 mg Eu2O3	150 mg Gd2O3 & 300 mg Eu2O3
Mean Radial Form Factor	Reference fuelling scheme	1.22	1.21	1.21	1.21	1.20	1.20	1.20
	F8 fuelling scheme	1.26	1.25	1.25	1.25	1.24	1.24	1.24

Table 6. Comparative mean values of the highest channel power peaking factors during the core followings using the regular NU fuel and all BNAFs

		Regular NU	120 mg Gd2O3	150 mg Gd2O3	150 mg Eu2O3	300 mg Eu2O3	120 mg Gd2O3 & 300 mg Eu2O3	150 mg Gd2O3 & 300 mg Eu2O3
Mean highest CPPF	Reference fuelling scheme	1.13	1.12	1.12	1.11	1.12	1.12	1.12
	F8 fuelling scheme	1.13	1.13	1.13	1.12	1.13	1.13	1.13

Table 7. Comparative minimum, mean, and the maximum values of the peak (highest) bundle power in the core during the core followings using the regular NU fuels and all BNAFs

		Regular NU	120 mg Gd2O3	150 mg Gd2O3	150 mg Eu2O3	300 mg Eu2O3	120 mg Gd2O3 & 300 mg Eu2O3	150 mg Gd2O3 & 300 mg Eu2O3
Reference fuelling scheme	Maximum peak bundle power (kW)	859	867	870	859	856	857	856
	Mean peak bundle power (kW)	821	820	819	814	809	809	809
	Minimum peak bundle power (kW)	789	787	785	773	770	772	773
F8 fuelling scheme	Maximum peak bundle power (kW)	912	905	902	901	888	877	874
	Mean peak bundle power (kW)	831	828	828	823	815	813	813
	Minimum peak bundle power (kW)	783	781	781	775	774	773	774

Chapter 7: Discussion

7.1 Refuelling Simulations

The refuelling simulations indicated significant reductions in the transient powers of the fresh-inserted fuel bundles and their respective fuel channels when BNAs are present. The reductions were shown to be greater when refuelling with BNAFs containing larger quantities of absorbers (Figures 23 to 34). Between the reference fuelling scheme and the entirely 8-bundle-shift scheme, it was shown that the average reductions in transient channel and bundle powers after refuelling were even more significant for the 8-bundle-shift refuelling mode (also Figures 23 to 34). This is consistent with the fact that twice the amount of fresh fuel bundles is inserted when refuelling with the 8-bundle-shift than with the 4-bundle-shift, resulting in twice the amount of absorbers being included. The magnitude of this difference, however, in recounting the results outlined in Figures 23 to 34, is shown to be greater than just twice that of the average reduction for the reference fuelling scheme, which does not correspond with the fact that twice the amount of fuel with twice the absorbers are inserted. This discrepancy is explained by the fact that an 8-bundle-shift inserts four of its eight fresh fuels directly into the high-power axial positions in the fuel channel (where the reactivity worth is larger) while 4-bundle-shift inserts fresh fuels only into the low-power axial positions and shifts irradiated fuels into the high-power positions. As a result of this, the effectiveness (reactivity worth) of the added absorbers is greater for the entirely 8-bundle-shift fuelling mode. Accordingly, the magnitude of the difference between the average reductions in the transient powers of the reference fuelling scheme and the entirely 8-bundle-shift fuelling mode is observed to be greater than simply a factor of two.

Based on the results of the refuelling simulations, the effectiveness of added absorbers is maximized for cores that refuel a larger fraction of their fuel channels using the 8-bundle-shift mode. A significant effect is, however, also observed for cores refuelled using the reference fuelling scheme (mostly 4-bundle-shift mode). By corollary,

the greater effectiveness of the added absorbers when refuelling with the 8-bundle-shift fuelling mode may be used advantageously to relax constraints on high-power limiting channels that are currently refuelled using the 4-bundle-shift mode. By increasing the number of channels that may be refuelled using the 8-bundle-shift mode, refuelling engineers are able to make fuelling decisions with greater operational flexibility, which may be utilized for core-flattening or reducing the frequency of refuelling to limit the daily strain experienced by refuelling machines.

7.2 Core-following Simulations

The core-following simulations, when refuelling with added absorbers, produced observable reductions in the minimum and the maximum ranges, as well as the mean values of the peak fuel bundle and channel powers within the core (Figures 43 to 50, Table 4, and Table 7). These reductions in power were observed to increase with greater quantities of added absorbers. However, the magnitudes of reductions for the peak powers were observed to be significantly lower than the magnitudes of reduction in transient powers observed for the refuelling simulations presented in Figures 23 to 34 (single refuelling events).

First, this difference between the reductions in transient powers during single refuelling events versus peak powers observed during core-followings is caused in part by the fact the refuelling simulations were conducted with LZCs' fill levels fixed. In this case, the impact of added absorbers is best reflected in the changes to the transient power of the fuel channels alone. In the core-following simulations, on the other hand, the LZCs' fill levels were allowed to change as they should for bulk and spatial controls such that the impact of absorbers are shared between the changes to the transient powers of fuel channels as well as the transient fill levels of LZCs. Secondly, the difference is also likely caused, in large part, by the fact that the peak powers are not always located at the most recently refuelled channels. Therefore, the reductions in peak powers observed during core-following do not always coincide with the reductions in the transient powers of fresh-fuelled channels observed in the refuelling simulations. For this reason, there is

an observable difference between the magnitude of the mean reduction in peak powers for the core-following simulations and the magnitude of reductions in transient powers for the refuelling simulations. This observation suggests that the large, positive impact of added absorbers observed at the fuel bundle and fuel channel levels diminishes at the core level. However, it should be considered that the refuelling and core-following simulations conducted using RFSP do not necessarily reflect the complete physical behaviours of real-life CANDU reactors. For one, the total thermal power of the core in this study is fixed at 2650 MW_{th}. This is an inherent limitation which caps the sum of the simulated magnitudes of powers of fuels and fuel channels in the core to the total power of 2650 MW_{th}. The distribution of power within the core can therefore change freely with respect to the refuelling history, but the magnitudes of the powers are capped in a way that the total power of the core remains constant. For this reason, the impact of added absorbers is also reduced as the transient powers of refuelled channels cannot decrease below that which is required to maintain the total thermal power of the core at 2650 MW_{th}. It is therefore expected that refuelling with added absorbers in a real CANDU core will have a greater impact than the values deduced from the simulations of this study. Nonetheless, the results shown in Figures 43 to 50, Table 4, and Table 7 still indicate significant reductions in the peak channel and fuel bundle powers in the core as result of refuelling with added absorbers. The radial form factor of the core is also shown to improve with increasing quantities of added absorbers as a result of reductions in the peak channel power (Figures 51 to 54 and Table 5). The addition of absorbers is therefore shown to provide some improvements for the power compliance of fuel bundles and channels in the core while also facilitating the practice of core-flattening.

Results shown in Table 2 indicate that among different quantities of added absorbers (but with same refuelling scheme), there are no observable losses in the average discharge burnup of fuels. This is contrary to the small loss in burnup that is expected as the consequence of added absorbers. The reason behind this absence of the expected loss in burnup is likely due to the fact the average fill level of LZCs decreases as the core is transitioned from regular NU fuels to BNAFs. In this case during core-following, the average fill level of LZCs was seen to decrease until it reached new equilibrium values

lower than those prior to the transition (Figures 39 to 42). The establishment of the new equilibrium value for the average fill occurs with respect to the equilibrium excess reactivity of the core. The equilibrium excess reactivity of the core depends on the rate of refuelling and the reactivity worth of fresh fuels that are inserted into the core. Therefore in this study, in which the rate of refuelling was kept constant (identical refuelling history) for fuels containing lower quantities of excess reactivity (due to added absorbers), the bulk excess reactivity of the core is expected to decline throughout the transition. However, as the bulk reactivity control feature of LZCs react to declines in the excess reactivity of the core by decreasing the average fill levels of LZCs, the excess reactivity of the core was consistently compensated in equal proportions to the decline caused by the presence of added absorbers. The core was therefore kept constant at critical with negligible changes between simulations for each fuel type.

A lower excess reactivity in the core as the result of refuelling with added absorbers is, in principle, beneficial for operating the core, because there is less excess reactivity that must be controlled and compensated by devices. For example, this allows the margin provided by the LZCs to be held in reserve as they are not as actively required for bulk reactivity control. Moreover, the rate of refuelling is not required to increase significantly, because the useful in-core irradiation time of fuels does not change significantly with the addition of small quantities of the two BNAs. This is because although fresh BNAFs initially possess lower excess reactivities than the fresh regular NU fuel, they also experience lower rates of decline in their excess reactivity than the regular NU fuel during the first ~50 FPDs that they are irradiated in the core. This is because BNAFs do not experience the rapid decrease in excess reactivity following the fuelling transient and the plutonium peak. It is shown in Figures 35 to 38 that, as the initial NU-fuelled core is transitioned to BNAFs, the rate of bulk reactivity decline of the core decreases and reaches equilibrium at lower values than prior to the transition. This is supported by the fact that, the average fill levels of LZCs for simulations using BNAFs are able to re-establish equilibrium values after declining, because it indicates that the core can be sustained at critical using the same rate of refuelling despite using fuels with lower excess reactivities. This coincides with the results of the WIMS-lattice calculations (Figures 5 to

7) in which it was shown that the decrease in the useful in-core residence time of fuels (before lattice sub-criticality) due to addition of absorbers is negligible for ~120 mg of added Gd_2O_3 and approximately 2 FPDs lower for 150 mg of Gd_2O_3 plus 300 mg of Eu_2O_3. The rate of refuelling should therefore not be required to be increased very much for the quantities of absorbers considered in this study.

The main assumption made in this study was that, by using simulations in RFSP, it would be possible to determine the impact that refuelling with added absorbers within fuel bundles would have on the operating margin of the model core during normal operating conditions. This includes the fundamental assumption that the lattice based method of simulation used in RFSP yields reasonable solutions for the distribution of flux and power within the core. Also, it includes the assumption that the model CANDU core used throughout the study is constructed accurately with good assumptions for its nominal operating condition. Moreover, the methodology for the simulations used in this study was developed with the assumption that a transition from the initial core state containing only regular NU fuels, to being refuelled with BNAFs, would best determine the impact of the added absorbers while keeping other parameters that affect fuel performance constant. Finally, the determination of the impact on the operating margin of the model core was made under the assumption that reductions in the transient powers observed during refuelling simulations and the average (plus the minimum and maximum range) values of the peak powers and the LZC fill levels in the core observed during core-following simulations would provide the best indication of a gain in margin.

The assumption that the RFSP-IST code yields reasonable solutions for the distribution of flux and power within the core is correct, because the code has been validated using benchmark power-reactor measurements. The numerical model of the CANDU core used in the study is assumed to be accurate because the reference data set was obtained directly from a supplier that performs analysis for CANDU NGS. Moreover, the instantaneous properties of the model core at the generic refuelling equilibrium (Figures 15 to 21), generated using RFSP, agrees reasonably with the typical known values for the modelled reactor during normal operation. The methodology used

for the simulations in the study is in line with current practice, and the results provided noticeable indications that there is an overall gain in the operating margin of the core when refuelling with added absorbers. However, the methodology used in the study is specific to transition refuelling from a core initially containing only regular NU fuel. The results of the simulation cannot be compared with benchmark measurements, because a generic equilibrium refuelling state was used as the starting point of simulations, and a generic refuelling history was used to refuel the core. The results of the simulations therefore cannot be compared to existing station operating data. Also, the period of 400 FPDs was used for core-following simulations in this study, because the direct access file (storage file) of the RFSP-IST code is only capable of indexing simulations data up to 400 FPDs worth of energy clock. Although the results of the core-following simulations have shown that in the transition from regular NU fuel to BNAFs, the core establishes new equilibrium values for the average LZC fill levels well within 400 FPDs to compensate for the lower reactivity fuel, the results of the simulations may have provided better average-over-time results if the they had been conducted for periods longer than 400 FPDs. The assumption that reductions in the transient powers during refuelling and the average values of peak powers and LZC fill levels during core-following are good indicators of gain in margin is correct based on the fundamental principles of fuel management, because they are parameters of great importance in determining compliance to operating power limits, which constrain the decision making of refuelling engineers when selecting channels for refuelling.

Overall, the results of the study indicate that there is indeed a tangible gain in the operating margin of the model CANDU core when transitioning from regular NU fuels to fuels containing added absorbers. The results, although based on simulations conducted at a generic refuelling equilibrium state using a generic refuelling history, are credible because the RFSP-IST code is known to yield reasonable results in comparison to power-reactor measurements [14], the reference data set of the model core was built by a qualified supplier, and the methodology used for the simulations is a good representation of the normal operating condition of the model reactor.

Chapter 8: Conclusion

The outcomes of the refuelling simulations indicated a large decrease in the transient powers of fresh-fuelled channels when refuelling with added absorbers. This reduction was shown to increase with greater quantities of added absorbers, and was shown to be greater when refuelling with the 8-bundle-shift mode than the 4-bundle-shift mode. The core-following simulations indicated a noticeable decrease in the minimum and the maximum range, as well as the average value of the peak powers observed in the core during 400 FPDs of core-following. This reduction was also shown to increase with greater quantities of added absorbers. In addition, the core-following simulations have shown that the average fill levels of LZCs decrease until establishing lower equilibrium values when the core is transitioned from refuelling with regular NU fuels to BNAFs. The rate of decrease of the average fill levels was shown to be greater, and the newly established equilibrium values lower, when refuelling with higher quantities of added absorbers. This is indicative of the fact that refuelling with fuels containing lower reactivities (due to higher quantities of BNAs) leads to lower equilibrium values for the excess reactivity in the core, which is compensated for by lower average fill levels of LZCs. The core-following simulations have also indicated that the expected decreases in the average discharge burnup of fuels due to the presence of added absorbers are offset by the reduction in the average fill of LZCs. Additionally, the maximum CPPF in the core, as well as the RFF of the core were shown to decrease during core-following simulations when refuelling with added absorbers. The magnitude of this decrease was also shown to be greater with higher quantities of added absorbers.

Overall, the results of the study have indicated that the impact of added absorbers indeed provides additional operating margins for the reference CANDU core used in the study. This gain is provided in the forms of reductions in the average fill levels of LZCs and decreases in the peak power densities (max channel and bundle powers) of the core. For the proposed quantity of absorbers (150 mg Gd_2O_3 & 300mg Eu_2O_3), a reduction of about 28% in the average fill levels of LZCs and reductions of about 70 kW (~1.4%) and 12 kW (~1.5%) respectively for peak channel and bundle powers were observed. The

principle of using burnable absorbers to tailor the reactivity of fuels therefore appears to present a considerable potential for applications in current, aging reactors as well as future generations of reactors fuelled with natural uranium or even slightly enriched uranium.

Chapter 9: Recommendations

It is recommended that further research and development of the BNAFs be pursued as it is likely that its methodology is applicable not only in the present, but in the future of the Canadian nuclear industry. The future direction for the research on using burnable absorbers to improve operating margins of CANDU reactors is to further optimize the quantity of absorbers to be added to the fuels for an optimal compromise between the gain in margin and losses in the useful reactivity of the fuel. Moreover, the locations to which the added absorbers should best be included are still under investigation. Some considerations include varying the concentrations of the added absorbers in the axial direction of the fuel bundle to help smooth out the end-flux peaking phenomenon (by adding higher quantities of absorbers near the end caps). With further simulation-based optimization and development, it is desired that a prototype of the burnable absorber fuel be built and tested in a research reactor. Additionally, the potential use of burnable absorbers for tailoring the excess reactivity of slightly enriched uranium (SEU) should also be investigated (SEU can also experience the refuelling transients). This is desired as there is potential for future generation of CANDU reactors to use SEU to seek higher discharge burnup of fuels and produce less nuclear waste. The use of BNAFs to refuel channels adjacent to empty channels (no fuels) should also be investigated to see if the effect of added absorbers can be used to reduce the high transient powers of channels adjacent to empty channels. The use of BNAFs for 13-bundle-shift refuelling to flush an entire channel, when a suspected fuel defect is detected, should also be investigated to determine the impact on the transient powers. The optimization of the refuelling scheme of the model core used in this study, for refuelling with BNAFs, should also be investigated to determine the number and the distribution of fuel channels that can be refuelled using the 8-bundle-shift mode and those that are restricted to the 4-bundle-shift mode. Finally, a refuelling study using the operating data and the history of the model core from a NGS should be conducted to produce simulation results that can be benchmarked to existing measurement data.

References

[1] Rouben, B. (1999). CANDU Fuel-Management Course. CANTEACH Publication Library. Retrieved January 29, 2014, from https://canteach.candu.org/Content%20Library/20031101.pdf

[2] Chan, P.K., Paquette, S., & Bonin, H.W. (2015). Variation of Burnable Neutron Absorbers in Heavy Water-Moderated Fuel Lattice: A Potential to Improve CANDU Reactor Operating Margins. *Nuclear Technology*, Accepted for publication.

[3] P. Chan provided background knowledge on safety analysis and licensing of CANDU reactors (personal communication, March 25, 2014).

[4] H. Bonin provided background knowledge and resources on the relationship between OCPL, CCP, CHF, dryout, and OBPL regarding fuel failures (personal communication, March 25, 2014).

[5] Assessment and Management of Aging of Major Nuclear Power Plant Components Important To Safety: CANDU Pressure Tubes. (1998). *International Atomic Energy Agency.* Retrieved January 29, 2014, from http://wwwpub.iaea.org/MTCD/publications/PDF/te_1037_prn.pdf

[6] Safety Analysis of CANDU Nuclear Power Plants: C-006 (REV. 1) (E) Draft Regulatory Guide. (1999). *Atomic Energy Control Board.* Retrieved March 24, 2014, from http://www.nuclearsafety.gc.ca/pubs_catalogue/uploads/c006r1_e.pdf

[7] Tezel, H., Christodoulou, N., & Mesmous, N. (2010). Assessment of the Impact of Aging on the Performance of CANDU Special Safety and Safety Related Systems; Safety Analysis Perspective. *Canadian Nuclear Safety Commission*. Retrieved March 25, 2014 from http://nuclearsafety.gc.ca/eng/resources/research/technical-papers-and-articles/2009/november-0-12-2009.Cfm

[8] P. Chan provided background knowledge on the conception of the idea for the burnable neutron absorber fuel (personal communication, March 25, 2014).

[9] Simnad, M.T. (2012). Nuclear Reactor Materials and Fuels. *University of California, San Diego*. Retrieved March 28, 2014 from http://cryptocomb.org/Nuclear%20Reactor%20Materials%20and%20Fuels.pdf

[10] H. Bonin provided background knowledge and resources on differential and bi-directional fuelling (personal communication, March 25, 2014).

[11] P. Chan provided background knowledge on advantages and disadvantages associated with 4-bundle and 8-bundle shift fuelling modes (personal communication, March 25, 2014).

[12] Paquette, S., Chan, P.K., Farahani, K., Pierce, D., & Bonin, H.W. (2014). Viable Fuel Designs with Neutron-absorbers Added in Fuel Elements. *CANDU Owners Group Inc*. Other Product Report, OP-13-2018.

[13] Shen, W. (2006). RFSP-IST Version REL_3-04: Theory Manual, SQAD-06-5058, 153-117360-STM-002. *CANDU Owners Group Inc*.

[14] Ovanes, M., Jenkins, D.A., Ardeshiri, F., Mao, A.C., Shad, M., Sissaoui, T., & Chow, H.C. (2001). Validation of the RFSP-IST Code Against Power-Reactor Measurements. *Atomic Energy of Canada Limited*. Retrieved February 20, 2015 from http://pbadupws.nrc.gov/docs/ML0236/ML023600327.pdf

[15] Jonkmans, G. (2006). WIMS-AECL Version 3.1 User's Manual, ISTP-05-5115. CANDU Owners Group Inc.

[16] Shen, W. (2002). CANDU Three-Dimensional Neutron Transport Calculations with DRAGON. *Atomic Energy of Canada Limited*. Retrieved October 23, 2014 from http://pbadupws.nrc.gov/docs/ML0236/ML023600301.pdf

[17] MATLAB version 8.1.0.604. (2013). *The Mathworks Inc*., Natick, Massachusets, United States.

[18] Microsoft Excel version 14.0.7143.5000. (2010). *Microsoft Corporation.*, Redmond, Washington, United States.

