Accepted for the TopFuel 2003 Conference, Würzburg, Germany, March 16-19, 2003

MINIMIZING PWR RELOADING TIME BY OPTIMIZING CORE DESIGN AND RESHUFFLING SEQUENCE

F.C.M. Verhagen
P.H. Wakker
J.T. van Bloois*

NRG
P.O. Box 9035
6800 ET ARNHEM
The Netherlands
email: f.verhagen@nrg-nl.com

* N.V. EPZ
P.O. Box 130
4380 AC VLISSINGEN
The Netherlands

Introduction

During the last decade refueling outages of nuclear power plants have been reduced from 4-6 weeks average to 1-2 weeks for top-performers nowadays. For these top-performers one of the time consuming efforts on the critical path is core reloading. In many PWR plants the complete core is still offloaded during an outage. In this way core reloading takes about 2.5 to 5 days, mainly depending on core size. Several European plants recently apply the so-called incore shuffling strategy, thereby saving approximately 1-2 days of outage time[1]. To further minimize the reloading time there are two options. First, one could optimize the core design by keeping assemblies on their positions and minimizing insert shuffles. Second, one could optimize the sequence of refueling machine steps needed to shuffle the old core to the new core. For both types of optimization NRG has developed software tools that are successfully applied for the two most recent cycles of the Dutch Borssele plant.

Core design with ROSA

The ROSA[2,3] code system has been developed at NRG for PWR core design and utilizes Simulated Annealing[4] as optimization technique in combination with a very fast and accurate 3-D core simulator code. Besides for Borssele, ROSA is also being used for several PWR's in the USA. With this code millions of loading patterns (including cycle depletion) can be evaluated on a workstation in a day. The code can be switched between manual and automatic mode. In addition to single cycle also equilibrium cycle optimization is possible.

A powerful graphical user interface (GUI) allows the user to interact with the optimization process by e.g. changing targets during the run or perform manual fuel movements. The graphical user interface of ROSA is shown in figure 1.

Loading pattern candidates are generated by randomly exchanging fuel assemblies and/or by rotating burned fuel assemblies and/or by changing the composition of fresh fuel assemblies (e.g. by changing the enrichment and/or poison loading). Next, they are checked against flexible user-defined rules, needed to apply logistic constraints (e.g. position lock the central assembly) and/or a-priori knowledge (e.g. no feed fuel on the core periphery).

When a loading pattern satisfies the user defined rules neutronics calculations are performed in quarter core geometry (rotational boundary conditions) and typically 12-18 axial nodes. A 1½ group 3-D coarse mesh kernel method, coupled with a 2-D fine mesh method (3x3 nodes per assembly) is used. Pin power reconstruction is available. About 10 to 20 burnup steps per cycle are normally used. Typical accuracy compared to license code results is within ±0.02 for normalized assembly powers, ±0.03 for maximum $F_{\Delta H}$, and ±3 days for natural cycle length. Typical full cycle evaluation takes 25 to 250 ms CPU time per loading pattern (on a 550 MHz HP C3600 workstation), depending on core-size, axial mesh, number of burnup steps, and the previously evaluated loading pattern (convergence).