[19] Monte Carlo N-Particle (MCNP) Transport Code version 6. (2012). *Los Alamos National Laboratory*, Los Alamos, New Mexico, United States.

[20] Schwanke, P. (2006). RFSP-IST Version REL_3-04: Users' Manual, SQAD-06-5054, 153-117360-UM-002. *CANDU Owners Group Inc*.

[21] Enhanced CANDU 6 Technical Summary. (2012). *CANDU Energy Inc*. Retrieved January 29, 2014, from http://www.candu.com/site/media/Parent/EC6%20Technical%20Summary_2012-04.pdf

[22] Experience from Operating and Fuelling Nuclear Power Plants. (1974). *International Atomic Energy Agency*. Retrieved January 29, 2014, from http://www.iaea.org/inis/collection/NCLCollectionStore/_Public/05/130/5130629.pdf.

[23] UxC Nuclear Fuel Price Indicators. (2014). *UxC: Ux Prices*. Retrieved January 21, 2014, from http://www.uxc.com/review/uxc_Prices.aspx.

[24] Assessment and Management of Aging of Major Nuclear Power Plant Components Important To Safety: CANDU Reactor Assemblies. (1998). *International Atomic Energy Agency.* Retrieved January 29, 2014, from http://wwwpub.iaea.org/MTCD/publications/PDF/te_1197_prn.pdf

[25] Brooks, L. G. (1993). A Short History of the CANDU Nuclear Power System. *CANTEACH Publication Library*. Retrieved March 24, 2014, from http://www.canteach.candu.org/Content%20Library/19930101.pdf

[26] Couture, M. (2013). Fuel performance in aging CANDU reactors – A quick overview of the CNSC regulatory oversight activities of the past 15 years and of the lessons learned. *Canadian Nuclear Safety Commission*.

[27] Technical Note: Positive coolant void reactivity feedback phenomenon in currently operating CANDU reactors (2009). *Canadian Nuclear Safety Commission.* Retrieved March 24, 2014, from http://www.ontario-sea.org/Storage/64/5551_Technical_note_on_positive_coolant_void_reactivity_feedback_phenomenon_E.pdf

The following reference is presented without a reference number as it was consulted after the completion of this thesis and appears here for the convenience of the reader because of its pertinence to the present work:

Choi, H. (1999). A fast-running fuel management program for a CANDU reactor. *Annals of Nuclear Energy*, 1-10.

Appendix A

Automatic WIMS Input and Output Generator and Comparator

Part 1: Input and Output Generation:

```
% Created by: David Tom & Jason Song (28-06-14)
clc;
clear;

%-------------------------------------------------------------------------
--%
%--------------------------- PARAMETER INPUTS --------------------------
--%
%-------------------------------------------------------------------------
--%

% Name and Enrichment/Poison Parameters

% Specify Poison Content Here
mass_GD2O3=0.100; % Gd
mass_EU2O3=0.300; % Eu
mass_SM2O3=0.0; % Sm

wimsinp_filename=strcat('fuel_2g_',num2str(mass_GD2O3*1000),'mg_',num2st
r(mass_EU2O3*1000),'mg');                  %Filename
case_name=wimsinp_filename;                          %Legend Name

%%%%%%%%%%%%%%%%%%%%%%%%%%%%      WIMS      BLANK      TEMPLATE      READ
%%%%%%%%%%%%%%%%%%%%%%%

fopen('WIMS_template.txt');
string_WIMS=fileread('WIMS_template.txt');

%-------------------------------------------------------------------------
--%
%---------------------------- CASE 1 BEGIN-------------------------------
--%
%-------------------------------------------------------------------------
--%



%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%
%%%%%%%%%%%%%%%%%%%%%%%%      CALCULATING      CASE      1      FUEL      INFO
%%%%%%%%%%%%%%%%%%%%%%%

U234_mm=234.0409456;  %Molar Masses
U235_mm=235.0439231;
U238_mm=238.0507826;
O_mm=15.9949146;
```

```
%NU_mm_1=((U234_molar_1/100)*U234_mm)+((U235_molar_1/100)*U235_mm)+((U23
8_molar_1/100)*U238_mm);

%UO2_mm_1=NU_mm_1+2*O_mm;
%NU_mf_1=NU_mm_1/UO2_mm_1;
%O_mf_1=((2*O_mm)/UO2_mm_1)*100;

U234_mf_1=.005408;  %Mass Fractions (ALREADY SPECIFIED)
U235_mf_1=0.710971;
U238_mf_1=99.28362;
O_mf=13.44251;


%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%
%%%%%%%%%%%%%%%%%%%%%%%     CALCULATING     CASE     1     POISON     INFO
%%%%%%%%%%%%%%%%%%%%

% Gadolinium Oxide
GD2O3_density=7.41;
volume_GD2O3=mass_GD2O3/GD2O3_density;

GD152_mf=0.001676407;
GD154_mf=0.018513527;
GD155_mf=0.126506177;
GD156_mf=0.176100569;
GD157_mf=0.135499838;
GD158_mf=0.216438836;
GD160_mf=0.192888915;
GD_O16_mf=0.132375731;

mass_GD152=GD152_mf*mass_GD2O3;
mass_GD154=GD154_mf*mass_GD2O3;
mass_GD155=GD155_mf*mass_GD2O3;
mass_GD156=GD156_mf*mass_GD2O3;
mass_GD157=GD157_mf*mass_GD2O3;
mass_GD158=GD158_mf*mass_GD2O3;
mass_GD160=GD160_mf*mass_GD2O3;
mass_GD_O16=GD_O16_mf*mass_GD2O3;

% Europium Oxide
EU2O3_density=7.4;
volume_EU2O3=mass_EU2O3/EU2O3_density;

EU151_mf=0.409972919;    % EU-153
EU153_mf=0.453673424;    % EU-153
EU_O16_mf=0.136353656;   % Oxygen


mass_EU151=EU151_mf*mass_EU2O3;
mass_EU153=EU153_mf*mass_EU2O3;
mass_EU_O16=EU_O16_mf*mass_EU2O3;
```

```
% Original Clad Content

mass_ZR90=13.3872109851195;
mass_ZR91=2.9519081394013;
mass_ZR92=4.5615838326246;
mass_ZR94=4.7234226385859;
mass_ZR96=0.7771694830476;
mass_FE54=0.0032180747163;
mass_FE56=0.0518783821487;
mass_FE57=0.0012090650269;
mass_FE58=0.0001640321372;
mass_CR50=0.0011223146764;
mass_CR52=0.0225070652933;
mass_CR53=0.0026009777749;
mass_CR54=0.0006597232257;
mass_NI58=0.0013348034227;
mass_NI60=0.0005278753851;
mass_NI64=0.0000196325070;
mass_B10=0.0000160319144;

% Samarium Oxide
SM2O3_density=8.347;
volume_SM2O3=mass_SM2O3/SM2O3_density;

SM144_mf=0.02558764;
SM147_mf=0.126394633;
SM148_mf=0.095865353;
SM149_mf=0.117867871;
SM150_mf=0.063628977;
SM152_mf=0.232646748;
SM154_mf=0.200400446;
SM_O16_mf=0.137608333;

mass_SM144=SM144_mf*mass_SM2O3;
mass_SM147=SM147_mf*mass_SM2O3;
mass_SM148=SM148_mf*mass_SM2O3;
mass_SM149=SM149_mf*mass_SM2O3;
mass_SM150=SM150_mf*mass_SM2O3;
mass_SM152=SM152_mf*mass_SM2O3;
mass_SM154=SM154_mf*mass_SM2O3;
mass_SM_O16=SM_O16_mf*mass_SM2O3;

%%%%%%%%%%%%%%%%%%%%%%%%%%%              DENSITY              CALCULATIONS
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
r_outclad=0.6096;
r_inclad=0.609;
ch_length=594;
True_Volume_Clad1=((pi*((r_outclad^2)-(r_inclad^2)))*ch_length*37)/12;
% 37 elements, channel length divided by 12 for 12 fuel bundles.
Clad1_pure_density=6.3918;

Volume_poison=volume_GD2O3+volume_EU2O3+volume_SM2O3;
Volume_pure_Clad1=True_Volume_Clad1-Volume_poison;

mass_poison_total=mass_EU2O3+mass_GD2O3+mass_SM2O3;
```

```
mass_Clad1_pure=True_Volume_Clad1*Clad1_pure_density;

Total_Clad1_mass=mass_poison_total+mass_Clad1_pure;
Total_Clad1_density=Total_Clad1_mass/True_Volume_Clad1;

%%%%%%%%%%%%%%%%%%%%%%%%          FINAL          MASS          FRACTIONS
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

mass_fraction_SM144=(mass_SM144/Total_Clad1_mass)*100;
mass_fraction_SM147=(mass_SM147/Total_Clad1_mass)*100;
mass_fraction_SM148=(mass_SM148/Total_Clad1_mass)*100;
mass_fraction_SM149=(mass_SM149/Total_Clad1_mass)*100;
mass_fraction_SM150=(mass_SM150/Total_Clad1_mass)*100;
mass_fraction_SM152=(mass_SM152/Total_Clad1_mass)*100;
mass_fraction_SM154=(mass_SM154/Total_Clad1_mass)*100;

mass_fraction_EU151=(mass_EU151/Total_Clad1_mass)*100;
mass_fraction_EU153=(mass_EU153/Total_Clad1_mass)*100;

mass_fraction_GD152=(mass_GD152/Total_Clad1_mass)*100;
mass_fraction_GD154=(mass_GD154/Total_Clad1_mass)*100;
mass_fraction_GD155=(mass_GD155/Total_Clad1_mass)*100;
mass_fraction_GD156=(mass_GD156/Total_Clad1_mass)*100;
mass_fraction_GD157=(mass_GD157/Total_Clad1_mass)*100;
mass_fraction_GD158=(mass_GD158/Total_Clad1_mass)*100;
mass_fraction_GD160=(mass_GD160/Total_Clad1_mass)*100;

mass_fraction_O16=((mass_EU_O16+mass_GD_O16+mass_SM_O16)/Total_Clad1_mas
s)*100;

mass_fraction_carbon=(mass_Clad1_pure/Total_Clad1_mass)*100;

mass_fraction_ZR90=13.3872109851195/Total_Clad1_mass*100;
mass_fraction_ZR91=2.9519081394013/Total_Clad1_mass*100;
mass_fraction_ZR92=4.5615838326246/Total_Clad1_mass*100;
mass_fraction_ZR94=4.7234226385859/Total_Clad1_mass*100;
mass_fraction_ZR96=0.7771694830476/Total_Clad1_mass*100;
mass_fraction_FE54=0.0032180747163/Total_Clad1_mass*100;
mass_fraction_FE56=0.0518783821487/Total_Clad1_mass*100;
mass_fraction_FE57=0.0012090650269/Total_Clad1_mass*100;
mass_fraction_FE58=0.0001640321372/Total_Clad1_mass*100;
mass_fraction_CR50=0.0011223146764/Total_Clad1_mass*100;
mass_fraction_CR52=0.0225070652933/Total_Clad1_mass*100;
mass_fraction_CR53=0.0026009777749/Total_Clad1_mass*100;
mass_fraction_CR54=0.0006597232257/Total_Clad1_mass*100;
mass_fraction_NI58=0.0013348034227/Total_Clad1_mass*100;
mass_fraction_NI60=0.0005278753851/Total_Clad1_mass*100;
mass_fraction_NI64=0.0000196325070/Total_Clad1_mass*100;
mass_fraction_B10=0.0000160319144/Total_Clad1_mass*100;

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%     WRITE     NEW     CASE     1     INPUTS
%%%%%%%%%%%%%%%%%%%%%%
```

```
% Identify Location of Material Properties

% ZR90
ZR90_num=mass_fraction_ZR90;
find_ZR90=strfind(string_WIMS,'ZR90=');
window=string_WIMS(find_ZR90(4)+5:find_ZR90(4)+18);
new_ZR90=sprintf('%0.9f',ZR90_num);
new=strrep(string_WIMS,window,new_ZR90);

% ZR91
ZR91_num=mass_fraction_ZR91;
find_ZR91=strfind(string_WIMS,'ZR91=');
window=string_WIMS(find_ZR91(4)+5:find_ZR91(4)+18);
new_ZR91=sprintf('%0.9f',ZR91_num);
new2=strrep(new,window,new_ZR91);

% ZR92
ZR92_num=mass_fraction_ZR92;
find_ZR92=strfind(string_WIMS,'ZR92=');
window=string_WIMS(find_ZR92(4)+5:find_ZR92(4)+18);
new_ZR92=sprintf('%0.9f',ZR92_num);
new3=strrep(new2,window,new_ZR92);

% ZR94
ZR94_num=mass_fraction_ZR94;
find_ZR94=strfind(string_WIMS,'ZR94=');
window=string_WIMS(find_ZR94(4)+5:find_ZR94(4)+18);
new_ZR94=sprintf('%0.9f',ZR94_num);
new4=strrep(new3,window,new_ZR94);

% ZR96
ZR96_num=mass_fraction_ZR96;
find_ZR96=strfind(string_WIMS,'ZR96=');
window=string_WIMS(find_ZR96(4)+5:find_ZR96(4)+18);
new_ZR96=sprintf('%0.9f',ZR96_num);
new5=strrep(new4,window,new_ZR96);

% FE54
FE54_num=mass_fraction_FE54;
find_FE54=strfind(string_WIMS,'FE54=');
window=string_WIMS(find_FE54(4)+5:find_FE54(4)+18);
new_FE54=sprintf('%0.9f',FE54_num);
new6=strrep(new5,window,new_FE54);

% FE56
FE56_num=mass_fraction_FE56;
find_FE56=strfind(string_WIMS,'FE56=');
window=string_WIMS(find_FE56(4)+5:find_FE56(4)+18);
new_FE56=sprintf('%0.9f',FE56_num);
new7=strrep(new6,window,new_FE56);

% FE57
FE57_num=mass_fraction_FE57;
find_FE57=strfind(string_WIMS,'FE57=');
window=string_WIMS(find_FE57(4)+5:find_FE57(4)+18);
```

```
new_FE57=sprintf('%0.9f',FE57_num);
new8=strrep(new7,window,new_FE57);

% FE58
FE58_num=mass_fraction_FE58;
find_FE58=strfind(string_WIMS,'FE58=');
window=string_WIMS(find_FE58(4)+5:find_FE58(4)+18);
new_FE58=sprintf('%0.9f',FE58_num);
new9=strrep(new8,window,new_FE58);

% CR50
CR50_num=mass_fraction_CR50;
find_CR50=strfind(string_WIMS,'CR50=');
window=string_WIMS(find_CR50(4)+5:find_CR50(4)+18);
new_CR50=sprintf('%0.9f',CR50_num);
new10=strrep(new9,window,new_CR50);

% CR52
CR52_num=mass_fraction_CR52;
find_CR52=strfind(string_WIMS,'CR52=');
window=string_WIMS(find_CR52(4)+5:find_CR52(4)+18);
new_CR52=sprintf('%0.9f',CR52_num);
new11=strrep(new10,window,new_CR52);

% CR53
CR53_num=mass_fraction_CR53;
find_CR53=strfind(string_WIMS,'CR53=');
window=string_WIMS(find_CR53(4)+5:find_CR53(4)+18);
new_CR53=sprintf('%0.9f',CR53_num);
new12=strrep(new11,window,new_CR53);

% CR54
CR54_num=mass_fraction_CR54;
find_CR54=strfind(string_WIMS,'CR54=');
window=string_WIMS(find_CR54(4)+5:find_CR54(4)+18);
new_CR54=sprintf('%0.9f',CR54_num);
new13=strrep(new12,window,new_CR54);

% NI58
NI58_num=mass_fraction_NI58;
find_NI58=strfind(string_WIMS,'NI58=');
window=string_WIMS(find_NI58(4)+5:find_NI58(4)+18);
new_NI58=sprintf('%0.9f',NI58_num);
new14=strrep(new13,window,new_NI58);

% NI60
NI60_num=mass_fraction_NI60;
find_NI60=strfind(string_WIMS,'NI60=');
window=string_WIMS(find_NI60(4)+5:find_NI60(4)+18);
new_NI60=sprintf('%0.9f',NI60_num);
new15=strrep(new14,window,new_NI60);

% NI64
NI64_num=mass_fraction_NI64;
find_NI64=strfind(string_WIMS,'NI64=');
```

```
window=string_WIMS(find_NI64(4)+5:find_NI64(4)+18);
new_NI64=sprintf('%0.9f',NI64_num);
new16=strrep(new15,window,new_NI64);

% B10
B10_num=mass_fraction_B10;
find_B10=strfind(string_WIMS,'B10=');
window=string_WIMS(find_B10(4)+4:find_B10(4)+18);
new_B10=sprintf('%0.9f',B10_num);
new17=strrep(new16,window,new_B10);

% EU151
EU151_num=mass_fraction_EU151;
find_EU151=strfind(string_WIMS,'EU151=');
window=string_WIMS(find_EU151+6:find_EU151+18);
new_EU151=sprintf('%0.9f',EU151_num);
new18=strrep(new17,window,new_EU151);

% EU153
EU153_num=mass_fraction_EU153;
find_EU153=strfind(string_WIMS,'EU153=');
window=string_WIMS(find_EU153+6:find_EU153+18);
new_EU153=sprintf('%0.9f',EU153_num);
new19=strrep(new18,window,new_EU153);

% GD152
GD152_num=mass_fraction_GD152;
find_GD152=strfind(string_WIMS,'GD152=');
window=string_WIMS(find_GD152+6:find_GD152+18);
new_GD152=sprintf('%0.9f',GD152_num);
new20=strrep(new19,window,new_GD152);

% GD154
GD154_num=mass_fraction_GD154;
find_GD154=strfind(string_WIMS,'GD154=');
window=string_WIMS(find_GD154+6:find_GD154+18);
new_GD154=sprintf('%0.9f',GD154_num);
new21=strrep(new20,window,new_GD154);

% GD155
GD155_num=mass_fraction_GD155;
find_GD155=strfind(string_WIMS,'GD155=');
window=string_WIMS(find_GD155+6:find_GD155+18);
new_GD155=sprintf('%0.9f',GD155_num);
new22=strrep(new21,window,new_GD155);

% GD156
GD156_num=mass_fraction_GD156;
find_GD156=strfind(string_WIMS,'GD156=');
window=string_WIMS(find_GD156+6:find_GD156+18);
new_GD156=sprintf('%0.9f',GD156_num);
new23=strrep(new22,window,new_GD156);

% GD157
GD157_num=mass_fraction_GD157;
```

```
find_GD157=strfind(string_WIMS,'GD157=');
window=string_WIMS(find_GD157+6:find_GD157+18);
new_GD157=sprintf('%0.9f',GD157_num);
new24=strrep(new23,window,new_GD157);

% GD158
GD158_num=mass_fraction_GD158;
find_GD158=strfind(string_WIMS,'GD158=');
window=string_WIMS(find_GD158+6:find_GD158+18);
new_GD158=sprintf('%0.9f',GD158_num);
new25=strrep(new24,window,new_GD158);

% GD160
GD160_num=mass_fraction_GD160;
find_GD160=strfind(string_WIMS,'GD160=');
window=string_WIMS(find_GD160+6:find_GD160+18);
new_GD160=sprintf('%0.9f',GD160_num);
new26=strrep(new25,window,new_GD160);

% SM144
%SM144_num=mass_fraction_SM144;
%find_SM144=strfind(string_WIMS,'SM144=');
%window=string_WIMS(find_SM144+6:find_SM144+18);
%new_SM144=sprintf('%0.9f',SM144_num);
%new10=strrep(new9,window,new_SM144);

% SM147
%SM147_num=mass_fraction_SM147;
%find_SM147=strfind(string_WIMS,'SM147=');
%window=string_WIMS(find_SM147+6:find_SM147+18);
%new_SM147=sprintf('%0.9f',SM147_num);
%new11=strrep(new10,window,new_SM147);

% SM148
%SM148_num=mass_fraction_SM148;
%find_SM148=strfind(string_WIMS,'SM148=');
%window=string_WIMS(find_SM148+6:find_SM148+18);
%new_SM148=sprintf('%0.9f',SM148_num);
%new12=strrep(new11,window,new_SM148);

% SM149
%SM149_num=mass_fraction_SM149;
%find_SM149=strfind(string_WIMS,'SM149=');
%window=string_WIMS(find_SM149+6:find_SM149+18);
%new_SM149=sprintf('%0.9f',SM149_num);
%new13=strrep(new12,window,new_SM149);

% SM150
%SM150_num=mass_fraction_SM150;
%find_SM150=strfind(string_WIMS,'SM150=');
%window=string_WIMS(find_SM150+6:find_SM150+18);
%new_SM150=sprintf('%0.9f',SM150_num);
%new14=strrep(new13,window,new_SM150);

% SM152
```

```
%SM152_num=mass_fraction_SM152;
%find_SM152=strfind(string_WIMS,'SM152=');
%window=string_WIMS(find_SM152+6:find_SM152+18);
%new_SM152=sprintf('%0.9f',SM152_num);
%new15=strrep(new14,window,new_SM152);

% SM154
%SM154_num=mass_fraction_SM154;
%find_SM154=strfind(string_WIMS,'SM154=');
%window=string_WIMS(find_SM154+6:find_SM154+18);
%new_SM154=sprintf('%0.9f',SM154_num);
%new16=strrep(new15,window,new_SM154);

% O16
O16_num=mass_fraction_O16;
find_O16=strfind(string_WIMS,'O16=');
window=string_WIMS(find_O16(3)+4:find_O16(3)+18);
new_O16=sprintf('%0.9f',O16_num);
new27=strrep(new26,window,new_O16);

% C
%C_num=mass_fraction_carbon;
%find_C=strfind(string_WIMS,'C=');
%window=string_WIMS(find_C+7:find_C+17);
%new_C=sprintf('%0.9f',C_num);
%new18=strrep(new17,window,new_C);

% Density Clad 1
density_num=Total_Clad1_density;
find_density=strfind(string_WIMS,'MATERIAL Clad1');
window=string_WIMS(find_density+15:find_density+21);
new_density=sprintf('%0.5f',density_num);
new28=strrep(new27,window,new_density);

% U235
%U235_num=U235_mf_1;
%find_U235=strfind(string_WIMS,'U235=');
%window=string_WIMS(find_U235+5:find_U235+13)
%new_U235=sprintf('%0.5f',U235_num);
%new20=strrep(new19,window,new_U235);

% U238
%U238_num=U238_mf_1;
%=strfind(string_WIMS,'U238=');
%window=string_WIMS(find_U238+5:find_U238+13)
%new_U238=sprintf('%0.5f',U238_num)
%new21=strrep(new20,window,new_U238);

% U234
%U234_num=U234_mf_1;
%find_U234=strfind(string_WIMS,'U234=');
%window=string_WIMS(find_U234+5:find_U234+12)
%new_U234=sprintf('%0.5f',U234_num)
%new22=strrep(new21,window,new_U234);
```

```
% O16 Fuel
%O16_fuel_num=O_mf_1;
%find_O16=strfind(string_WIMS,'O16=');
%window=string_WIMS(find_O16(2)+4:find_O16(2)+12)
%new_O16_fuel=sprintf('%0.5f',O16_fuel_num)
%new23=strrep(new22,window,new_O16_fuel);

wimsinp=strcat(wimsinp_filename,'.inp');
wimsout=strcat(wimsinp_filename,'.out');
fid=fopen(wimsinp,'w');
fprintf(fid,'%s',new28);

%%%%%%%%%%%%%%%%%%%%%%%%%%%%       RUN       COMMAND       PROMPT       1
%%%%%%%%%%%%%%%%%%%%%%%%%%%
delete(wimsout);

cmd_wimsinp=horzcat('wims3121 ',wimsinp_filename);
[status1,cmdout1]=system('dir');
[status2,cmdout2]=system(cmd_wimsinp);

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%
%%%%%%%%%%%%%%%%%%%%%%%%       CASE       1       OUTPUT       PROCESSSING
%%%%%%%%%%%%%%%%%%%%%%%%%%

fclose('all');

fopen(wimsout);                     % Read Input file
wimsout_string=fileread(wimsout);     %  Convert  file  into  a  single
character string


% Identify locations where string says certain KEY words
keff_start=strfind(wimsout_string,'                        K-INFINITY');
%Location tag for k-eff
day_start=strfind(wimsout_string,'TIME                              =');
%Location tag for number of days
burnup_start=strfind(wimsout_string,'BURNUP                           =');
%Location tag for burnup
power_start=strfind(wimsout_string,'POWER                             =');
%Location tag for power
total_irr_start=strfind(wimsout_string,'INITIAL      IRRADIATION   =');
%Location tag for total irradiation
thermal_irr_start=strfind(wimsout_string,'THERMAL    IRRADIATION    =');
%Location tag for thermal irradiation

% Setup loops to make matrix of elements %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%K-eff Loop
for i=1:1:length(keff_start);
k_eff(i)=str2num(wimsout_string(keff_start(i)+18:keff_start(i)+24));
end

%Time Loop
```

```
for i=1:1:length(day_start)
    h=0;
for j=1:1:10

day_test=wimsout_string(day_start(i)+j:day_start(i)+13);
numerical=isstrprop(day_test,'digit');

if numerical(1)==1 && h==0
    day(i)=str2num(day_test);
    h=1;
end
end
end

%Burnup Loop
for i=1:1:length(burnup_start)
    h=0;
for j=1:1:18

burnup_test=wimsout_string(burnup_start(i)+j:burnup_start(i)+18);
numerical=isstrprop(burnup_test,'digit');

if numerical(1)==1 && h==0
    burnup(i)=str2num(burnup_test);
    h=1;
end
end
end

%Power Loop
for i=1:1:length(power_start)
    h=0;
for j=1:1:15

power_test=wimsout_string(power_start(i)+j:power_start(i)+15);
numerical=isstrprop(power_test,'digit');

if numerical(1)==1 && h==0
    power(i)=str2num(power_test);
    h=1;
end
end
end

%Total Irradiation Loop
for i=1:1:length(total_irr_start)
    h=0;
for j=1:1:29

total_irr_test=wimsout_string(total_irr_start(i)+j:total_irr_start(i)+29
);
numerical=isstrprop(total_irr_test,'digit');

if numerical(1)==1 && h==0
    total_irr(i)=str2num(total_irr_test);
```

```
    h=1;
end
end
end

%Thermal Irradiation Loop
for i=1:1:length(thermal_irr_start)
    h=0;
for j=1:1:28

thermal_irr_test=wimsout_string(thermal_irr_start(i)+j:thermal_irr_start
(i)+28);
numerical=isstrprop(thermal_irr_test,'digit');

if numerical(1)==1 && h==0
    thermal_irr(i)=str2num(thermal_irr_test);
    h=1;
end
end
end


%-------------------------------------------------------------------------
--%
%------------------------ PLOT RESPECTIVE GRAPHS -------------------------
--%
%-------------------------------------------------------------------------
--%

hold off;

% K-effective plot
figure(1);
plot(day,k_eff)
xlabel('Time (days)')
ylabel('K-effective')
title ('K-effective of a single fuel bundle as a function of time')
hold on;
hline=refline(0,1);
set(hline,'Color','k')
legend(case_name)

hold off;

% Burnup plot
figure(2);
plot(day,burnup);
xlabel('Time (days)')
ylabel('burnup (MWD)')
title ('Burnup of a single fuel bundle as a function of time')
legend(case_name)

hold off;

% Power plot
```

```
figure(3);
plot(day,power)
xlabel('Time (days)')
ylabel('Power (MW)')
title ('Power of a single fuel bundle as a function of time')
legend(case_name)

hold off;

% Total irradiation plot
figure(4);
plot(day,total_irr);
xlabel('Time (days)')
ylabel('Total Irradiation (N/kB)')
title ('Total Irradiation of a single fuel bundle as a function of
time')
legend(case_name)

hold off;

% Thermal irradiation plot
figure(5);
plot(day,thermal_irr);
xlabel('Time (days)')
ylabel('Thermal Irradiation (N/kB)')
title ('Thermal Irradiation of a single fuel bundle as a function of
time')
legend(case_name)

hold off;

% K-eff vs. Burnup plot
figure(6);
plot(burnup,k_eff);
xlabel('Burnup (MWD)')
ylabel('K-effective')
title ('K-effective of a single fuel bundle as a function of burnup')
hold on;
hline=refline(0,1);
set(hline,'Color','k')
legend(case_name)

hold off;

%------------------------------ CODE END -------------------------------
--%
%}
fclose('all');
```

Part 2: Output Comparison (Graphical Plot)

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%% AUTOMATION OF WIMS OUTPUT PLOTS %%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

% Created by: David Tom & Jason Song

clc;
clear;

case1='outputfile#1.txt’;
case1_name='outputfile#1';

case2='outputfile#2.txt';
case2_name='outputfile#2';

case3='outputfile#3.txt';
case3_name='outputfile#3';

case4='outputfile#4.txt';
case4_name='outputfile#4';

case5='outputfile#5.txt';
case5_name='outputfile#5';

case6='outputfile#6.txt';
case6_name='outputfile#6';

case7='outputfile#7.txt';
case7_name='outputfile#7';

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

fopen(case1);                                                    % Read Input file
string=fileread(case1);                                          % Convert file into
a single character string


% Identify locations where string says certain KEY words
keff_start=strfind(string,'       K-INFINITY');                  %Location
tag for k-eff
day_start=strfind(string,'TIME =');                              %Location
tag for number of days
burnup_start=strfind(string,'BURNUP =');                         %Location
tag for burnup
power_start=strfind(string,'POWER =');                           %Location
tag for power
```

```
total_irr_start=strfind(string,'INITIAL   IRRADIATION  =');     %Location
tag for total irradiation
thermal_irr_start=strfind(string,'THERMAL  IRRADIATION  =');    %Location
tag for thermal irradiation

% Setup loops to make matrix of elements %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%K-eff Loop
for i=1:1:length(keff_start);
k_eff(i)=str2num(string(keff_start(i)+18:keff_start(i)+24));
end

%Time Loop
for i=1:1:length(day_start)
    h=0;
for j=1:1:10

day_test=string(day_start(i)+j:day_start(i)+13);
numerical=isstrprop(day_test,'digit');

if numerical(1)==1 && h==0
    day(i)=str2num(day_test);
    h=1;
end
end
end

%Burnup Loop
for i=1:1:length(burnup_start)
    h=0;
for j=1:1:18

burnup_test=string(burnup_start(i)+j:burnup_start(i)+18);
numerical=isstrprop(burnup_test,'digit');

if numerical(1)==1 && h==0
    burnup(i)=str2num(burnup_test);
    h=1;
end
end
end

%Power Loop
for i=1:1:length(power_start)
    h=0;
for j=1:1:15

power_test=string(power_start(i)+j:power_start(i)+15);
numerical=isstrprop(power_test,'digit');

if numerical(1)==1 && h==0
    power(i)=str2num(power_test);
    h=1;
end
```

```
end
end

%Total Irradiation Loop
for i=1:1:length(total_irr_start)
    h=0;
for j=1:1:29

total_irr_test=string(total_irr_start(i)+j:total_irr_start(i)+29);
numerical=isstrprop(total_irr_test,'digit');

if numerical(1)==1 && h==0
    total_irr(i)=str2num(total_irr_test);
    h=1;
end
end
end

%Thermal Irradiation Loop
for i=1:1:length(thermal_irr_start)
    h=0;
for j=1:1:28

thermal_irr_test=string(thermal_irr_start(i)+j:thermal_irr_start(i)+28);
numerical=isstrprop(thermal_irr_test,'digit');

if numerical(1)==1 && h==0
    thermal_irr(i)=str2num(thermal_irr_test);
    h=1;
end
end
end

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%                  CASE                    2
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%


fopen(case2);                                           % Read Input file
string=fileread(case2);                                 % Convert file into
a single character string


% Identify locations where string says certain KEY words
keff_start=strfind(string,'       K-INFINITY');                  %Location
tag for k-eff
day_start=strfind(string,'TIME =');                              %Location
tag for number of days
burnup_start=strfind(string,'BURNUP =');                         %Location
tag for burnup
power_start=strfind(string,'POWER =');                           %Location
tag for power
total_irr_start=strfind(string,'INITIAL   IRRADIATION =');       %Location
tag for total irradiation
```

```
thermal_irr_start=strfind(string,'THERMAL  IRRADIATION  =');    %Location
tag for total irradiation

% Setup loops to make matrix of elements %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%K-eff Loop
for i=1:1:length(keff_start);
k_eff2(i)=str2num(string(keff_start(i)+18:keff_start(i)+24));
end

%Time Loop
for i=1:1:length(day_start)
    h=0;
for j=1:1:10

day_test=string(day_start(i)+j:day_start(i)+13);
numerical=isstrprop(day_test,'digit');

if numerical(1)==1 && h==0
    day2(i)=str2num(day_test);
    h=1;
end
end
end

%Burnup Loop
for i=1:1:length(burnup_start)
    h=0;
for j=1:1:18

burnup_test=string(burnup_start(i)+j:burnup_start(i)+18);
numerical=isstrprop(burnup_test,'digit');

if numerical(1)==1 && h==0
    burnup2(i)=str2num(burnup_test);
    h=1;
end
end
end

%Power Loop
for i=1:1:length(power_start)
    h=0;
for j=1:1:15

power_test=string(power_start(i)+j:power_start(i)+15);
numerical=isstrprop(power_test,'digit');

if numerical(1)==1 && h==0
    power2(i)=str2num(power_test);
    h=1;
end
end
end
```

```
%Total Irradiation Loop
for i=1:1:length(total_irr_start)
    h=0;
for j=1:1:29

total_irr_test=string(total_irr_start(i)+j:total_irr_start(i)+29);
numerical=isstrprop(total_irr_test,'digit');

if numerical(1)==1 && h==0
    total_irr2(i)=str2num(total_irr_test);
    h=1;
end
end
end

%Thermal Irradiation Loop
for i=1:1:length(thermal_irr_start)
    h=0;
for j=1:1:28

thermal_irr_test=string(thermal_irr_start(i)+j:thermal_irr_start(i)+28);
numerical=isstrprop(thermal_irr_test,'digit');

if numerical(1)==1 && h==0
    thermal_irr2(i)=str2num(thermal_irr_test);
    h=1;
end
end
end

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%                  CASE                  3
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%


fopen(case3);                                         % Read Input file
string=fileread(case3);                               % Convert file into
a single character string


% Identify locations where string says certain KEY words
keff_start=strfind(string,'       K-INFINITY');                   %Location
tag for k-eff
day_start=strfind(string,'TIME =');                               %Location
tag for number of days
burnup_start=strfind(string,'BURNUP =');                          %Location
tag for burnup
power_start=strfind(string,'POWER =');                            %Location
tag for power
total_irr_start=strfind(string,'INITIAL   IRRADIATION =');     %Location
tag for total irradiation
thermal_irr_start=strfind(string,'THERMAL  IRRADIATION  =');    %Location
tag for total irradiation
```

```
% Setup loops to make matrix of elements %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%K-eff Loop
for i=1:1:length(keff_start);
k_eff3(i)=str2num(string(keff_start(i)+18:keff_start(i)+24));
end

%Time Loop
for i=1:1:length(day_start)
    h=0;
for j=1:1:10

day_test=string(day_start(i)+j:day_start(i)+13);
numerical=isstrprop(day_test,'digit');

if numerical(1)==1 && h==0
    day3(i)=str2num(day_test);
    h=1;
end
end
end

%Burnup Loop
for i=1:1:length(burnup_start)
    h=0;
for j=1:1:18

burnup_test=string(burnup_start(i)+j:burnup_start(i)+18);
numerical=isstrprop(burnup_test,'digit');

if numerical(1)==1 && h==0
    burnup3(i)=str2num(burnup_test);
    h=1;
end
end
end

%Power Loop
for i=1:1:length(power_start)
    h=0;
for j=1:1:15

power_test=string(power_start(i)+j:power_start(i)+15);
numerical=isstrprop(power_test,'digit');

if numerical(1)==1 && h==0
    power3(i)=str2num(power_test);
    h=1;
end
end
end
```

```
%Total Irradiation Loop
for i=1:1:length(total_irr_start)
    h=0;
for j=1:1:29

total_irr_test=string(total_irr_start(i)+j:total_irr_start(i)+29);
numerical=isstrprop(total_irr_test,'digit');

if numerical(1)==1 && h==0
    total_irr3(i)=str2num(total_irr_test);
    h=1;
end
end
end

%Thermal Irradiation Loop
for i=1:1:length(thermal_irr_start)
    h=0;
for j=1:1:28

thermal_irr_test=string(thermal_irr_start(i)+j:thermal_irr_start(i)+28);
numerical=isstrprop(thermal_irr_test,'digit');

if numerical(1)==1 && h==0
    thermal_irr3(i)=str2num(thermal_irr_test);
    h=1;
end
end
end

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%                  CASE            4
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%


fopen(case4);                                       % Read Input file
string=fileread(case4);                             % Convert file into
a single character string


% Identify locations where string says certain KEY words
keff_start=strfind(string,'       K-INFINITY');               %Location
tag for k-eff
day_start=strfind(string,'TIME =');                            %Location
tag for number of days
burnup_start=strfind(string,'BURNUP =');                       %Location
tag for burnup
power_start=strfind(string,'POWER =');                         %Location
tag for power
total_irr_start=strfind(string,'INITIAL   IRRADIATION =');     %Location
tag for total irradiation
thermal_irr_start=strfind(string,'THERMAL  IRRADIATION  =');   %Location
tag for total irradiation
```

```
% Setup loops to make matrix of elements %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%K-eff Loop
for i=1:1:length(keff_start);
k_eff4(i)=str2num(string(keff_start(i)+18:keff_start(i)+24));
end

%Time Loop
for i=1:1:length(day_start)
    h=0;
for j=1:1:10

day_test=string(day_start(i)+j:day_start(i)+13);
numerical=isstrprop(day_test,'digit');

if numerical(1)==1 && h==0
    day4(i)=str2num(day_test);
    h=1;
end
end
end

%Burnup Loop
for i=1:1:length(burnup_start)
    h=0;
for j=1:1:18

burnup_test=string(burnup_start(i)+j:burnup_start(i)+18);
numerical=isstrprop(burnup_test,'digit');

if numerical(1)==1 && h==0
    burnup4(i)=str2num(burnup_test);
    h=1;
end
end
end

%Power Loop
for i=1:1:length(power_start)
    h=0;
for j=1:1:15

power_test=string(power_start(i)+j:power_start(i)+15);
numerical=isstrprop(power_test,'digit');

if numerical(1)==1 && h==0
    power4(i)=str2num(power_test);
    h=1;
end
end
end

%Total Irradiation Loop
for i=1:1:length(total_irr_start)
```

```
    h=0;
for j=1:1:29

total_irr_test=string(total_irr_start(i)+j:total_irr_start(i)+29);
numerical=isstrprop(total_irr_test,'digit');

if numerical(1)==1 && h==0
    total_irr4(i)=str2num(total_irr_test);
    h=1;
end
end
end

%Thermal Irradiation Loop
for i=1:1:length(thermal_irr_start)
    h=0;
for j=1:1:28

thermal_irr_test=string(thermal_irr_start(i)+j:thermal_irr_start(i)+28);
numerical=isstrprop(thermal_irr_test,'digit');

if numerical(1)==1 && h==0
    thermal_irr4(i)=str2num(thermal_irr_test);
    h=1;
end
end
end

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%                  CASE                    5
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%


fopen(case5);                                              % Read Input file
string=fileread(case5);                                    % Convert file into
a single character string


% Identify locations where string says certain KEY words
keff_start=strfind(string,'       K-INFINITY');                     %Location
tag for k-eff
day_start=strfind(string,'TIME =');                                 %Location
tag for number of days
burnup_start=strfind(string,'BURNUP =');                            %Location
tag for burnup
power_start=strfind(string,'POWER =');                              %Location
tag for power
total_irr_start=strfind(string,'INITIAL   IRRADIATION  =');     %Location
tag for total irradiation
thermal_irr_start=strfind(string,'THERMAL  IRRADIATION  =');    %Location
tag for total irradiation