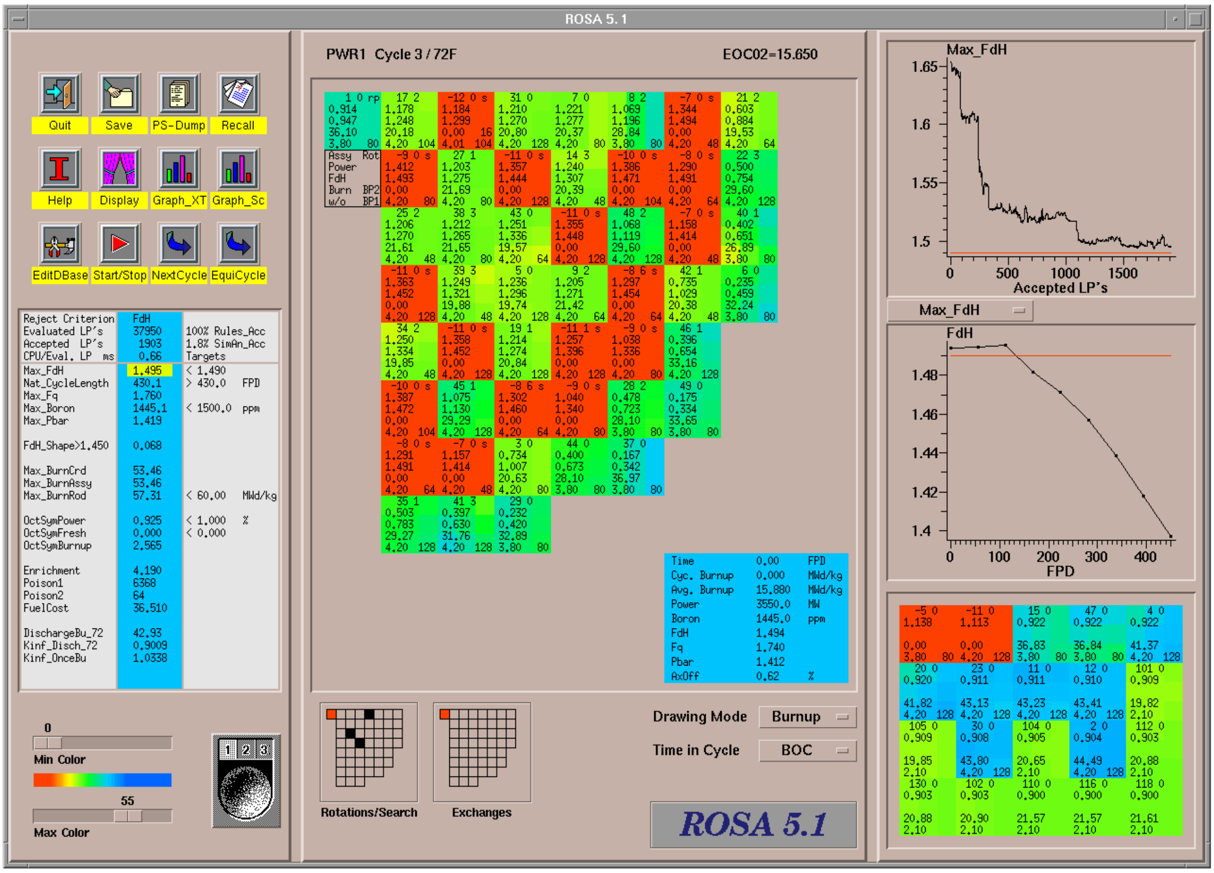

Figure 1 ROSA graphical user interface.

Up to 26 parameters can be optimized simultaneously, amongst others: peaking factors, natural cycle length, burnup limits, economics (fuel cost), shutdown margin, and octant symmetry. Acceptance of a candidate loading pattern depends on the results for the activated optimization parameters, compared to the results of the previously accepted loading pattern, and the so-called annealing temperature of each parameter. If all parameters improve the candidate loading pattern is always accepted. Worsening of certain parameter results may be acceptable, depending on temperatures. During a run temperatures are automatically lowered to converge towards a “frozen” solution. An effective calculation time in the order of 1 ms CPU time per evaluated loading pattern is usually achieved, due to a high fraction of rejected cases with parameter violations early in the cycle. Therefore about 100 million loading patterns can be evaluated in a day.

One of the recently developed parameters is “Reload_Time”. This is an estimate of the time (hours) needed to shuffle the previous cycle core to the new (candidate) core, accounting for all required assembly and insert moves. The user can specify the average time needed for an incore assembly move, a core to pool assembly move, a control rod cluster move, a thimble plug move, a pull action of a removable poison cluster, and optionally a mast sipping action for fuel that does not need to be moved (in case of leakers in the core). Different thimble plug types and control rod types can be specified on a full core basis. This can be used to take into account the impact on reloading time of black and gray control rods and/or a control rod management strategy. Also asymmetries in plug type positions between quadrants (asymmetric instrumentation) can be taken into account. The estimated reloading time for each evaluated loading pattern is the sum of the number of different actions multiplied by their required time. The user has the choice to select between incore and excore shuffling mode. The extra calculation time for this parameter is negligible compared to the average evaluation time per loading pattern.

Optimization of a loading pattern to minimize reloading time may reduce the outage by several hours, but may also cost some of the achievable natural cycle length of the pattern. This is shown in figure 2. The reloading time optimized cycle starts up earlier at BOC (Δt_1: the curve is shifted right so here shown as EOC production gain). If the core designer allows a reduction in natural cycle length, the optimized cycle starts earlier with its coast down (Δt_2). When this is the case (Δt_2 is positive), then it depends on the length of the coast down period whether a net production increase will be achieved. In addition to the potential of extra production a reduction in outage time of several hours may also be an interesting saving in personnel cost.