% Setup loops to make matrix of elements %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

```
%K-eff Loop
for i=1:1:length(keff_start);
k_eff5(i)=str2num(string(keff_start(i)+18:keff_start(i)+24));
end

%Time Loop
for i=1:1:length(day_start)
    h=0;
for j=1:1:10

day_test=string(day_start(i)+j:day_start(i)+13);
numerical=isstrprop(day_test,'digit');

if numerical(1)==1 && h==0
    day5(i)=str2num(day_test);
    h=1;
end
end
end

%Burnup Loop
for i=1:1:length(burnup_start)
    h=0;
for j=1:1:18

burnup_test=string(burnup_start(i)+j:burnup_start(i)+18);
numerical=isstrprop(burnup_test,'digit');

if numerical(1)==1 && h==0
    burnup5(i)=str2num(burnup_test);
    h=1;
end
end
end

%Power Loop
for i=1:1:length(power_start)
    h=0;
for j=1:1:15

power_test=string(power_start(i)+j:power_start(i)+15);
numerical=isstrprop(power_test,'digit');

if numerical(1)==1 && h==0
    power5(i)=str2num(power_test);
    h=1;
end
end
end

%Total Irradiation Loop
for i=1:1:length(total_irr_start)
    h=0;
for j=1:1:29
```

```
total_irr_test=string(total_irr_start(i)+j:total_irr_start(i)+29);
numerical=isstrprop(total_irr_test,'digit');

if numerical(1)==1 && h==0
    total_irr5(i)=str2num(total_irr_test);
    h=1;
end
end
end

%Thermal Irradiation Loop
for i=1:1:length(thermal_irr_start)
    h=0;
for j=1:1:28

thermal_irr_test=string(thermal_irr_start(i)+j:thermal_irr_start(i)+28);
numerical=isstrprop(thermal_irr_test,'digit');

if numerical(1)==1 && h==0
    thermal_irr5(i)=str2num(thermal_irr_test);
    h=1;
end
end
end

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%                   CASE                 6
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%


fopen(case6);                                        % Read Input file
string=fileread(case6);                              % Convert file into
a single character string


% Identify locations where string says certain KEY words
keff_start=strfind(string,'       K-INFINITY');                %Location
tag for k-eff
day_start=strfind(string,'TIME =');                            %Location
tag for number of days
burnup_start=strfind(string,'BURNUP =');                       %Location
tag for burnup
power_start=strfind(string,'POWER =');                         %Location
tag for power
total_irr_start=strfind(string,'INITIAL   IRRADIATION =');     %Location
tag for total irradiation
thermal_irr_start=strfind(string,'THERMAL  IRRADIATION  =');   %Location
tag for total irradiation

% Setup loops to make matrix of elements %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

```
%K-eff Loop
for i=1:1:length(keff_start);
k_eff6(i)=str2num(string(keff_start(i)+18:keff_start(i)+24));
end

%Time Loop
for i=1:1:length(day_start)
    h=0;
for j=1:1:10

day_test=string(day_start(i)+j:day_start(i)+13);
numerical=isstrprop(day_test,'digit');

if numerical(1)==1 && h==0
    day6(i)=str2num(day_test);
    h=1;
end
end
end

%Burnup Loop
for i=1:1:length(burnup_start)
    h=0;
for j=1:1:18

burnup_test=string(burnup_start(i)+j:burnup_start(i)+18);
numerical=isstrprop(burnup_test,'digit');

if numerical(1)==1 && h==0
    burnup6(i)=str2num(burnup_test);
    h=1;
end
end
end

%Power Loop
for i=1:1:length(power_start)
    h=0;
for j=1:1:15

power_test=string(power_start(i)+j:power_start(i)+15);
numerical=isstrprop(power_test,'digit');

if numerical(1)==1 && h==0
    power6(i)=str2num(power_test);
    h=1;
end
end
end

%Total Irradiation Loop
for i=1:1:length(total_irr_start)
    h=0;
for j=1:1:29
```

```
total_irr_test=string(total_irr_start(i)+j:total_irr_start(i)+29);
numerical=isstrprop(total_irr_test,'digit');

if numerical(1)==1 && h==0
    total_irr6(i)=str2num(total_irr_test);
    h=1;
end
end
end

%Thermal Irradiation Loop
for i=1:1:length(thermal_irr_start)
    h=0;
for j=1:1:28

thermal_irr_test=string(thermal_irr_start(i)+j:thermal_irr_start(i)+28);
numerical=isstrprop(thermal_irr_test,'digit');

if numerical(1)==1 && h==0
    thermal_irr6(i)=str2num(thermal_irr_test);
    h=1;
end
end
end

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%                              CASE                7
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%


fopen(case7);                                               % Read Input file
string=fileread(case7);                                     % Convert file into
a single character string


% Identify locations where string says certain KEY words
keff_start=strfind(string,'       K-INFINITY');                       %Location
tag for k-eff
day_start=strfind(string,'TIME =');                                   %Location
tag for number of days
burnup_start=strfind(string,'BURNUP =');                              %Location
tag for burnup
power_start=strfind(string,'POWER =');                                %Location
tag for power
total_irr_start=strfind(string,'INITIAL   IRRADIATION =');            %Location
tag for total irradiation
thermal_irr_start=strfind(string,'THERMAL  IRRADIATION  =');          %Location
tag for total irradiation

% Setup loops to make matrix of elements %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%K-eff Loop
for i=1:1:length(keff_start);
```

```
k_eff7(i)=str2num(string(keff_start(i)+18:keff_start(i)+24));
end

%Time Loop
for i=1:1:length(day_start)
    h=0;
for j=1:1:10

day_test=string(day_start(i)+j:day_start(i)+13);
numerical=isstrprop(day_test,'digit');

if numerical(1)==1 && h==0
    day7(i)=str2num(day_test);
    h=1;
end
end
end

%Burnup Loop
for i=1:1:length(burnup_start)
    h=0;
for j=1:1:18

burnup_test=string(burnup_start(i)+j:burnup_start(i)+18);
numerical=isstrprop(burnup_test,'digit');

if numerical(1)==1 && h==0
    burnup7(i)=str2num(burnup_test);
    h=1;
end
end
end

%Power Loop
for i=1:1:length(power_start)
    h=0;
for j=1:1:15

power_test=string(power_start(i)+j:power_start(i)+15);
numerical=isstrprop(power_test,'digit');

if numerical(1)==1 && h==0
    power7(i)=str2num(power_test);
    h=1;
end
end
end

%Total Irradiation Loop
for i=1:1:length(total_irr_start)
    h=0;
for j=1:1:29

total_irr_test=string(total_irr_start(i)+j:total_irr_start(i)+29);
numerical=isstrprop(total_irr_test,'digit');
```

```
if numerical(1)==1 && h==0
    total_irr7(i)=str2num(total_irr_test);
    h=1;
end
end
end

%Thermal Irradiation Loop
for i=1:1:length(thermal_irr_start)
    h=0;
for j=1:1:28

thermal_irr_test=string(thermal_irr_start(i)+j:thermal_irr_start(i)+28);
numerical=isstrprop(thermal_irr_test,'digit');

if numerical(1)==1 && h==0
    thermal_irr7(i)=str2num(thermal_irr_test);
    h=1;
end
end
end




%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%
%%%%%%%%%%%%%%%%%%%%%%%       Plot       the       respective       graphs
%%%%%%%%%%%%%%%%%%%%%%%%

hold off;

% K-effective plot
figure(1);
plot(day,k_eff)
xlabel('Time (days)')
ylabel('K-effective')
title ('K-effective of a single fuel bundle as a function of time')
hold on;
plot(day2,k_eff2,'r')
hold on;
plot(day3,k_eff3,'g')
hold on;
plot(day4,k_eff4,'c')
hold on;
plot(day5,k_eff5,'m')
hold on;
plot(day6,k_eff6,'y')
hold on;
plot(day7,k_eff7,'k')
hold on;
hline=refline(0,1);
set(hline,'Color','k')
```

```
legend(case1_name,case2_name,case3_name,case4_name,case5_name,case6_name
,case7_name)

hold off;

% Burnup plot
figure(2);
plot(day,burnup);
xlabel('Time (days)')
ylabel('burnup (MWD)')
title ('Burnup of a single fuel bundle as a function of time')
hold on;
plot(day2,burnup2,'r')
hold on;
plot(day3,burnup3,'g')
hold on;
plot(day4,burnup4,'c')
hold on;
plot(day5,burnup5,'m')
hold on;
plot(day6,burnup6,'y')
hold on;
plot(day7,burnup7,'k')
legend(case1_name,case2_name,case3_name,case4_name,case5_name,case6_name
,case7_name)

hold off;

% Power plot
figure(3);
plot(day,power)
xlabel('Time (days)')
ylabel('Power (MW)')
title ('Power of a single fuel bundle as a function of time')
hold on;
plot(day2,power2,'r')
hold on;
plot(day3,power3,'g')
hold on;
plot(day4,power4,'c')
hold on;
plot(day5,power5,'m')
hold on;
plot(day6,power6,'y')
hold on;
plot(day7,power7,'k')
legend(case1_name,case2_name,case3_name,case4_name,case5_name,case6_name
,case7_name)

hold off;

% Total irradiation plot
figure(4);
plot(day,total_irr);
xlabel('Time (days)')
ylabel('Total Irradiation (N/kB)')
```

```
title ('Total Irradiation of a single fuel bundle as a function of
time')
hold on;
plot(day2,total_irr2,'r')
hold on;
plot(day3,total_irr3,'g')
hold on;
plot(day4,total_irr4,'c')
hold on;
plot(day5,total_irr5,'m')
hold on;
plot(day6,total_irr6,'y')
hold on;
plot(day7,total_irr7,'k')
legend(case1_name,case2_name,case3_name,case4_name,case5_name,case6_name
,case7_name)
hold off;

% Thermal irradiation plot
figure(5);
plot(day,thermal_irr);
xlabel('Time (days)')
ylabel('Thermal Irradiation (N/kB)')
title ('Thermal Irradiation of a single fuel bundle as a function of
time')
hold on;
plot(day2,thermal_irr2,'r')
hold on;
plot(day3,thermal_irr3,'g')
hold on;
plot(day4,thermal_irr4,'c')
hold on;
plot(day5,thermal_irr5,'m')
hold on;
plot(day6,thermal_irr6,'y')
hold on;
plot(day7,thermal_irr7,'k')
legend(case1_name,case2_name,case3_name,case4_name,case5_name,case6_name
,case7_name)

hold off;

% K-eff vs. Burnup plot
figure(6);
plot(burnup,k_eff);
xlabel('Burnup (MWD)')
ylabel('K-effective')
title ('K-effective of a single fuel bundle as a function of burnup')
hold on;
plot(burnup2,k_eff2,'r');
hold on;
plot(burnup3,k_eff3,'g');
hold on;
plot(burnup4,k_eff4,'c');
hold on;
plot(burnup5,k_eff5,'m');
```

```
hold on;
plot(burnup6,k_eff6,'y');
hold on;
plot(burnup7,k_eff7,'k');
hold on;
hline=refline(0,1);
set(hline,'Color','k')
legend(case1_name,case2_name,case3_name,case4_name,case5_name,case6_name
,case7_name)

fclose('all');
```

Appendix B

Automatic Refuelling and Core Monitoring Program

```
clc;
clear;

disp('AUTO_RFSP: All-access and post processing utility for RFSP')
disp('This program was created By: Jae Song & David Tom')
disp('.') % spacing till next comment

Number_Of_FPDs=400;
loopcrit=0; % initialization for while loop criterion for termination
loopmax=Number_Of_FPDs/0.25; % max number of iterations

plustilt=importdata('C:\0_DAY_STARTPOINT_CDrive\Loaded_Data\plustilt.mat
'); % if positive dir channel has been fuelled
minustilt=importdata('C:\0_DAY_STARTPOINT_CDrive\Loaded_Data\minustilt.m
at'); % if negative dir channel has been fuelled
tiltmap=importdata('C:\0_DAY_STARTPOINT_CDrive\Loaded_Data\tiltmap.mat')
; % map of fuelling directions of each channel
Time_Aver_NP=importdata('C:\0_DAY_STARTPOINT_CDrive\Loaded_Data\Time_Ave
r_NP.mat'); % time average power distribution without absorbers
Time_Aver_WP=importdata('C:\0_DAY_STARTPOINT_CDrive\Loaded_Data\Time_Ave
r_WP.mat'); % time average power distribution with absorbers
Reference_Core=importdata('C:\0_DAY_STARTPOINT_CDrive\Loaded_Data\Refere
nce_Core.mat'); % Reference Time-average core power distribution

CPPF=zeros(24:24); % initialization for CPPF

Day=zeros(1,loopmax); % Initialization for matrix containing all FPDs
yChosenrecord=zeros(1,loopmax); % Initialization for matrix containing
past fuelled channels
xChosenrecord=zeros(1,loopmax);
ChPowerplot=zeros(24,24,loopmax); % Initialization for matrix containing
all Channel Powers over all FPD
MBPowerplot=zeros(24,24,loopmax); % Initialization for matrix containing
all Bundle Powers over all FPD
Kchangeplot=zeros(24,24,loopmax); % similar to above
Burnupplot=zeros(24,24,loopmax); % similar to above
RBurnupplot=zeros(24,24,loopmax); % similar to above
ATiltplot=zeros(24,24,loopmax); % similar to above
CPPFplot=zeros(24,24,loopmax); % similar to above
TotalScoreplot=zeros(24,24,loopmax); % similar to above
MaxChPplot=zeros(1,loopmax); % similar to above
MaxMBdPplot=zeros(1,loopmax); % similar to above
FormFactorplot=zeros(1,loopmax); % similar to above
Keffplot=zeros(1,loopmax); % similar to above
RDeclineplot=zeros(1,loopmax); % similar to above
e_reactplot=zeros(1,loopmax); % similar to above
avgzcrplot=zeros(1,loopmax); % similar to above
zcr1plot=zeros(1,loopmax); % similar to above
zcr2plot=zeros(1,loopmax); % similar to above
```

```
zcr3plot=zeros(1,loopmax); % similar to above
zcr4plot=zeros(1,loopmax); % similar to above
zcr5plot=zeros(1,loopmax); % similar to above
zcr6plot=zeros(1,loopmax); % similar to above
zcr7plot=zeros(1,loopmax); % similar to above
zcr8plot=zeros(1,loopmax); % similar to above
zcr9plot=zeros(1,loopmax); % similar to above
zcr10plot=zeros(1,loopmax); % similar to above
zcr11plot=zeros(1,loopmax); % similar to above
zcr12plot=zeros(1,loopmax); % similar to above
zcr13plot=zeros(1,loopmax); % similar to above
zcr14plot=zeros(1,loopmax); % similar to above


while loopcrit <= loopmax;
timerVal=tic;
Day(1,loopcrit+1)=0.25*(loopcrit);

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%          DATA           IMPORT           BEGIN
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%

display('Importing Core Data from Previous RFSP Output...')
disp('.') % spacing till next comment

fopen('rfsp_output');                  % Read Input file
string=fileread('rfsp_output');           % Convert file into a single
character string

modstring1=string;
modstring2=regexprep(modstring1, '1     ','1   0 ');
modstring3=regexprep(modstring2, '2     ','2   0 ');
modstring4=regexprep(modstring3, '3     ','3   0 ');
modstring5=regexprep(modstring4, '4     ','4   0 ');
modstring6=regexprep(modstring5, '5     ','5   0 ');
modstring7=regexprep(modstring6, '6     ','6   0 ');
modstring8=regexprep(modstring7, '7     ','7   0 ');
modstring9=regexprep(modstring8, '8     ','8   0 ');
modstring10=regexprep(modstring9, '9     ','9   0 ');
modstring11=regexprep(modstring10, '0     ','0   0 ');

modstring12=regexprep(modstring11, '     1',' 0   1');
modstring13=regexprep(modstring12, '     2',' 0   2');
modstring14=regexprep(modstring13, '     3',' 0   3');
modstring15=regexprep(modstring14, '     4',' 0   4');
modstring16=regexprep(modstring15, '     5',' 0   5');
modstring17=regexprep(modstring16, '     6',' 0   6');
modstring18=regexprep(modstring17, '     7',' 0   7');
```

```
modstring19=regexprep(modstring18, '     8',' 0   8');
modstring20=regexprep(modstring19, '     9',' 0   9');
modstring21=regexprep(modstring20, ':        0',':   0    0');
modstring22=regexprep(modstring21, '     0',' 0   0');
modstring23=regexprep(modstring22, '     -1',' 0   -1');
modstring24=regexprep(modstring23, '     -2',' 0   -2');
modstring25=regexprep(modstring24, '     -3',' 0   -3');
modstring26=regexprep(modstring25, '     -4',' 0   -4');
modstring27=regexprep(modstring26, '     -5',' 0   -5');
modstring28=regexprep(modstring27, '     -6',' 0   -6');
modstring29=regexprep(modstring28, '     -7',' 0   -7');
modstring30=regexprep(modstring29, '     -8',' 0   -8');
modstring31=regexprep(modstring30, '     -9',' 0   -9');

modstring32=regexprep(modstring31, '          :    : ',' :  0 :
');
modstring33=regexprep(modstring32, ' :      :            ',' :   0 :
');

modstring34=regexprep(modstring33, ': 0 ',':  0');
modstring35=regexprep(modstring34, '0   0','0    ');

string=modstring35;

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%
%%%%%%%%%%%%%%%%%%%%%%%%          POWER          DISTRIBUTION          AUTO
%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%

% Identify locations where string says certain KEY words
Ch_Power_start=strfind(string,'CHANNEL     POWER     DISTRIBUTION(kW)');
%Location tag for Channel Powers

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Setup Matrix of Channel Power Values %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

Ch_Power_region=string(Ch_Power_start+300:Ch_Power_start+6250);
%% GOOD GUESS
start_point=300;
end_point=6250;

count=0;

for j=1:1:length(Ch_Power_region)-5
window=string(Ch_Power_start+start_point+j:Ch_Power_start+start_point+j+
5);
numerical=isstrprop(window,'digit');

if    numerical(1)==0    &&    numerical(2)==0    &&    numerical(3)==0    &&
numerical(4)==1 && numerical(5)==0 && numerical(6)==0
    count=count+1;
```

```
window2=string(Ch_Power_start+start_point+j+3:Ch_Power_start+start_point
+j+3);
    Ch_Power_matrix(count)=str2num(window2);
end

if   numerical(1)==0   &&   numerical(2)==0   &&   numerical(3)==1   &&
numerical(4)==1 && numerical(5)==0 && numerical(6)==0
    count=count+1;

window2=string(Ch_Power_start+start_point+j+2:Ch_Power_start+start_point
+j+3);
    Ch_Power_matrix(count)=str2num(window2);
end

if   numerical(1)==0   &&   numerical(2)==0   &&   numerical(3)==1   &&
numerical(4)==1 && numerical(5)==1 && numerical(6)==0
    count=count+1;

window2=string(Ch_Power_start+start_point+j+2:Ch_Power_start+start_point
+j+4);
    Ch_Power_matrix(count)=str2num(window2);
end

if   numerical(1)==0   &&   numerical(2)==1   &&   numerical(3)==1   &&
numerical(4)==1 && numerical(5)==1 && numerical(6)==0
    count=count+1;

window2=string(Ch_Power_start+start_point+j+1:Ch_Power_start+start_point
+j+4);
    Ch_Power_matrix(count)=str2num(window2);
end

end

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Organize matrix into actual core geometry %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

Ch_Power_geometry=zeros(24,24);

if length(Ch_Power_matrix)==480
%Row A
for i=1:1:10
    Ch_Power_geometry(1,i+7)=Ch_Power_matrix(i);
end

%Row B
for i=11:1:24
    Ch_Power_geometry(2,i-5)=Ch_Power_matrix(i);
end

%Row C
for i=25:1:40
    Ch_Power_geometry(3,i-20)=Ch_Power_matrix(i);
```

```
end

%Row D
for i=41:1:58
    Ch_Power_geometry(4,i-37)=Ch_Power_matrix(i);
end

%Row E
for i=59:1:78
    Ch_Power_geometry(5,i-56)=Ch_Power_matrix(i);
end

%Row F
for i=79:1:100
    Ch_Power_geometry(6,i-77)=Ch_Power_matrix(i);
end

%Row G
for i=101:1:122
    Ch_Power_geometry(7,i-99)=Ch_Power_matrix(i);
end

%Row H
for i=123:1:144
    Ch_Power_geometry(8,i-121)=Ch_Power_matrix(i);
end

%Row J
for i=145:1:168
    Ch_Power_geometry(9,i-144)=Ch_Power_matrix(i);
end

%Row K
for i=169:1:192
    Ch_Power_geometry(10,i-168)=Ch_Power_matrix(i);
end

%Row L
for i=193:1:216
    Ch_Power_geometry(11,i-192)=Ch_Power_matrix(i);
end

%Row M
for i=217:1:240
    Ch_Power_geometry(12,i-216)=Ch_Power_matrix(i);
end

%Row N
for i=241:1:264
    Ch_Power_geometry(13,i-240)=Ch_Power_matrix(i);
end

%Row O
for i=265:1:288
    Ch_Power_geometry(14,i-264)=Ch_Power_matrix(i);
```

```
end

%Row P
for i=289:1:312
    Ch_Power_geometry(15,i-288)=Ch_Power_matrix(i);
end

%Row Q
for i=313:1:336
    Ch_Power_geometry(16,i-312)=Ch_Power_matrix(i);
end

%Row R
for i=337:1:358
    Ch_Power_geometry(17,i-335)=Ch_Power_matrix(i);
end

%Row S
for i=359:1:380
    Ch_Power_geometry(18,i-357)=Ch_Power_matrix(i);
end

%Row T
for i=381:1:402
    Ch_Power_geometry(19,i-379)=Ch_Power_matrix(i);
end

%Row U
for i=403:1:422
    Ch_Power_geometry(20,i-400)=Ch_Power_matrix(i);
end

%Row V
for i=423:1:440
    Ch_Power_geometry(21,i-419)=Ch_Power_matrix(i);
end

%Row W
for i=441:1:456
    Ch_Power_geometry(22,i-436)=Ch_Power_matrix(i);
end

%Row X
for i=457:1:470
    Ch_Power_geometry(23,i-451)=Ch_Power_matrix(i);
end

%Row Y
for i=471:1:480
    Ch_Power_geometry(24,i-463)=Ch_Power_matrix(i);
end

else
    disp('FATAL ERROR IN CH_POWER: NUMBER OF MATRIX ELEMENTS DOES NOT
EQUAL 480')
```

```
    disp('# of matrix elements is:')
    disp(length(Ch_Power_matrix))
    pause
end

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%
%%%%%%%%%%%%%%%%%%%%%%%       AVERAGE       EXIT       BURNUP       AUTO
%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%

% Identify locations where string says certain KEY words
Ave_Burnup_start=strfind(string,'AVERAGE          EXIT          BURNUP');
%Location tag for Channel Powers

% Day Location
Ave_Burnup_FPD=str2num(string(Ave_Burnup_start+58:Ave_Burnup_start+58));

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Setup Matrix of Exit Burnup Values %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

Ave_Burnup_region=string(Ave_Burnup_start+300:Ave_Burnup_start+6250);
%% GOOD GUESS
start_point=300;
end_point=6250;

count=0;

for j=1:1:length(Ave_Burnup_region)
window=string(Ave_Burnup_start+start_point+j:Ave_Burnup_start+start_poin
t+j+5);
numerical=isstrprop(window,'digit');

if   numerical(1)==0   &&   numerical(2)==0   &&   numerical(3)==0   &&
numerical(4)==1 && numerical(5)==0 && numerical(6)==0
    count=count+1;

window2=string(Ave_Burnup_start+start_point+j+3:Ave_Burnup_start+start_p
oint+j+3);
    Ave_Burnup_matrix(count)=str2num(window2);
end

if   numerical(1)==0   &&   numerical(2)==0   &&   numerical(3)==1   &&
numerical(4)==1 && numerical(5)==0 && numerical(6)==0
    count=count+1;

window2=string(Ave_Burnup_start+start_point+j+2:Ave_Burnup_start+start_p
oint+j+3);
    Ave_Burnup_matrix(count)=str2num(window2);
end

if   numerical(1)==0   &&   numerical(2)==0   &&   numerical(3)==1   &&
numerical(4)==1 && numerical(5)==1 && numerical(6)==0
```

```
    count=count+1;

window2=string(Ave_Burnup_start+start_point+j+2:Ave_Burnup_start+start_p
oint+j+4);
    Ave_Burnup_matrix(count)=str2num(window2);
end

if   numerical(1)==0   &&   numerical(2)==1   &&   numerical(3)==1   &&
numerical(4)==1 && numerical(5)==1 && numerical(6)==0
    count=count+1;

window2=string(Ave_Burnup_start+start_point+j+1:Ave_Burnup_start+start_p
oint+j+4);
    Ave_Burnup_matrix(count)=str2num(window2);
end

end

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Organize matrix into actual core geometry %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

Ave_Burnup_geometry=zeros(24,24);

if length(Ave_Burnup_matrix)==480
%Row A
for i=1:1:10
    Ave_Burnup_geometry(1,i+7)=Ave_Burnup_matrix(i);
end

%Row B
for i=11:1:24
    Ave_Burnup_geometry(2,i-5)=Ave_Burnup_matrix(i);
end

%Row C
for i=25:1:40
    Ave_Burnup_geometry(3,i-20)=Ave_Burnup_matrix(i);
end

%Row D
for i=41:1:58
    Ave_Burnup_geometry(4,i-37)=Ave_Burnup_matrix(i);
end

%Row E
for i=59:1:78
    Ave_Burnup_geometry(5,i-56)=Ave_Burnup_matrix(i);
end

%Row F
for i=79:1:100
    Ave_Burnup_geometry(6,i-77)=Ave_Burnup_matrix(i);
end
```

```
%Row G
for i=101:1:122
    Ave_Burnup_geometry(7,i-99)=Ave_Burnup_matrix(i);
end

%Row H
for i=123:1:144
    Ave_Burnup_geometry(8,i-121)=Ave_Burnup_matrix(i);
end

%Row J
for i=145:1:168
    Ave_Burnup_geometry(9,i-144)=Ave_Burnup_matrix(i);
end

%Row K
for i=169:1:192
    Ave_Burnup_geometry(10,i-168)=Ave_Burnup_matrix(i);
end

%Row L
for i=193:1:216
    Ave_Burnup_geometry(11,i-192)=Ave_Burnup_matrix(i);
end

%Row M
for i=217:1:240
    Ave_Burnup_geometry(12,i-216)=Ave_Burnup_matrix(i);
end

%Row N
for i=241:1:264
    Ave_Burnup_geometry(13,i-240)=Ave_Burnup_matrix(i);
end

%Row O
for i=265:1:288
    Ave_Burnup_geometry(14,i-264)=Ave_Burnup_matrix(i);
end

%Row P
for i=289:1:312
    Ave_Burnup_geometry(15,i-288)=Ave_Burnup_matrix(i);
end

%Row Q
for i=313:1:336
    Ave_Burnup_geometry(16,i-312)=Ave_Burnup_matrix(i);
end

%Row R
for i=337:1:358
    Ave_Burnup_geometry(17,i-335)=Ave_Burnup_matrix(i);
end
```

```
%Row S
for i=359:1:380
    Ave_Burnup_geometry(18,i-357)=Ave_Burnup_matrix(i);
end

%Row T
for i=381:1:402
    Ave_Burnup_geometry(19,i-379)=Ave_Burnup_matrix(i);
end

%Row U
for i=403:1:422
    Ave_Burnup_geometry(20,i-400)=Ave_Burnup_matrix(i);
end

%Row V
for i=423:1:440
    Ave_Burnup_geometry(21,i-419)=Ave_Burnup_matrix(i);
end

%Row W
for i=441:1:456
    Ave_Burnup_geometry(22,i-436)=Ave_Burnup_matrix(i);
end

%Row X
for i=457:1:470
    Ave_Burnup_geometry(23,i-451)=Ave_Burnup_matrix(i);
end

%Row Y
for i=471:1:480
    Ave_Burnup_geometry(24,i-463)=Ave_Burnup_matrix(i);
end

else
    disp('FATAL ERROR IN AVE_BURNUP: NUMBER OF MATRIX ELEMENTS DOES NOT
EQUAL 480')
    disp('# of matrix elements is:')
    disp(length(Ave_Burnup_matrix))
    pause
end

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%
%%%%%%%%%%%%%%%%%%%%%%%%      BURNUP      OVER      T.A      BURNUP      AUTO
%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%


% Identify locations where string says certain KEY words
Burnup_TA_start=strfind(string,'BURNUP            OVER            T.A');
%Location tag for Channel Powers
```

```
% Day Location
%Ave_Burnup_FPD=str2num(string(Ave_Burnup_start+58:Ave_Burnup_start+58))

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Setup Matrix of Burnup Over TA Values %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

Burnup_TA_region=string(Burnup_TA_start+300:Burnup_TA_start+6250);
%% GOOD GUESS
start_point=300;
end_point=6250;

count=0;

for j=1:1:length(Burnup_TA_region)
window=string(Burnup_TA_start+start_point+j:Burnup_TA_start+start_point+
j+5);
numerical=isstrprop(window,'digit');

if   numerical(1)==0   &&   numerical(2)==0   &&   numerical(3)==0   &&
numerical(4)==1 && numerical(5)==0 && numerical(6)==0
    count=count+1;

window2=string(Burnup_TA_start+start_point+j+3:Burnup_TA_start+start_poi
nt+j+3);
    Burnup_TA_matrix(count)=str2num(window2);
end

if   numerical(1)==0   &&   numerical(2)==0   &&   numerical(3)==1   &&
numerical(4)==1 && numerical(5)==0 && numerical(6)==0
    count=count+1;

window2=string(Burnup_TA_start+start_point+j+2:Burnup_TA_start+start_poi
nt+j+3);
    Burnup_TA_matrix(count)=str2num(window2);
end

if   numerical(1)==0   &&   numerical(2)==0   &&   numerical(3)==1   &&
numerical(4)==1 && numerical(5)==1 && numerical(6)==0
    count=count+1;

window2=string(Burnup_TA_start+start_point+j+2:Burnup_TA_start+start_poi
nt+j+4);
    Burnup_TA_matrix(count)=str2num(window2);
end

if   numerical(1)==0   &&   numerical(2)==1   &&   numerical(3)==1   &&
numerical(4)==1 && numerical(5)==1 && numerical(6)==0
    count=count+1;

window2=string(Burnup_TA_start+start_point+j+1:Burnup_TA_start+start_poi
nt+j+4);
    Burnup_TA_matrix(count)=str2num(window2);
end
```

```
end

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Organize matrix into actual core geometry %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

Burnup_TA_geometry=zeros(24,24);

if length(Burnup_TA_matrix)==480
%Row A
for i=1:1:10
    Burnup_TA_geometry(1,i+7)=Burnup_TA_matrix(i);
end

%Row B
for i=11:1:24
    Burnup_TA_geometry(2,i-5)=Burnup_TA_matrix(i);
end

%Row C
for i=25:1:40
    Burnup_TA_geometry(3,i-20)=Burnup_TA_matrix(i);
end

%Row D
for i=41:1:58
    Burnup_TA_geometry(4,i-37)=Burnup_TA_matrix(i);
end

%Row E
for i=59:1:78
    Burnup_TA_geometry(5,i-56)=Burnup_TA_matrix(i);
end

%Row F
for i=79:1:100
    Burnup_TA_geometry(6,i-77)=Burnup_TA_matrix(i);
end

%Row G
for i=101:1:122
    Burnup_TA_geometry(7,i-99)=Burnup_TA_matrix(i);
end

%Row H
for i=123:1:144
    Burnup_TA_geometry(8,i-121)=Burnup_TA_matrix(i);
end

%Row J
for i=145:1:168
    Burnup_TA_geometry(9,i-144)=Burnup_TA_matrix(i);
end
```

```
%Row K
for i=169:1:192
    Burnup_TA_geometry(10,i-168)=Burnup_TA_matrix(i);
end

%Row L
for i=193:1:216
    Burnup_TA_geometry(11,i-192)=Burnup_TA_matrix(i);
end

%Row M
for i=217:1:240
    Burnup_TA_geometry(12,i-216)=Burnup_TA_matrix(i);
end

%Row N
for i=241:1:264
    Burnup_TA_geometry(13,i-240)=Burnup_TA_matrix(i);
end

%Row O
for i=265:1:288
    Burnup_TA_geometry(14,i-264)=Burnup_TA_matrix(i);
end

%Row P
for i=289:1:312
    Burnup_TA_geometry(15,i-288)=Burnup_TA_matrix(i);
end

%Row Q
for i=313:1:336
    Burnup_TA_geometry(16,i-312)=Burnup_TA_matrix(i);
end

%Row R
for i=337:1:358
    Burnup_TA_geometry(17,i-335)=Burnup_TA_matrix(i);
end

%Row S
for i=359:1:380
    Burnup_TA_geometry(18,i-357)=Burnup_TA_matrix(i);
end

%Row T
for i=381:1:402
    Burnup_TA_geometry(19,i-379)=Burnup_TA_matrix(i);
end

%Row U
for i=403:1:422
    Burnup_TA_geometry(20,i-400)=Burnup_TA_matrix(i);
end
```

```
%Row V
for i=423:1:440
    Burnup_TA_geometry(21,i-419)=Burnup_TA_matrix(i);
end

%Row W
for i=441:1:456
    Burnup_TA_geometry(22,i-436)=Burnup_TA_matrix(i);
end

%Row X
for i=457:1:470
    Burnup_TA_geometry(23,i-451)=Burnup_TA_matrix(i);
end

%Row Y
for i=471:1:480
    Burnup_TA_geometry(24,i-463)=Burnup_TA_matrix(i);
end

else
    disp('FATAL ERROR IN BURNUP_TA: NUMBER OF MATRIX ELEMENTS DOES NOT
EQUAL 480')
    disp('# of matrix elements is:')
    disp(length(Burnup_TA_matrix))
    pause
end

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%
%%%%%%%%%%%%%%%%%%%%%%%%%     MAX     BUNDLE     PWR     /     CHNL     AUTO
%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%



% Identify locations where string says certain KEY words
Max_Bndl_start=strfind(string,'MAX    BUNDLE    POWER    PER    CHNL');
%Location tag for Channel Powers

% Day Location
%Max_Bndl_FPD=str2num(string(Max_Bndl_start+58:Max_Bndl_start+58));

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Setup Matrix of Channel Power Values %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

Max_Bndl_region=string(Max_Bndl_start+300:Max_Bndl_start+6250);
%% GOOD GUESS
start_point=300;
end_point=6250;

count=0;
```

```
for j=1:1:length(Max_Bndl_region)-5
window=string(Max_Bndl_start+start_point+j:Max_Bndl_start+start_point+j+
5);
numerical=isstrprop(window,'digit');

if   numerical(1)==0   &&   numerical(2)==0   &&   numerical(3)==0   &&
numerical(4)==1 && numerical(5)==0 && numerical(6)==0
    count=count+1;

window2=string(Max_Bndl_start+start_point+j+3:Max_Bndl_start+start_point
+j+3);
    Max_Bndl_matrix(count)=str2num(window2);
end

if   numerical(1)==0   &&   numerical(2)==0   &&   numerical(3)==1   &&
numerical(4)==1 && numerical(5)==0 && numerical(6)==0
    count=count+1;

window2=string(Max_Bndl_start+start_point+j+2:Max_Bndl_start+start_point
+j+3);
    Max_Bndl_matrix(count)=str2num(window2);
end

if   numerical(1)==0   &&   numerical(2)==0   &&   numerical(3)==1   &&
numerical(4)==1 && numerical(5)==1 && numerical(6)==0
    count=count+1;

window2=string(Max_Bndl_start+start_point+j+2:Max_Bndl_start+start_point
+j+4);
    Max_Bndl_matrix(count)=str2num(window2);
end

if   numerical(1)==0   &&   numerical(2)==1   &&   numerical(3)==1   &&
numerical(4)==1 && numerical(5)==1 && numerical(6)==0
    count=count+1;

window2=string(Max_Bndl_start+start_point+j+1:Max_Bndl_start+start_point
+j+4);
    Max_Bndl_matrix(count)=str2num(window2);
end

end

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Organize matrix into actual core geometry %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

Max_Bndl_geometry=zeros(24,24);

if length(Max_Bndl_matrix)==480
%Row A
for i=1:1:10
    Max_Bndl_geometry(1,i+7)=Max_Bndl_matrix(i);
end
```

```
%Row B
for i=11:1:24
    Max_Bndl_geometry(2,i-5)=Max_Bndl_matrix(i);
end

%Row C
for i=25:1:40
    Max_Bndl_geometry(3,i-20)=Max_Bndl_matrix(i);
end

%Row D
for i=41:1:58
    Max_Bndl_geometry(4,i-37)=Max_Bndl_matrix(i);
end

%Row E
for i=59:1:78
    Max_Bndl_geometry(5,i-56)=Max_Bndl_matrix(i);
end

%Row F
for i=79:1:100
    Max_Bndl_geometry(6,i-77)=Max_Bndl_matrix(i);
end

%Row G
for i=101:1:122
    Max_Bndl_geometry(7,i-99)=Max_Bndl_matrix(i);
end

%Row H
for i=123:1:144
    Max_Bndl_geometry(8,i-121)=Max_Bndl_matrix(i);
end

%Row J
for i=145:1:168
    Max_Bndl_geometry(9,i-144)=Max_Bndl_matrix(i);
end

%Row K
for i=169:1:192
    Max_Bndl_geometry(10,i-168)=Max_Bndl_matrix(i);
end

%Row L
for i=193:1:216
    Max_Bndl_geometry(11,i-192)=Max_Bndl_matrix(i);
end

%Row M
for i=217:1:240
    Max_Bndl_geometry(12,i-216)=Max_Bndl_matrix(i);
end
```

```
%Row N
for i=241:1:264
    Max_Bndl_geometry(13,i-240)=Max_Bndl_matrix(i);
end

%Row O
for i=265:1:288
    Max_Bndl_geometry(14,i-264)=Max_Bndl_matrix(i);
end

%Row P
for i=289:1:312
    Max_Bndl_geometry(15,i-288)=Max_Bndl_matrix(i);
end

%Row Q
for i=313:1:336
    Max_Bndl_geometry(16,i-312)=Max_Bndl_matrix(i);
end

%Row R
for i=337:1:358
    Max_Bndl_geometry(17,i-335)=Max_Bndl_matrix(i);
end

%Row S
for i=359:1:380
    Max_Bndl_geometry(18,i-357)=Max_Bndl_matrix(i);
end

%Row T
for i=381:1:402
    Max_Bndl_geometry(19,i-379)=Max_Bndl_matrix(i);
end

%Row U
for i=403:1:422
    Max_Bndl_geometry(20,i-400)=Max_Bndl_matrix(i);
end

%Row V
for i=423:1:440
    Max_Bndl_geometry(21,i-419)=Max_Bndl_matrix(i);
end

%Row W
for i=441:1:456
    Max_Bndl_geometry(22,i-436)=Max_Bndl_matrix(i);
end

%Row X
for i=457:1:470
    Max_Bndl_geometry(23,i-451)=Max_Bndl_matrix(i);
end
```

```
%Row Y
for i=471:1:480
    Max_Bndl_geometry(24,i-463)=Max_Bndl_matrix(i);
end

else
    disp('FATAL ERROR IN MAX_BNDL: NUMBER OF MATRIX ELEMENTS DOES NOT
EQUAL 480')
    disp('# of matrix elements is:')
    disp(length(Max_Bndl_matrix))
    pause
end
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%
%%%%%%%%%%%%%%%%%%%%%%%        K-INCREASE       ON       REFUEL       AUTO
%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%


% Identify locations where string says certain KEY words
K_Increase_start=strfind(string,'K INCREASE ON REFUELLING');
             %Location tag for Channel Powers

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Setup Matrix of K-change Values %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

K_Increase_region=string(K_Increase_start+300:K_Increase_start+6250);
%% GOOD GUESS
start_point=300;
end_point=6250;

count=0;

for j=1:1:length(Ch_Power_region)-5
window=string(K_Increase_start+start_point+j:K_Increase_start+start_poin
t+j+5);
numerical=isstrprop(window,'digit');

windowspecial=string(K_Increase_start+start_point+j:K_Increase_start+sta
rt_point+j+10);
numerical2=isstrprop(windowspecial,'digit');

window3=string(K_Increase_start+start_point+j:K_Increase_start+start_poi
nt+j+6);
numerical3=isstrprop(window3,'digit');

if   numerical2(1)==1   &&   numerical2(2)==0   &&   numerical2(3)==0   &&
numerical2(4)==0    &&    numerical2(5)==0    &&    numerical2(6)==0    &&
numerical2(7)==0    &&    numerical2(8)==0    &&    numerical2(9)==1    &&
numerical2(10)==1 && numerical2(11)==1
count=count+1;
K_Increase_matrix(count)=0;
```