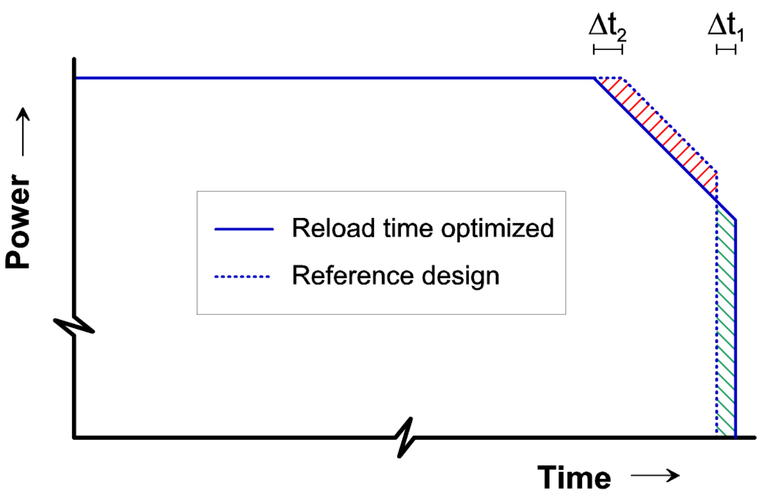

Figure 2 Reactor power versus time in cycle for two scenarios (not to scale)

Shuffling sequence optimization

Basically optimizing incore shuffling is solving a travelling salesman problem. The refueling machine must visit a number of core and spent fuel pool locations within the shortest possible amount of time. Contrary to the traditional travelling salesman problem however, the refueling machine must visit several places twice or even three times (to address assembly and inserts shuffles) and in a specific order (e.g. a fresh fuel assembly cannot be placed before the old one has been moved). Furthermore, closed loops of assemblies or inserts can be embedded in the problem, when for example one group of assemblies is exchanged with another. In case of such a closed loop an additional step is needed to create an open position, after which the assemblies or inserts in the closed loop can be shuffled.

Next to the shuffling sequence, also the positions of the to be loaded and unloaded fuel in the spent fuel pool can be optimized. For example, a spent assembly giving its control rod cluster to a fresh assembly should be placed next to the receiving assembly.

NRG has developed a software tool SOSA (Shuffling Optimization by Simulated Annealing) that performs the above optimizations while respecting a number of constraints. For example the maximum number of control rods that may leave the core at the same time (subcriticality), the support for each assembly from its neighbors at each step during the shuffling process, and the number of empty core locations around an ex-core detector.

First, an initial solution is generated, which takes about 1 s CPU time. Next, the optimization process generates small changes in the shuffling plan and subsequently evaluates the shuffling time. Generating and evaluating a shuffling solution in the optimization process takes about 60 μs CPU time, therefore about 1.4 billion shuffling plans can be evaluated in a day.

SOSA can handle different groups of inserts (e.g. control rod clusters and thimble plugs with or without instrumentation holes) and sipping of the assemblies remaining on their position can be optionally turned on or off. Distinction is made between times for separate actions, such as driving the refueling machine, positioning the refueling machine, lifting and lowering of assemblies and inserts either in the core or in the spent fuel pool, rotating inserts, and switching from assembly to insert handling and vice versa. For all steps in the shuffling plan the times of the appropriate actions are summed. The result is the total time needed to go from one loading pattern to another, which is the parameter to be optimized.

Borssele PWR plant information

The Dutch Borssele plant is a 480 MWe Siemens PWR that started to operate in 1973. The plant is currently (early 2003) in its 29th cycle of operation (12-month cycles). The core consists of 121 fuel assemblies (15x15-20), 28 control rods, and 24 instrumented assemblies (aeroball system entering at the top of the core). The core geometry with control rod and instrumented locations is shown in figure 3.

Due to the asymmetric position of the instrumentation finger inside an assembly (no central guide tube), three different types of thimble plugs are used: with no, one, and two holes. The plugs with two holes accommodate a thermocouple in addition to an aeroball system finger. Note that the instrumented locations are not distributed symmetrically over the core quadrants.

Core design is full low leakage without poison. Approximately 25% of the fuel is replaced each cycle. The maximum allowed fresh fuel enrichment is 4.0 w/o.