```
elseif  numerical2(1)==1  &&  numerical2(2)==0  &&  numerical2(3)==0  &&
numerical2(4)==0    &&    numerical2(5)==0    &&    numerical2(6)==0    &&
numerical2(7)==0    &&    numerical2(8)==0    &&    numerical2(9)==0    &&
numerical2(10)==1 && numerical2(11)==1
 count=count+1;
 K_Increase_matrix(count)=0;

elseif  numerical2(1)==1  &&  numerical2(2)==0  &&  numerical2(3)==0  &&
numerical2(4)==0    &&    numerical2(5)==0    &&    numerical2(6)==0    &&
numerical2(7)==0    &&    numerical2(8)==0    &&    numerical2(9)==0    &&
numerical2(10)==0 && numerical2(11)==1
 count=count+1;
 K_Increase_matrix(count)=0;

elseif  numerical3(1)==1  &&  numerical3(2)==0  &&  numerical3(3)==0  &&
numerical3(4)==0    &&    numerical3(5)==0    &&    numerical3(6)==0    &&
numerical3(7)==0 && window3(7)==':'
  count=count+1;
 K_Increase_matrix(count)=0;

end

if   numerical(1)==0   &&   numerical(2)==0   &&   numerical(3)==0   &&
numerical(4)==1 && numerical(5)==0 && numerical(6)==0

    if window(3)=='-'
       count=count+1;

window2=string(K_Increase_start+start_point+j+2:K_Increase_start+start_p
oint+j+3);
       K_Increase_matrix(count)=str2num(window2);

    else
    count=count+1;

window2=string(K_Increase_start+start_point+j+3:K_Increase_start+start_p
oint+j+3);
    K_Increase_matrix(count)=str2num(window2);

    end
end

if   numerical(1)==0   &&   numerical(2)==0   &&   numerical(3)==1   &&
numerical(4)==1 && numerical(5)==0 && numerical(6)==0

    if window(2)=='-'
       count=count+1;

window2=string(K_Increase_start+start_point+j+1:K_Increase_start+start_p
oint+j+3);
       K_Increase_matrix(count)=str2num(window2);

    else
    count=count+1;
```

```
window2=string(K_Increase_start+start_point+j+2:K_Increase_start+start_p
oint+j+3);
    K_Increase_matrix(count)=str2num(window2);
    end
end

if   numerical(1)==0   &&   numerical(2)==0   &&   numerical(3)==1   &&
numerical(4)==1 && numerical(5)==1 && numerical(6)==0

    if window(2)=='-'
       count=count+1;

window2=string(K_Increase_start+start_point+j+1:K_Increase_start+start_p
oint+j+4);
       K_Increase_matrix(count)=str2num(window2);

    else
    count=count+1;

window2=string(K_Increase_start+start_point+j+2:K_Increase_start+start_p
oint+j+4);
    K_Increase_matrix(count)=str2num(window2);
    end

end

if   numerical(1)==0   &&   numerical(2)==1   &&   numerical(3)==1   &&
numerical(4)==1 && numerical(5)==1 && numerical(6)==0

    if window(1)=='-'
       count=count+1;

window2=string(K_Increase_start+start_point+j:K_Increase_start+start_poi
nt+j+4);
       K_Increase_matrix(count)=str2num(window2);

    else
    count=count+1;

window2=string(K_Increase_start+start_point+j+1:K_Increase_start+start_p
oint+j+4);
    K_Increase_matrix(count)=str2num(window2);
    end
end

end

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Organize matrix into actual core geometry %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%


K_Increase_geometry=zeros(24,24);
```

```
if length(K_Increase_matrix)==480
%Row A
for i=1:1:10
    K_Increase_geometry(1,i+7)=K_Increase_matrix(i);
end

%Row B
for i=11:1:24
    K_Increase_geometry(2,i-5)=K_Increase_matrix(i);
end

%Row C
for i=25:1:40
    K_Increase_geometry(3,i-20)=K_Increase_matrix(i);
end

%Row D
for i=41:1:58
    K_Increase_geometry(4,i-37)=K_Increase_matrix(i);
end

%Row E
for i=59:1:78
    K_Increase_geometry(5,i-56)=K_Increase_matrix(i);
end

%Row F
for i=79:1:100
    K_Increase_geometry(6,i-77)=K_Increase_matrix(i);
end

%Row G
for i=101:1:122
    K_Increase_geometry(7,i-99)=K_Increase_matrix(i);
end

%Row H
for i=123:1:144
    K_Increase_geometry(8,i-121)=K_Increase_matrix(i);
end

%Row J
for i=145:1:168
    K_Increase_geometry(9,i-144)=K_Increase_matrix(i);
end

%Row K
for i=169:1:192
    K_Increase_geometry(10,i-168)=K_Increase_matrix(i);
end

%Row L
for i=193:1:216
    K_Increase_geometry(11,i-192)=K_Increase_matrix(i);
```

```
end

%Row M
for i=217:1:240
    K_Increase_geometry(12,i-216)=K_Increase_matrix(i);
end

%Row N
for i=241:1:264
    K_Increase_geometry(13,i-240)=K_Increase_matrix(i);
end

%Row O
for i=265:1:288
    K_Increase_geometry(14,i-264)=K_Increase_matrix(i);
end

%Row P
for i=289:1:312
    K_Increase_geometry(15,i-288)=K_Increase_matrix(i);
end

%Row Q
for i=313:1:336
    K_Increase_geometry(16,i-312)=K_Increase_matrix(i);
end

%Row R
for i=337:1:358
    K_Increase_geometry(17,i-335)=K_Increase_matrix(i);
end

%Row S
for i=359:1:380
    K_Increase_geometry(18,i-357)=K_Increase_matrix(i);
end

%Row T
for i=381:1:402
    K_Increase_geometry(19,i-379)=K_Increase_matrix(i);
end

%Row U
for i=403:1:422
    K_Increase_geometry(20,i-400)=K_Increase_matrix(i);
end

%Row V
for i=423:1:440
    K_Increase_geometry(21,i-419)=K_Increase_matrix(i);
end

%Row W
for i=441:1:456
    K_Increase_geometry(22,i-436)=K_Increase_matrix(i);
```

```
end

%Row X
for i=457:1:470
    K_Increase_geometry(23,i-451)=K_Increase_matrix(i);
end

%Row Y
for i=471:1:480
    K_Increase_geometry(24,i-463)=K_Increase_matrix(i);
end

else
    disp('FATAL ERROR IN K_INCREASE: NUMBER OF MATRIX ELEMENTS DOES NOT
EQUAL 480')
    disp('# of matrix elements is:')
    disp(length(K_Increase_matrix))
    pause
end
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%           AXIAL           TILT           AUTO
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%


% Identify locations where string says certain KEY words
Axial_Tilt_start=strfind(string,'AXIAL TILT');             %Location tag
for Channel Powers

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Setup Matrix of Axial  Tilt Values %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

Axial_Tilt_region=string(Axial_Tilt_start+300:Axial_Tilt_start+6250);
%% GOOD GUESS
start_point=300;
end_point=6250;

count=0;

for j=1:1:length(Ch_Power_region)-5
window=string(Axial_Tilt_start+start_point+j:Axial_Tilt_start+start_poin
t+j+5);
numerical=isstrprop(window,'digit');

windowspecial=string(Axial_Tilt_start+start_point+j:Axial_Tilt_start+sta
rt_point+j+10);
numerical2=isstrprop(windowspecial,'digit');

window3=string(Axial_Tilt_start+start_point+j:Axial_Tilt_start+start_poi
nt+j+6);
numerical3=isstrprop(window3,'digit');
```

```
if   numerical2(1)==1   &&   numerical2(2)==0   &&   numerical2(3)==0   &&
numerical2(4)==0    &&    numerical2(5)==0    &&    numerical2(6)==0    &&
numerical2(7)==0    &&    numerical2(8)==0    &&    numerical2(9)==1    &&
numerical2(10)==1 && numerical2(11)==1
count=count+1;
Axial_Tilt_matrix(count)=0;

elseif  numerical2(1)==1  &&  numerical2(2)==0  &&  numerical2(3)==0  &&
numerical2(4)==0    &&    numerical2(5)==0    &&    numerical2(6)==0    &&
numerical2(7)==0    &&    numerical2(8)==0    &&    numerical2(9)==0    &&
numerical2(10)==1 && numerical2(11)==1
 count=count+1;
 Axial_Tilt_matrix(count)=0;

elseif  numerical2(1)==1  &&  numerical2(2)==0  &&  numerical2(3)==0  &&
numerical2(4)==0    &&    numerical2(5)==0    &&    numerical2(6)==0    &&
numerical2(7)==0    &&    numerical2(8)==0    &&    numerical2(9)==0    &&
numerical2(10)==0 && numerical2(11)==1
 count=count+1;
 Axial_Tilt_matrix(count)=0;

elseif  numerical3(1)==1  &&  numerical3(2)==0  &&  numerical3(3)==0  &&
numerical3(4)==0    &&    numerical3(5)==0    &&    numerical3(6)==0    &&
numerical3(7)==0 && window3(7)==':'
  count=count+1;
 Axial_Tilt_matrix(count)=0;

end

if   numerical(1)==0   &&   numerical(2)==0   &&   numerical(3)==0   &&
numerical(4)==1 && numerical(5)==0 && numerical(6)==0

    if window(3)=='-'
       count=count+1;

window2=string(Axial_Tilt_start+start_point+j+2:Axial_Tilt_start+start_p
oint+j+3);
       Axial_Tilt_matrix(count)=str2num(window2);

    else
    count=count+1;

window2=string(Axial_Tilt_start+start_point+j+3:Axial_Tilt_start+start_p
oint+j+3);
    Axial_Tilt_matrix(count)=str2num(window2);

    end
end

if   numerical(1)==0   &&   numerical(2)==0   &&   numerical(3)==1   &&
numerical(4)==1 && numerical(5)==0 && numerical(6)==0

    if window(2)=='-'
       count=count+1;
```

```
window2=string(Axial_Tilt_start+start_point+j+1:Axial_Tilt_start+start_p
oint+j+3);
       Axial_Tilt_matrix(count)=str2num(window2);

    else
    count=count+1;

window2=string(Axial_Tilt_start+start_point+j+2:Axial_Tilt_start+start_p
oint+j+3);
    Axial_Tilt_matrix(count)=str2num(window2);
    end
end

if   numerical(1)==0   &&   numerical(2)==0   &&   numerical(3)==1   &&
numerical(4)==1 && numerical(5)==1 && numerical(6)==0

    if window(2)=='-'
       count=count+1;

window2=string(Axial_Tilt_start+start_point+j+1:Axial_Tilt_start+start_p
oint+j+4);
       Axial_Tilt_matrix(count)=str2num(window2);

    else
    count=count+1;

window2=string(Axial_Tilt_start+start_point+j+2:Axial_Tilt_start+start_p
oint+j+4);
    Axial_Tilt_matrix(count)=str2num(window2);
    end

end

if   numerical(1)==0   &&   numerical(2)==1   &&   numerical(3)==1   &&
numerical(4)==1 && numerical(5)==1 && numerical(6)==0

    if window(1)=='-'
       count=count+1;

window2=string(Axial_Tilt_start+start_point+j:Axial_Tilt_start+start_poi
nt+j+4);
       Axial_Tilt_matrix(count)=str2num(window2);

    else
    count=count+1;

window2=string(Axial_Tilt_start+start_point+j+1:Axial_Tilt_start+start_p
oint+j+4);
    Axial_Tilt_matrix(count)=str2num(window2);
    end
end