The first incore shuffling campaign was applied during the loading of the cycle 28 core in 2001 and the outage time was 14.6 days, including defective fuel repair. A record load-factor of 95.0% was achieved in 2001, including outage and coast down. The loading of the cycle 29 core in 2002 was an excore campaign due to inspection needs and lasted 18.4 days. A load-factor of 93.5% was achieved in 2002. The next refueling outage (autumn 2003) is planned to have a total length of 10.8 days with an incore shuffling campaign.

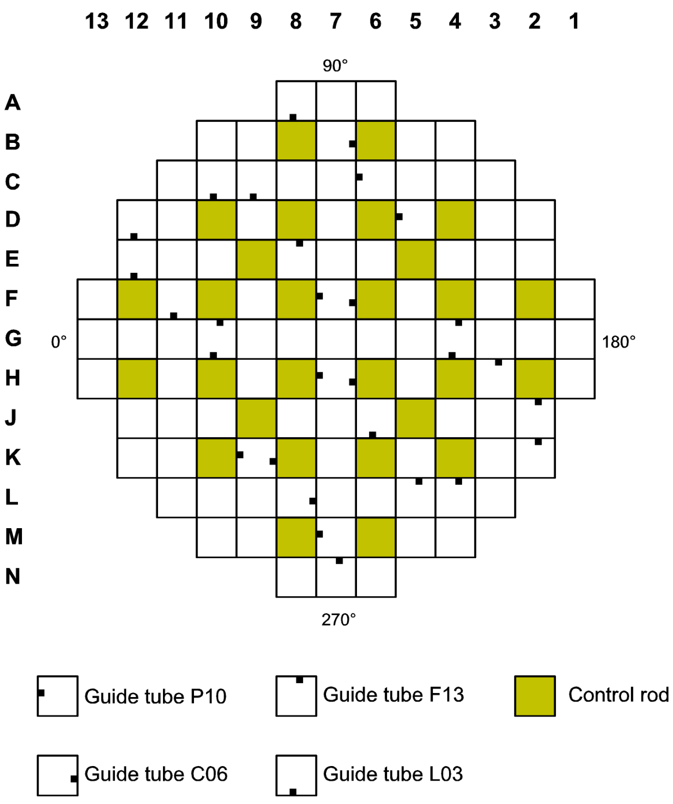

Figure 3 Borssele core geometry

Borssele core design with reloading time optimization

The cycle 28 core design in 2001 was the first time that reloading time optimization was included in the core design process. Other parameters that were optimized / constrained in this design were $F_{\Delta H}$, natural cycle length, maximum assembly, rod, and pellet burnup, and octant symmetry of the feed fuel.

The individual times needed for the core optimization code ROSA to calculate core reloading time are listed in table 1. It should be noted that calculated incore assembly moving time is partially based upon actual distance moved and actual positioning time. In the core design phase the second part of the ride (full) is known and taken into account, as old and new assembly positions are known for each evaluated loading pattern candidate. For the rest of the step (empty and lifting) an average time is assumed (base time). Loading patterns will therefore show a slight preference for incore assembly moves over short distances and in one direction. For all other actions average times are used.

Table 1 ROSA times needed for separate shuffling actions.

Action	Time [min]
Incore assembly move (base time)	8.0
Incore assembly driving per assembly width	0.054
Incore assembly positioning per co-ordinate	0.11
Assembly move from core to pool or vice versa	10.3
Control rod move	11.0
Thimble plug move	11.0
Assembly mast sipping (if needed)	8.0

During the core design phase typically hundreds of millions of loading patterns are evaluated. Optimization targets of $F_{\Delta H}$<1.755 and $T_{natural}$>320.5 days were used in an example case to investigate the possible variation in reloading time for incore shuffling. Results for the actual core inventory are plotted in two scatter plots in figure 4, where each dot represents a loading pattern. Sipping was not included in this optimization.

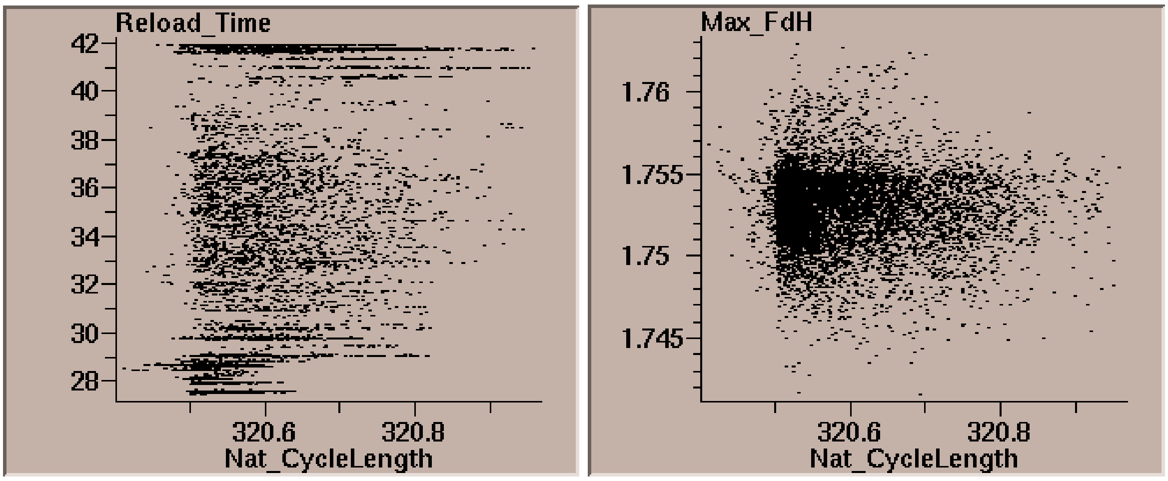

Figure 4 Borssele cycle 28 optimization example for 10,000 loading patterns

The left plot shows that variation in reloading time between 27.4 and 41.9 hours is possible, whereas the correlation with natural cycle length (days) appears to be weak. The right plot shows the correlation between $F_{\Delta H}$ and $T_{natural}$ for the same set of loading patterns. A best reloading time case with 27.43 h was found that also satisfied the above mentioned optimization criteria for the other two parameters. The finally selected loading pattern had an estimated reloading time of 28.5 h. A reference case with maximum achievable natural cycle length of +0.8 days could be found with a reloading time of +7.0 h. The +0.8 days difference in natural cycle length is equivalent to 0.8*24*(1-0.86)=2.7 h of energy production, as EOC reactor power was approximately 86%. A net extra production of 0.86*7-2.7=3.3 h (see also figure 2) could be achieved due to the reloading time optimization. For this cycle 28 slightly more production (4-5 h) might have been achievable by maximizing an energy production parameter, defined as the difference between reloading time gain and coast down loss.

In the finally selected cycle 28 loading pattern 32 fuel assemblies did not have to change position and 12 of the 28 assemblies with a control rod could be kept on a rodded location. In figure 5 (left part) the number of insert shuffles is plotted for cycles 6-29. A reduction in total number of insert shuffles from ~95 to ~60 is achieved for the last two optimized cycles, compared to non-optimized cycles 20-27. The theoretical maximum is 121 for this reactor. In the right part of figure 5 total and best achievable number of insert shuffles is shown. This minimum is the number of assemblies that come from the pool into the core (new + old fuel from pool). These assemblies always have to get an insert, assuming there is neither a stockpile of inserts for preparation nor usage of removable poison rod clusters. The remaining fuel may need a new insert, but that depends on the core design. In the cycle 29 design it was possible to keep the total fraction of insert changes for the remaining fuel below 23% (only 19 out of 84 inserts had to be changed).