end
```

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Organize matrix into actual core geometry %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%


Axial_Tilt_geometry=zeros(24,24);

if length(Axial_Tilt_matrix)==480
%Row A
for i=1:1:10
    Axial_Tilt_geometry(1,i+7)=Axial_Tilt_matrix(i);
end

%Row B
for i=11:1:24
    Axial_Tilt_geometry(2,i-5)=Axial_Tilt_matrix(i);
end

%Row C
for i=25:1:40
    Axial_Tilt_geometry(3,i-20)=Axial_Tilt_matrix(i);
end

%Row D
for i=41:1:58
    Axial_Tilt_geometry(4,i-37)=Axial_Tilt_matrix(i);
end

%Row E
for i=59:1:78
    Axial_Tilt_geometry(5,i-56)=Axial_Tilt_matrix(i);
end

%Row F
for i=79:1:100
    Axial_Tilt_geometry(6,i-77)=Axial_Tilt_matrix(i);
end

%Row G
for i=101:1:122
    Axial_Tilt_geometry(7,i-99)=Axial_Tilt_matrix(i);
end

%Row H
for i=123:1:144
    Axial_Tilt_geometry(8,i-121)=Axial_Tilt_matrix(i);
end

%Row J
for i=145:1:168
    Axial_Tilt_geometry(9,i-144)=Axial_Tilt_matrix(i);
end

%Row K
```

```
for i=169:1:192
    Axial_Tilt_geometry(10,i-168)=Axial_Tilt_matrix(i);
end

%Row L
for i=193:1:216
    Axial_Tilt_geometry(11,i-192)=Axial_Tilt_matrix(i);
end

%Row M
for i=217:1:240
    Axial_Tilt_geometry(12,i-216)=Axial_Tilt_matrix(i);
end

%Row N
for i=241:1:264
    Axial_Tilt_geometry(13,i-240)=Axial_Tilt_matrix(i);
end

%Row O
for i=265:1:288
    Axial_Tilt_geometry(14,i-264)=Axial_Tilt_matrix(i);
end

%Row P
for i=289:1:312
    Axial_Tilt_geometry(15,i-288)=Axial_Tilt_matrix(i);
end

%Row Q
for i=313:1:336
    Axial_Tilt_geometry(16,i-312)=Axial_Tilt_matrix(i);
end

%Row R
for i=337:1:358
    Axial_Tilt_geometry(17,i-335)=Axial_Tilt_matrix(i);
end

%Row S
for i=359:1:380
    Axial_Tilt_geometry(18,i-357)=Axial_Tilt_matrix(i);
end

%Row T
for i=381:1:402
    Axial_Tilt_geometry(19,i-379)=Axial_Tilt_matrix(i);
end

%Row U
for i=403:1:422
    Axial_Tilt_geometry(20,i-400)=Axial_Tilt_matrix(i);
end

%Row V
```

```
for i=423:1:440
    Axial_Tilt_geometry(21,i-419)=Axial_Tilt_matrix(i);
end

%Row W
for i=441:1:456
    Axial_Tilt_geometry(22,i-436)=Axial_Tilt_matrix(i);
end

%Row X
for i=457:1:470
    Axial_Tilt_geometry(23,i-451)=Axial_Tilt_matrix(i);
end

%Row Y
for i=471:1:480
    Axial_Tilt_geometry(24,i-463)=Axial_Tilt_matrix(i);
end

else
    disp('FATAL ERROR IN AXIAL_TILT: NUMBER OF MATRIX ELEMENTS DOES NOT
EQUAL 480')
    disp('# of matrix elements is:')
    disp(length(Axial_Tilt_matrix))
    pause
end

% assign pre-designated matrix names with corresponding core snap data
Burnup=Ave_Burnup_geometry;
RBurnup=Burnup_TA_geometry;
Chpower=Ch_Power_geometry;
MBdpower=Max_Bndl_geometry;
Kchange=K_Increase_geometry/1000;
ATilt=Axial_Tilt_geometry/10;

% find/calculate important core snap informations (FPD, e_reactivity,
% k-eff, etc.)
find_energy_clock=strfind(string,'ENERGY      CLOCK      FOR      PRESENT
SIMULATION=');
energy_clock=str2num(string(find_energy_clock+37:find_energy_clock+47));
FPD=sum(energy_clock)/15906*0.25; % find what FPD the run is at

find_e_reactivity=strfind(string,'EXCESS REACTIVITY= 1.0-1/KEFF=');
e_reactivity=str2num(string(find_e_reactivity+34:find_e_reactivity+41));
k_eff=1/(1-(e_reactivity/1000));

find_r_decline=strfind(string,'INSTANTANEOUS CORE REACTIVITY DECAY RATE
=');
r_decline=str2num(string(find_r_decline+43:find_r_decline+62));

find_m_chp=strfind(string,'MAXIMUM CHANNEL POWER IS');
m_chp=str2num(string(find_m_chp+28:find_m_chp+35))*1000;
m_chp_loc=string(find_m_chp+43:find_m_chp+47);

find_m_bdp=strfind(string,'MAXIMUM BUNDLE  POWER IS');
```

```
m_bdp=str2num(string(find_m_bdp+25:find_m_bdp+35));
m_bdp_loc=string(find_m_bdp+43:find_m_bdp+59);

zcr_massloc=strfind(string,'NEW ZONE CONTROLLER FILLS ARE');
correct_zcr_loc=max(max(zcr_massloc));

coeffaaaa1=0;
coeffaaaa2=0;
for n=1:14  % find zone controller fills
if n>=6
coeffaaaa1=1;
end
if n>=11
coeffaaaa2=1;
end
zcrstartpoint=(correct_zcr_loc)+42+17*(n-1)+3*coeffaaaa1+5*coeffaaaa2;
zcrendpoint=zcrstartpoint+6;
zcr(1,n)=str2num(string(zcrstartpoint:zcrendpoint));
end
avgzcr=mean(zcr); % find average zone controller fill

maxChpower=max(max(Chpower)); % maximum channel power
avgChpower=sum(sum(Chpower))/(24*24-24*4); % average channel power
FormFactor=maxChpower/avgChpower; % calculate form factor
maxMBdpower=max(max(MBdpower)); % maximum max bundle power (ironically,
this is not redundant)

for s=1:24;
for t=1:24;
if Chpower(s,t)~=0;
CPPF(s,t)=Chpower(s,t)/Reference_Core(s,t);  % current channel power
ratio to reference T.A channel power
end
end
end

Burnupzz=Burnup;
RBurnupzz=RBurnup;
Chpowerzz=Chpower;
MBdpowerzz=MBdpower;
Kchangezz=Kchange;
ATiltzz=ATilt;
CPPFzz=CPPF;

for a=1:24
for b=1:24
if Chpowerzz(a,b)==0;
Chpowerzz(a,b)=NaN;
Burnupzz(a,b)=NaN;
RBurnupzz(a,b)=NaN;
MBdpowerzz(a,b)=NaN;
Kchangezz(a,b)=NaN;
ATiltzz(a,b)=NaN;
CPPFzz(a,b)=NaN;
end
end
```

```
end

MaxChPplot(1,loopcrit+1)=m_chp;  %   assign important plotdata, at each
FPDs' run
MaxMBdPplot(1,loopcrit+1)=m_bdp;
FormFactorplot(1,loopcrit+1)=FormFactor;
Keffplot(1,loopcrit+1)=k_eff;
RDeclineplot(1,loopcrit+1)=r_decline;
e_reactplot(1,loopcrit+1)=e_reactivity;
avgzcrplot(1,loopcrit+1)=avgzcr;
zcr1plot(1,loopcrit+1)=zcr(1,1);
zcr2plot(1,loopcrit+1)=zcr(1,2);
zcr3plot(1,loopcrit+1)=zcr(1,3);
zcr4plot(1,loopcrit+1)=zcr(1,4);
zcr5plot(1,loopcrit+1)=zcr(1,5);
zcr6plot(1,loopcrit+1)=zcr(1,6);
zcr7plot(1,loopcrit+1)=zcr(1,7);
zcr8plot(1,loopcrit+1)=zcr(1,8);
zcr9plot(1,loopcrit+1)=zcr(1,9);
zcr10plot(1,loopcrit+1)=zcr(1,10);
zcr11plot(1,loopcrit+1)=zcr(1,11);
zcr12plot(1,loopcrit+1)=zcr(1,12);
zcr13plot(1,loopcrit+1)=zcr(1,13);
zcr14plot(1,loopcrit+1)=zcr(1,14);


for a=1:24 % incorporate snapshot data into plot data, at each FPDs' run
for b=1:24

ChPowerplot(a,b,loopcrit+1)=Chpower(a,b);
MBPowerplot(a,b,loopcrit+1)=MBdpower(a,b);
Kchangeplot(a,b,loopcrit+1)=Kchange(a,b);
Burnupplot(a,b,loopcrit+1)=Burnup(a,b);
RBurnupplot(a,b,loopcrit+1)=RBurnup(a,b);
ATiltplot(a,b,loopcrit+1)=ATilt(a,b);
CPPFplot(a,b,loopcrit+1)=CPPF(a,b);

end
end



%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%

fclose('all');
display('Data Import Complete')
```

```
disp('.') % spacing till next comment

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%          DATA          PROCESS          BEGIN
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%

display(strcat(['Processing  Core  Snapshot  Data  for:'],num2str(FPD),'
FPD'));
disp('.') % spacing till next comment

% expand total matrix size for bundle and channel power distribution,
% so neighboring channels for periphery channels can also be scanned
% (the neighboring channels of periphery channels will be equal to zero
Chnpower=zeros(34:34);
for p=1:24
for q=1:24
Chnpower(p+5,q+5)=Chpower(p,q);
end
end
MBdlpower=zeros(34:34);
for p=1:24
for q=1:24
MBdlpower(p+5,q+5)=MBdpower(p,q);
end
end

ChLim=6800; % maximum channel power
IdealCh=5500; % "desirable" channel power
BdLim=800; % maximum bundle power
IdealBd=650; % "desirable" bundle power
RDecay=abs(r_decline); % Set RDecay for next run
RReq=RDecay/4;  % k_change insertion requirement for every 0.25 FPD is
calculated

% if e_reactivity > 1.0 % relaxation for k_change if excess reactivity
% RReq=RReq*0.5;
% end

wChP=8.0; % weight of importance of channel power compliance score
wBdP=8.0; % weight of importance of bundle power compliance score
wKchange=5.0; % weight of importance of k-change score
wFlatnRPPF=4.0; % weight of importance of core flattening score
wATilt=3.0; % weight of importance of Axial Tilt score
wChSep=3.0;  %  weight  of  importance  of  channel  separation  from  last
fuelled channel
wRBurnup=5.0; % weight of importance of Relative Burnup score
wCPPF=3.0; % weight of importance of CPPF score
wBurnup=0.5; % weight of importance of Burnup score
```

```
wAvoidMChBdP=3.0; % weight of importance of avoiding the max bd/ch power
channel

Coordinate=ones(24:24); % location identifier for each channels

scoreChP=zeros(24:24); % channel power score
scoreBdP=zeros(24:24); % bundle power score
scoreK=zeros(24:24); % initial distribution of points for k-change
scoreRBurn=zeros(24:24); % initial distribution of points for Burnup
(use relative burnup)
scoreFlatnRPPF=zeros(24:24); % initial distribution of points for
achieving flat core
scoreATilt=zeros(24:24); % initial distribution of points for balancing
axial tilt
scoreCPPF=zeros(24:24); % initial distribution of points for CPPF

subscoreChSep1=zeros(24:24); % initial distribution of points for sub-
score of ChSep score
subscoreChSep2=zeros(24:24);
subscoreChSep3=zeros(24:24);
subscoreChSep4=zeros(24:24);
subscoreChSep5=zeros(24:24);
subscoreChSep6=zeros(24:24);
subscoreChSep7=zeros(24:24);
subscoreChSep8=zeros(24:24);
avoidChannel=zeros(24:24); % initial distribution of points for sub-
score of avoidMBdChPower
avoidBundle=zeros(24:24);

for y=1:24 % Lattice Rows
for x=1:24 % Lattice Columns

%------------------------locate max power Chn / Bdl---------------------
--%
if Chpower(y,x)==maxChpower;
yMaxCh=y;
xMaxCh=x;
end
if MBdpower(y,x)==maxMBdpower;
yMaxBd=y;
xMaxBd=x;
end
%------------------------locate max power Chn / Bdl---------------------
--%

%---------------------------score core flattening----------------------
--%

TL=zeros(12:8);
for e=1:12
for f=1:8
TL(e,f)=Chpower(e,f);
end
end
TopLeft=sum(sum(TL))/1000;
```

```
regPower(1,1)=TopLeft;

T=zeros(8:8);
for e=1:8
for f=1:8
T(e,f)=Chpower(e,f+8);
end
end
Top=sum(sum(T))/1000;
regPower(1,2)=Top;

TR=zeros(12:8);
for e=1:12
for f=1:8
TR(e,f)=Chpower(e,f+16);
end
end
TopRight=sum(sum(TR))/1000;
regPower(1,3)=TopRight;

M=zeros(8:8);
for e=1:8
for f=1:8
M(e,f)=Chpower(e+8,f+8);
end
end
Middle=sum(sum(M))/1000;
regPower(1,4)=Middle;

BL=zeros(12:8);
for e=1:12
for f=1:8
BL(e,f)=Chpower(e+12,f);
end
end
BotLeft=sum(sum(BL))/1000;
regPower(1,5)=BotLeft;

B=zeros(8:8);
for e=1:8
for f=1:8
B(e,f)=Chpower(e+16,f+8);
end
end
Bot=sum(sum(B))/1000;
regPower(1,6)=Bot;

BR=zeros(12:8);
for e=1:12
for f=1:8
BR(e,f)=Chpower(e+12,f+16);
end
end
BotRight=sum(sum(BR))/1000;
regPower(1,7)=BotRight;
```

```
% core flattening score based on regional power over total power
regScorez1(1,1)=(max(regPower)-regPower(1,1))/sum(regPower);
regScorez1(1,2)=(max(regPower)-regPower(1,2))/sum(regPower);
regScorez1(1,3)=(max(regPower)-regPower(1,3))/sum(regPower);
regScorez1(1,4)=(max(regPower)-regPower(1,4))/sum(regPower);
regScorez1(1,5)=(max(regPower)-regPower(1,5))/sum(regPower);
regScorez1(1,6)=(max(regPower)-regPower(1,6))/sum(regPower);
regScorez1(1,7)=(max(regPower)-regPower(1,7))/sum(regPower);

coeffzz=10/max(regScorez1);
regScorez1=regScorez1*coeffzz;

% core flattening score based on regional power over time-average
regional power
regScorez2(1,1)=(1-(TopLeft/397.747));
regScorez2(1,2)=(1-(Top/371.309));
regScorez2(1,3)=(1-(TopRight/395.437));
regScorez2(1,4)=(1-(Middle/394.322));
regScorez2(1,5)=(1-(BotLeft/373.114));
regScorez2(1,6)=(1-(Bot/348.436));
regScorez2(1,7)=(1-(BotRight/370.642));

if min(regScorez2)<0 % ensures no negative score
regScorez2=regScorez2-min(regScorez2);
end

coeffzz2=10/max(regScorez2);
regScorez2=regScorez2*coeffzz2;

% the two core flattening scores are added together
regScorez=regScorez1+1.5*regScorez2; % regional power proportional to
time
% average is more important, thus multiplied by 1.5

coeffregscore=10/max(regScorez); % normalize to 10
regScorez=regScorez*coeffregscore;

%if regScorez(1,4) < max(regPower)  %%%%%% See if this is good or not.
%regScorez(1,4)=11;
%coeff=10/max(regScorez);
%regScorez=regScorez*coeff;
%end

for e=1:12
for f=1:8
scoreFlatnRPPF(e,f)=regScorez(1,1);
end
end
for e=1:8
for f=9:16
scoreFlatnRPPF(e,f)=regScorez(1,2);
end
end
for e=1:12
```

```
for f=17:24
scoreFlatnRPPF(e,f)=regScorez(1,3);
end
end
for e=9:16
for f=9:16
scoreFlatnRPPF(e,f)=regScorez(1,4);
end
end
for e=13:24
for f=1:8
scoreFlatnRPPF(e,f)=regScorez(1,5);
end
end
for e=17:24
for f=9:16
scoreFlatnRPPF(e,f)=regScorez(1,6);
end
end
for e=13:24
for f=17:24
scoreFlatnRPPF(e,f)=regScorez(1,7);
end
end

for a=1:24
for b=1:24

if Chpower(a,b)==0;
scoreFlatnRPPF(a,b)=-100;
end

end
end

%---------------------------score core flattening-----------------------
--%

end
end

%--------------------------------score CPPF-----------------------------
--%
for y=1:24
for x=1:24

if CPPF(y,x)==0;
scoreCPPF(y,x)=0;
elseif abs(1-CPPF(y,x))>=0.2
scoreCPPF(y,x)=4*10*(1-CPPF(y,x)); % if CPPF is less than 0.8, greater
score is given
elseif abs(1-CPPF(y,x))>=0.15
scoreCPPF(y,x)=3*10*(1-CPPF(y,x)); % if CPPF is less than 0.8, greater
score is given
elseif abs(1-CPPF(y,x))>=0.1
```

```
scoreCPPF(y,x)=2*10*(1-CPPF(y,x)); % if CPPF is less than 0.8, greater
score is given
elseif abs(1-CPPF(y,x))<0.1
scoreCPPF(y,x)=1*10*(1-CPPF(y,x)); % if CPPF is less than 0.8, greater
score is given
end

end
end

for a=1:24 % ensure there is no negative score
for b=1:24
if Chpower(a,b)~=0 && min(min(scoreCPPF))<0;
scoreCPPF(a,b)=scoreCPPF(a,b)-min(min(scoreCPPF));
end
end
end

CPPFcoeff=10/max(max(scoreCPPF));
for a=1:24 % ensure there is no negative score
for b=1:24
if Chpower(a,b)~=0;
scoreCPPF(a,b)=scoreCPPF(a,b)*CPPFcoeff;
elseif Chpower(a,b)==0;
scoreCPPF(a,b)=-100;
end
end
end

%---------------------------------score CPPF-----------------------------
--%

%---------------------------score Chn / Bdl power-----------------------
--%
for y=1:24
for x=1:24

c=1;
for a=-4:4
for b=-4:4

if Chnpower(y+5+a,x+5+b)<=IdealCh;
ChSc(1,c)=(1/(abs(a)+1))*(1/(abs(b)+1))*0.62;
elseif Chnpower(y+5+a,x+5+b)>IdealCh;
ChSc(1,c)=(1/(abs(a)+1))*(1/(abs(b)+1))*((ChLim-
Chnpower(y+5+a,x+5+b))/(ChLim-IdealCh));
end
if MBdlpower(y+5+a,x+5+b)<=IdealBd;
BdSc(1,c)=(1/(abs(a)+1))*(1/(abs(b)+1))*0.70;
elseif MBdlpower(y+5+a,x+5+b)>IdealBd;
BdSc(1,c)=(1/(abs(a)+1))*(1/(abs(b)+1))*((BdLim-
MBdlpower(y+5+a,x+5+b))/(BdLim-IdealBd));
end

c=c+1;
```

```
end
end

if Chnpower(y+5,x+5)==0; % for lattice pitches with no channel
scoreChP(y,x)=-100;
else
scoreChP(y,x)=sum(ChSc);
end
if MBdlpower(y+5,x+5)==0;
scoreBdP(y,x)=-100;
else
scoreBdP(y,x)=sum(BdSc);
end

end
end

coeff11=10/max(max(scoreChP));
coeff12=10/max(max(scoreBdP));
for n=1:24 % normalize score to 10
for o=1:24
if scoreChP(n,o)~=-100
scoreChP(n,o)=coeff11*scoreChP(n,o);
end
if scoreBdP(n,o)~=-100
scoreBdP(n,o)=coeff12*scoreBdP(n,o);
end
end
end
%---------------------------score Chn / Bdl power-----------------------
--%

%-----------------------score avoid max Ch / Bd power-------------------
--%

for l=1:24
for m=1:24

% this section ensures fuelling does not occur within 5 channel
separation
% from the max power channel (in terms of ChP and BdP)

if Chpower(l,m)==0;
avoidChannel(l,m)=-50;
elseif abs(yMaxCh-l)<=5 && abs(xMaxCh-m)<=5 && Chpower(l,m)~=0;
avoidChannel(l,m)=0;
else
avoidChannel(l,m)=abs(yMaxCh-l)+abs(xMaxCh-m);
end

if Chpower(l,m)==0;
avoidBundle(l,m)=-50;
elseif abs(yMaxBd-l)<=5 && abs(xMaxBd-m)<=5 && Chpower(l,m)~=0;
avoidBundle(l,m)=0;
else
```

```
avoidBundle(l,m)=abs(yMaxBd-l)+abs(xMaxBd-m);
end

scoreAvoidMChBdP=avoidChannel+avoidBundle;

coeffavoid=10/max(max(scoreAvoidMChBdP));
for j=1:24
for k=1:24
if scoreAvoidMChBdP(j,k)~=-100;
scoreAvoidMChBdP(j,k)=scoreAvoidMChBdP(j,k)*coeffavoid;
end
end
end

end
end

%-----------------------score avoid max Ch / Bd power-------------------
--%

%-----------------------------score axial tilt--------------------------
--%

if yChosenrecord(1)~=0 && xChosenrecord(1)~=0;
if  tiltmap(yChosenrecord(1),xChosenrecord(1))==-1;  % if last fuelled
channel was negative direction
scoreATilt=minustilt; % all positive direction channels get 10 points
elseif tiltmap(yChosenrecord(1),xChosenrecord(1))==1; % if last fuelled
channel was positive direction
scoreATilt=plustilt;  % all negative direction channels get 10 points
end
end
%-----------------------------score axial tilt--------------------------
--%

%-------------------------score channel separation----------------------
--%
for l=1:24
for m=1:24

% this section ensures channel separation from 9 most recent channels
that
% were refuelled

if yChosenrecord(1)==0 && xChosenrecord(1)==0;
subscoreChSep1(l,m)=0;
elseif yChosenrecord(1)~=0 && xChosenrecord(1)~=0 && Chpower(l,m)==0;
subscoreChSep1(l,m)=-100;
elseif yChosenrecord(1)~=0 && xChosenrecord(1)~=0 && Chpower(l,m)~=0 &&
abs(yChosenrecord(1)-l)<=5 && abs(xChosenrecord(1)-m)<=5;
subscoreChSep1(l,m)=0;
elseif yChosenrecord(1)~=0 && xChosenrecord(1)~=0 && Chpower(l,m)~=0 &&
(abs(yChosenrecord(1)-l)>5 || abs(xChosenrecord(1)-m)>5);
subscoreChSep1(l,m)=abs(yChosenrecord(1)-l)+abs(xChosenrecord(1)-m);
end
```

```
if yChosenrecord(2)==0 && xChosenrecord(2)==0;
subscoreChSep2(l,m)=0;
elseif yChosenrecord(2)~=0 && xChosenrecord(2)~=0 && Chpower(l,m)==0;
subscoreChSep2(l,m)=0;
elseif yChosenrecord(2)~=0 && xChosenrecord(2)~=0 && Chpower(l,m)~=0 &&
abs(yChosenrecord(2)-l)<=5 && abs(xChosenrecord(2)-m)<=5;
subscoreChSep2(l,m)=0;
elseif yChosenrecord(2)~=0 && xChosenrecord(2)~=0 && Chpower(l,m)~=0 &&
(abs(yChosenrecord(2)-l)>5 || abs(xChosenrecord(2)-m)>5);
subscoreChSep2(l,m)=abs(yChosenrecord(2)-l)+abs(xChosenrecord(2)-m);
end

if yChosenrecord(3)==0 && xChosenrecord(3)==0;
subscoreChSep3(l,m)=0;
elseif yChosenrecord(3)~=0 && xChosenrecord(3)~=0 && Chpower(l,m)==0;
subscoreChSep3(l,m)=0;
elseif yChosenrecord(3)~=0 && xChosenrecord(3)~=0 && Chpower(l,m)~=0 &&
abs(yChosenrecord(3)-l)<=5 && abs(xChosenrecord(3)-m)<=5;
subscoreChSep3(l,m)=0;
elseif yChosenrecord(3)~=0 && xChosenrecord(3)~=0 && Chpower(l,m)~=0 &&
(abs(yChosenrecord(3)-l)>5 || abs(xChosenrecord(3)-m)>5);
subscoreChSep3(l,m)=abs(yChosenrecord(3)-l)+abs(xChosenrecord(3)-m);
end

if yChosenrecord(4)==0 && xChosenrecord(4)==0;
subscoreChSep4(l,m)=0;
elseif yChosenrecord(4)~=0 && xChosenrecord(4)~=0 && Chpower(l,m)==0;
subscoreChSep4(l,m)=0;
elseif yChosenrecord(4)~=0 && xChosenrecord(4)~=0 && Chpower(l,m)~=0 &&
abs(yChosenrecord(4)-l)<=5 && abs(xChosenrecord(4)-m)<=5;
subscoreChSep4(l,m)=0;
elseif yChosenrecord(4)~=0 && xChosenrecord(4)~=0 && Chpower(l,m)~=0 &&
(abs(yChosenrecord(4)-l)>5 || abs(xChosenrecord(4)-m)>5);
subscoreChSep4(l,m)=abs(yChosenrecord(4)-l)+abs(xChosenrecord(4)-m);
end

if yChosenrecord(5)==0 && xChosenrecord(5)==0;
subscoreChSep5(l,m)=0;
elseif yChosenrecord(5)~=0 && xChosenrecord(5)~=0 && Chpower(l,m)==0;
subscoreChSep5(l,m)=0;
elseif yChosenrecord(5)~=0 && xChosenrecord(5)~=0 && Chpower(l,m)~=0 &&
abs(yChosenrecord(5)-l)<=5 && abs(xChosenrecord(5)-m)<=5;
subscoreChSep5(l,m)=0;
elseif yChosenrecord(5)~=0 && xChosenrecord(5)~=0 && Chpower(l,m)~=0 &&
(abs(yChosenrecord(5)-l)>5 || abs(xChosenrecord(5)-m)>5);
subscoreChSep5(l,m)=abs(yChosenrecord(5)-l)+abs(xChosenrecord(5)-m);
end

if yChosenrecord(6)==0 && xChosenrecord(6)==0;
subscoreChSep6(l,m)=0;
elseif yChosenrecord(6)~=0 && xChosenrecord(6)~=0 && Chpower(l,m)==0;
subscoreChSep6(l,m)=0;
elseif yChosenrecord(6)~=0 && xChosenrecord(6)~=0 && Chpower(l,m)~=0 &&
abs(yChosenrecord(6)-l)<=5 && abs(xChosenrecord(6)-m)<=5;
subscoreChSep6(l,m)=0;
```

```
elseif yChosenrecord(6)~=0 && xChosenrecord(6)~=0 && Chpower(l,m)~=0 &&
(abs(yChosenrecord(6)-l)>5 || abs(xChosenrecord(6)-m)>5);
subscoreChSep6(l,m)=abs(yChosenrecord(6)-l)+abs(xChosenrecord(6)-m);
end

if yChosenrecord(7)==0 && xChosenrecord(7)==0;
subscoreChSep7(l,m)=0;
elseif yChosenrecord(7)~=0 && xChosenrecord(7)~=0 && Chpower(l,m)==0;
subscoreChSep7(l,m)=0;
elseif yChosenrecord(7)~=0 && xChosenrecord(7)~=0 && Chpower(l,m)~=0 &&
abs(yChosenrecord(7)-l)<=5 && abs(xChosenrecord(7)-m)<=5;
subscoreChSep7(l,m)=0;
elseif yChosenrecord(7)~=0 && xChosenrecord(7)~=0 && Chpower(l,m)~=0 &&
(abs(yChosenrecord(7)-l)>5 || abs(xChosenrecord(7)-m)>5);
subscoreChSep7(l,m)=abs(yChosenrecord(7)-l)+abs(xChosenrecord(7)-m);
end

if yChosenrecord(8)==0 && xChosenrecord(8)==0;
subscoreChSep8(l,m)=0;
elseif yChosenrecord(8)~=0 && xChosenrecord(8)~=0 && Chpower(l,m)==0;
subscoreChSep8(l,m)=0;
elseif yChosenrecord(8)~=0 && xChosenrecord(8)~=0 && Chpower(l,m)~=0 &&
abs(yChosenrecord(8)-l)<=5 && abs(xChosenrecord(8)-m)<=5;
subscoreChSep8(l,m)=0;
elseif yChosenrecord(8)~=0 && xChosenrecord(8)~=0 && Chpower(l,m)~=0 &&
(abs(yChosenrecord(8)-l)>5 || abs(xChosenrecord(8)-m)>5);
subscoreChSep8(l,m)=abs(yChosenrecord(8)-l)+abs(xChosenrecord(8)-m);
end

end
end

scoreChSep=subscoreChSep1+(0.95)*subscoreChSep2+(0.9)*subscoreChSep3+(0.
85)*subscoreChSep4+(0.8)*subscoreChSep5+(0.75)*subscoreChSep6+(0.7)*subs
coreChSep7+(0.75)*subscoreChSep8;

coeffchsep=10/max(max(scoreChSep));
for j=1:24
for k=1:24
if Chpower(j,k)~=0 && max(max(scoreChSep))>0;
scoreChSep(j,k)=scoreChSep(j,k)*coeffchsep;
end
end
end

%-------------------------score channel separation----------------------
--%

%-------------------------------score burnup---------------------------
--%
for y=1:24
for x=1:24

if RBurnup(y,x)==0;
scoreRBurn(y,x)=-100;
```

```
elseif RBurnup(y,x)<90 && RBurnup(y,x)>=70
scoreRBurn(y,x)=10*RBurnup(y,x)/100; % score based on how close to 100%
of time-average burnup
elseif RBurnup(y,x)>=90 && RBurnup(y,x)<100 % clause to prevent any
channels from reaching over 110% Rel-Burnup
scoreRBurn(y,x)=10;
elseif RBurnup(y,x)>=100 && RBurnup(y,x)<110 % clause to prevent any
channels from reaching over 110% Rel-Burnup
scoreRBurn(y,x)=12;
elseif RBurnup(y,x)>=110 % clause to prevent any channels from reaching
over 110% Rel-Burnup
scoreRBurn(y,x)=15;
end

end
end

RBurncoeff=10/max(max(scoreRBurn));
for j=1:24
for k=1:24
if scoreRBurn(j,k)~=-100;
scoreRBurn(j,k)=scoreRBurn(j,k)*RBurncoeff;
end
end
end

%-------------------------------score burnup-----------------------------
--%

%-------------------------------score k-change---------------------------
--%
for y=1:24
for x=1:24

if Kchange(y,x)==0;
scoreK(y,x)=-100;
elseif (Kchange(y,x)<(RReq*0.8) && RBurnup(y,x)<80) || RBurnup(y,x)<50
% channels providing way too small k-change is not even considered,
unless
% high relative burnup. Same with any channel with relative burnup < 50
scoreK(y,x)=0;
scoreRBurn(y,x)=0;
scoreFlatnRPPF(y,x)=0;
scoreChP(y,x)=0;
scoreBdP(y,x)=0;
scoreATilt(y,x)=0;
scoreChSep(y,x)=0;
scoreCPPF(y,x)=0;
scoreAvoidMChBdP(y,x)=0;
elseif (Kchange(y,x)<(RReq*0.8) && RBurnup(y,x)>=80) % even if low k-
change, if high burnup, channel may be fuelled
scoreK(y,x)=10*Kchange(y,x)/(RReq);
elseif Kchange(y,x)>=(RReq*0.8) && RBurnup(y,x)>=50 %&&
Kchange(y,x)<(RReq)
scoreK(y,x)=10*Kchange(y,x)/(RReq);
elseif Kchange(y,x)>=(RReq)
```

```
scoreK(y,x)=12;
elseif Kchange(y,x)>=1.1*(RReq)
scoreK(y,x)=15;
% elseif Kchange(y,x)>=(RReq) && RBurnup(y,x)>=80
% scoreK(y,x)=10'

end

if e_reactivity>0.5 && Kchange(y,x)>0;  % filter clause for k_change if
large excess reactivity; causes it to not fuel for that day.
scoreK(y,x)=0;
scoreRBurn(y,x)=0;
scoreFlatnRPPF(y,x)=0;
scoreChP(y,x)=0;
scoreBdP(y,x)=0;
scoreATilt(y,x)=0;
scoreChSep(y,x)=0;
scoreCPPF(y,x)=0;
scoreAvoidMChBdP(y,x)=0;

%elseif e_reactivity<0.25 && Kchange(y,x)>0 && max(max(Kchange))>=(RReq)
&& Kchange(y,x)<(RReq)   % filter clause for k_change if too low excess
reactivity; causes it to consider only high k-change
%scoreK(y,x)=0;
%scoreRBurn(y,x)=0;
%scoreFlatnRPPF(y,x)=0;
%scoreChP(y,x)=0;
%scoreBdP(y,x)=0;
%scoreATilt(y,x)=0;
%scoreChSep(y,x)=0;
%scoreCPPF(y,x)=0;
%scoreAvoidMChBdP(y,x)=0;

%elseif e_reactivity<0.25 && Kchange(y,x)>0 && max(max(Kchange))<(RReq)
&& Kchange(y,x)<0.70*max(max(Kchange))   % if low excess reactivity and
no channel has higher than 1/4 of required k-change, exclude only bottom
75% percentile of low k-change channels
%scoreK(y,x)=0;
%scoreRBurn(y,x)=0;
%scoreFlatnRPPF(y,x)=0;
%scoreChP(y,x)=0;
%scoreBdP(y,x)=0;
%scoreATilt(y,x)=0;
%scoreChSep(y,x)=0;
%scoreCPPF(y,x)=0;
%scoreAvoidMChBdP(y,x)=0;
end

end
end

kcoeffz=10/max(max(scoreK));
for j=1:24
for k=1:24
if scoreK(j,k)~=-100 && max(max(scoreK(j,k)))>0;
scoreK(j,k)=scoreK(j,k)*kcoeffz;
```

```
end
end
end

%------------------------------score k-change-----------------------------
--%

TotalScore=wChP*scoreChP+wBdP*scoreBdP+wKchange*scoreK+wRBurnup*scoreRBu
rn+wFlatnRPPF*scoreFlatnRPPF+wATilt*scoreATilt+wChSep*scoreChSep+wCPPF*s
coreCPPF+wAvoidMChBdP*scoreAvoidMChBdP;

for i=1:24
for j=1:24
if TotalScore(i,j)<0
TotalScore(i,j)=0;
end
end
end



if max(max(TotalScore))>0;
TS_coeff=10/max(max(TotalScore)); % normalize total score in scale of 1-
10
TotalScore=TS_coeff*TotalScore;
end

TotalScorez=TotalScore;

for a=1:24
for b=1:24
if TotalScorez(a,b)==0;
TotalScorez(a,b)=NaN;
end
end
end

TotalScoreFilter=TotalScore;
maxFTS=max(max(TotalScoreFilter));
for i=1:24
for j=1:24
if TotalScoreFilter(i,j)<maxFTS*0.90
TotalScoreFilter(i,j)=0;
end
end
end

for i=1:24
for j=1:24
if  TotalScore(i,j)==max(max(TotalScore))  &&  max(max(TotalScore))>0;  %
indicate chosen channel
yChos=i; % newly chosen channel column identifier
xChosen=j; % newly chosen channel row identifier
elseif max(max(TotalScore))<=0;
yChos=0;
xChosen=0;
```

```
end
end
end

if yChos==1;
yChosenz='A';
elseif yChos==2;
yChosenz='B';
elseif yChos==3;
yChosenz='C';
elseif yChos==4;
yChosenz='D';
elseif yChos==5;
yChosenz='E';
elseif yChos==6;
yChosenz='F';
elseif yChos==7;
yChosenz='G';
elseif yChos==8;
yChosenz='H';
elseif yChos==9;
yChosenz='J';
elseif yChos==10;
yChosenz='K';
elseif yChos==11;
yChosenz='L';
elseif yChos==12;
yChosenz='M';
elseif yChos==13;
yChosenz='N';
elseif yChos==14;
yChosenz='O';
elseif yChos==15;
yChosenz='P';
elseif yChos==16;
yChosenz='Q';
elseif yChos==17;
yChosenz='R';
elseif yChos==18;
yChosenz='S';
elseif yChos==19;
yChosenz='T';
elseif yChos==20;
yChosenz='U';
elseif yChos==21;
yChosenz='V';
elseif yChos==22;
yChosenz='W';
elseif yChos==23;
yChosenz='X';
elseif yChos==24;
yChosenz='Y';
end

for i=length(xChosenrecord):-1:2
yChosenrecord(i)=yChosenrecord(i-1);
```

```
xChosenrecord(i)=xChosenrecord(i-1);
end
yChosenrecord(1)=yChos;
xChosenrecord(1)=xChosen;

if yChos~=0 && xChosen~=0;
Fuelling_History(loopcrit+1)=cellstr(strcat(yChosenz,num2str(xChosen)));
% keep a string log of fuelled channels for each FPD
elseif yChos==0 && xChosen==0;
Fuelling_History(loopcrit+1)=cellstr('NO CHANNEL SELECTED');
end

for a=1:24
for b=1:24

TotalScoreplot(a,b,loopcrit+1)=TotalScorez(a,b);

end
end

MChploc_History(loopcrit+1)=cellstr(m_chp_loc);
MBdploc_History(loopcrit+1)=cellstr(m_bdp_loc);

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%          DATA          PROCESS          END
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%

display('Data Process Complete')
disp('.') % spacing till next comment
display('Displayng Results...')
disp('.') % spacing till next comment

display(strcat('K_Eff: ',num2str(k_eff)))
display(strcat('Form Factor: ',num2str(FormFactor)))
display(strcat('Excess Reactivity: ',num2str(e_reactivity),' milli-k'))
display(strcat('Rate  of  Reactivity  Decline:  ',num2str(r_decline),'
milli-k/FPD'))
display(strcat('Max  Channel  Power:  ',num2str(m_chp),'  kW','  at
',num2str(m_chp_loc)))
display(strcat('Max  Bundle  Power:  ',num2str(m_bdp),'  kW','  at
',num2str(m_bdp_loc)))
display(strcat('Average ZCR Fill: ',num2str(avgzcr)))

if yChos~=0 && xChosen~=0;
yChosen=strcat(yChosenz,num2str(xChosen),'          (',num2str(yChos),'-
',num2str(xChosen),')');
elseif yChos==0 && xChosen==0;
yChosen='NO CHANNEL SELECTED';
```

```
end

display(strcat('Chosen Channel for Fuelling:',yChosen))


%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%          NEXT          RUN          CREATE
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%

display('Creating New RFSP Input File...')
disp('.') % spacing till next comment

delete('rfsp_input')

fopen('C:\0_DAY_STARTPOINT_CDrive\excel_template\rfsp_input_template.txt
');
string_RFSP=fileread('C:\0_DAY_STARTPOINT_CDrive\excel_template\rfsp_inp
ut_template.txt');

% DAF
sumfpd=num2str(FPD);
daf_file_num=strcat(sumfpd, 'FPD_DAF');
find_daf_file_num_location=strfind(string_RFSP,'##########');
window=string_RFSP(find_daf_file_num_location+0:find_daf_file_num_locati
on+9);
new_daf_file_num=daf_file_num;
new=strrep(string_RFSP,window,new_daf_file_num);

% Simulate Clock
eclock_file_num=sum(energy_clock);
find_eclock_num_location=strfind(string_RFSP,'@@@@@@@@@@');
window=string_RFSP(find_eclock_num_location+0:find_eclock_num_location+9
);
new_eclock_num=sprintf('%10.0f',eclock_file_num);
new2=strrep(new,window,new_eclock_num);

% Process Clock
process_clock_num=sum(energy_clock)+15906;
find_process_clock_location=strfind(string_RFSP,'$$$$$$$$$$');
window=string_RFSP(find_process_clock_location+0:find_process_clock_loca
tion+9);
new_process_clock_num=sprintf('%10.0f',process_clock_num);
new3=strrep(new2,window,new_process_clock_num);

% Dlete old SIMULDATA to conserve space - only keep latest 100 energy
clocks
oldenergy=sum(energy_clock)-15906*100;
```

```
if oldenergy>=0

find_oldenergy_num_location=strfind(string_RFSP,'%%%%%%%%%%');
window=string_RFSP(find_oldenergy_num_location+0:find_oldenergy_num_loca
tion+9);
new_oldenergy_num=sprintf('%10.0f',oldenergy);
new4=strrep(new3,window,new_oldenergy_num);

elseif oldenergy<0

find_oldenergy_num_location=strfind(string_RFSP,'%%%%%%%%%%');
window=string_RFSP(find_oldenergy_num_location-
41:find_oldenergy_num_location+40);
new_oldenergy_num='';
new4=strrep(new3,window,new_oldenergy_num);

end

n=oldenergy/636240;

if n<1
oldblock=0;
elseif n>=1
oldblock=636240*floor(n);
end

find_oldblock_num_location=strfind(string_RFSP,'++++++++++');
window=string_RFSP(find_oldblock_num_location+0:find_oldblock_num_locati
on+9);
new_oldblock_num=sprintf('%10.0f',oldblock);
new5=strrep(new4,window,new_oldblock_num);

if  max(max(TotalScore))>0;  % If there is a channel suitable for
fuelling, refuell that channel.

% Refuel Row
rrow=yChosenz;
find_rrow_location=strfind(string_RFSP,'&');
window=string_RFSP(find_rrow_location+0:find_rrow_location+0);
new_rrow=rrow;
new6=strrep(new5,window,new_rrow);

% Refuel Column
xthechosen=num2str(xChosen);
if max(size(xthechosen))==1;
xthechosen=[' ' num2str(xChosen)];
end
ccol=xthechosen;
find_ccol_location=strfind(string_RFSP,'!!');
window=string_RFSP(find_ccol_location+0:find_ccol_location+1);
new_ccol=ccol;
new7=strrep(new6,window,new_ccol);

elseif max(max(TotalScore))<=0; % If no suitable channel for refuelling,
do not refuel.
```

```
% Delete fuelling card
find_ccol_location=strfind(string_RFSP,'!!');
window=string_RFSP(find_ccol_location-179:find_ccol_location+66);
new_ccol='';
new7=strrep(new5,window,new_ccol);

end

case1_string=strcat('rfsp_input');
fid=fopen(case1_string,'w');
fprintf(fid,'%s',new7);

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%        NEXT        RUN        CREATE        END
%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%

fclose('all');
display('New RFSP Input File Created')
disp('.') % spacing till next comment

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%         RUN          COMMAND          PROMPT
%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%

display('Running RFSP...')
disp('.') % spacing till next comment

cmd_case1_string=horzcat('runrfsp');
[status1,cmdout1]=system('dir');
[status2,cmdout2]=system(cmd_case1_string);

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%          RUN          COMMAND          PROMPT
%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%
```

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%

display('RFSP Run Completed')
disp('.') % spacing till next comment
fclose('all');

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%
%%%%%%%%%%%%%%%%%%%%%%%%      SAVE      PLOTDATA      FOR      ANALYSIS
%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%

display('Saving Core-Following Plotdata Upto Current FPD...')
disp('.') % spacing till next comment

save('C:\0_DAY_STARTPOINT_CDrive\Saved_Plotdata\Day.mat','Day');
save('C:\0_DAY_STARTPOINT_CDrive\Saved_Plotdata\ChPowerplot.mat','ChPowe
rplot');
save('C:\0_DAY_STARTPOINT_CDrive\Saved_Plotdata\MBPowerplot.mat','MBPowe
rplot');
save('C:\0_DAY_STARTPOINT_CDrive\Saved_Plotdata\Kchangeplot.mat','Kchang
eplot');
save('C:\0_DAY_STARTPOINT_CDrive\Saved_Plotdata\Burnupplot.mat','Burnupp
lot');
save('C:\0_DAY_STARTPOINT_CDrive\Saved_Plotdata\RBurnupplot.mat','RBurnu
pplot');
save('C:\0_DAY_STARTPOINT_CDrive\Saved_Plotdata\ATiltplot.mat','ATiltplo
t');
save('C:\0_DAY_STARTPOINT_CDrive\Saved_Plotdata\CPPFplot.mat','CPPFplot'
);
save('C:\0_DAY_STARTPOINT_CDrive\Saved_Plotdata\TotalScoreplot.mat','Tot
alScorez');


save('C:\0_DAY_STARTPOINT_CDrive\Saved_Plotdata\yChosenrecord.mat','yCho
senrecord');
save('C:\0_DAY_STARTPOINT_CDrive\Saved_Plotdata\xChosenrecord.mat','xCho
senrecord');
save('C:\0_DAY_STARTPOINT_CDrive\Saved_Plotdata\Fuelling_History.mat','F
uelling_History');
save('C:\0_DAY_STARTPOINT_CDrive\Saved_Plotdata\MChploc_History.mat','MC
hploc_History');
save('C:\0_DAY_STARTPOINT_CDrive\Saved_Plotdata\MBdploc_History.mat','MC
hploc_History');

save('C:\0_DAY_STARTPOINT_CDrive\Saved_Plotdata\Keffplot.mat','Keffplot'
);
save('C:\0_DAY_STARTPOINT_CDrive\Saved_Plotdata\RDeclineplot.mat','RDecl
ineplot');
```

```
save('C:\0_DAY_STARTPOINT_CDrive\Saved_Plotdata\e_reactplot.mat','e_reac
tplot');

save('C:\0_DAY_STARTPOINT_CDrive\Saved_Plotdata\avgzcrplot.mat','avgzcrp
lot');
save('C:\0_DAY_STARTPOINT_CDrive\Saved_Plotdata\zcr1plot.mat','zcr1plot'
);
save('C:\0_DAY_STARTPOINT_CDrive\Saved_Plotdata\zcr2plot.mat','zcr2plot'
);
save('C:\0_DAY_STARTPOINT_CDrive\Saved_Plotdata\zcr3plot.mat','zcr3plot'
);
save('C:\0_DAY_STARTPOINT_CDrive\Saved_Plotdata\zcr4plot.mat','zcr4plot'
);
save('C:\0_DAY_STARTPOINT_CDrive\Saved_Plotdata\zcr5plot.mat','zcr5plot'
);
save('C:\0_DAY_STARTPOINT_CDrive\Saved_Plotdata\zcr6plot.mat','zcr6plot'
);
save('C:\0_DAY_STARTPOINT_CDrive\Saved_Plotdata\zcr7plot.mat','zcr7plot'
);
save('C:\0_DAY_STARTPOINT_CDrive\Saved_Plotdata\zcr8plot.mat','zcr8plot'
);
save('C:\0_DAY_STARTPOINT_CDrive\Saved_Plotdata\zcr9plot.mat','zcr9plot'
);
save('C:\0_DAY_STARTPOINT_CDrive\Saved_Plotdata\zcr10plot.mat','zcr10plo
t');
save('C:\0_DAY_STARTPOINT_CDrive\Saved_Plotdata\zcr11plot.mat','zcr11plo
t');
save('C:\0_DAY_STARTPOINT_CDrive\Saved_Plotdata\zcr12plot.mat','zcr12plo
t');
save('C:\0_DAY_STARTPOINT_CDrive\Saved_Plotdata\zcr13plot.mat','zcr13plo
t');
save('C:\0_DAY_STARTPOINT_CDrive\Saved_Plotdata\zcr14plot.mat','zcr14plo
t');

display('Core-Following Plotdata Saved')
disp('.') % spacing till next comment

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%
%%%%%%%%%%%%%%%%%%%%%%%%       SAVE       PLOTDATA       FOR       ANALYSIS
%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%
%%%%%%%%%%%%%%%%%%%%%%%%        MAKE        BACKUPS        FOR        CRASH
%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%
```

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%

display('Creating Backup for "input" and ".output" Files...')

copyfile('C:\0_DAY_STARTPOINT_CDrive\rfsp_output','C:\0_DAY_STARTPOINT_C
Drive\Crash_Backup')
copyfile('C:\0_DAY_STARTPOINT_CDrive\rfsp_input','C:\0_DAY_STARTPOINT_CD
rive\Crash_Backup')

STORE='STORE';
nextsumfpd=num2str(FPD+0.25);
next_daf_file_num=strcat(nextsumfpd, 'FPD_DAF');

system(['rename ' STORE ' ' next_daf_file_num]);

Prev_DAF=strcat('C:\0_DAY_STARTPOINT_CDrive\',daf_file_num);
delete(Prev_DAF)

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%
%%%%%%%%%%%%%%%%%%%%%%%%%         MAKE        BACKUPS        FOR        CRASH
%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%

display('Backups Created')
disp('.') % spacing till next comment

display('********************PROCESS        OF        CURRENT        FPD
COMPLETED********************')
elapsed_time=toc(timerVal);
display(strcat('Elapsed Computing Time:',num2str(elapsed_time),'s'))

loopcrit=loopcrit+1;

display('*****************************BEGIN                         NEXT
FPD*****************************')

fclose('all');

end

display('******************************PROGRAM
END******************************')
```