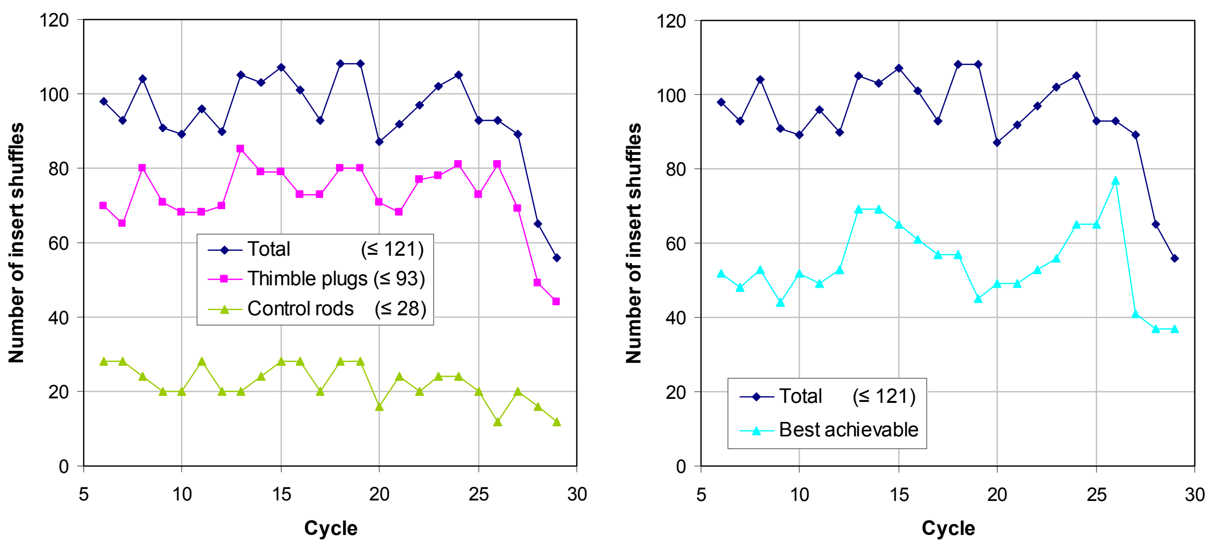

Figure 5 Borssele insert shuffles per cycle

Borssele shuffling sequence optimization

For the cycle 28 reloading campaign SOSA has been used and compared to a hand-made shuffling plan. Initially the times for separate actions had to be roughly estimated since little data was available. During the cycle 28 reloading campaign the time needed for each step in the shuffling plan has been recorded. From that data more accurate estimates for the action times have been derived as listed in table 2.

Table 2 Times needed for separate shuffling actions.

Action	Time [min]
Moving refueling machine one assembly width	0.054
Positioning refueling machine per co-ordinate	0.11
Lift/lower assembly in the core	4.0
Lift/lower assembly in the spent fuel pool	2.9
Lift/lower insert in the core	5.3
Lift/lower insert in the spent fuel pool	3.8
Lifting time saved when taking insert directly from assembly in the core	3.5
Lifting time saved when taking insert directly from assembly in the pool	2.0
Switching from assembly to insert handling or vice versa	0.25
Rotating an assembly or insert	1.0

Traditionally the shuffling plan was made with as little switches between assembly and insert handling as possible. Also the SOSA shuffling plan for the cycle 28 reloading campaign was designed with this constraint. A second constraint imposed on the design was to keep all control rods but one in the core. Two other constraints were to keep all assemblies supported by at least one straight neighbor and to create as little empty core locations as possible around the ex-core detectors next to positions A6 and N8. However, these last two constraints are not limiting since only a few empty core locations are created at the same time.

Because during cycle 27 several leakers were detected sipping of all 32 assemblies remaining on their old position had to be performed. In SOSA sipping is simulated as lifting and immediately lowering of the assembly. The additional time needed for sipping (about 4 h) explains most of the difference between the ROSA and SOSA reloading time.

The results of the SOSA optimization are listed in table 3, column SOSA-1. Savings of about 2.7 hours in comparison with the hand-made shuffling plan were realized. Since the number of switches was limited no savings in lifting time could be realized by for example taking an insert directly after moving the assembly. Therefore the 2.7 hours saving was mainly realized by optimizing the travelling and positioning time of the refueling machine. This can be seen from figures 6 and 7. The first shows the total traveled path of the refueling machine, the second the empty trips only. One can clearly see that the total traveled path is shorter for the SOSA plan (less blue) and that the SOSA plan has a preference for trips perpendicular to the axes (less positioning time needed). Another striking feature is the significantly smaller number of empty trips to the spent fuel pool for the SOSA plan. For most trips to or from the pool SOSA puts the return trip to good use, while the engineer is often forced to return empty in the hand made plan.

After the cycle 28 reloading campaign is was concluded that two imposed constraints were not necessary. The constraint on switches could be lifted because switching proved not to take as much time and trouble as was initially thought. The constraint on control rods was unnecessary since the subcriticality calculations are performed under the assumption of all control rods out. To see what could have been saved, the shuffling plan was redesigned without the above constraints. The results of this redesign are listed in table 3, column SOSA-2. As could be expected additional savings are made for the lifting time, while losing some time in switching between assembly and insert handling. The extra savings are about 2.5 hours.

Table 3 Borssele cycle 28 shuffling sequence optimization results. Shuffling times [h], total and split into separate actions, number of shuffle steps, maximum number of control rods out of the core, and maximum number of empty core locations.

	Hand-made	SOSA-1	SOSA-2
Total time	35.95	33.23	30.76
Empty rides	2.19	0.91	0.67
Full rides	4.86	3.88	4.15
Positioning	1.25	0.94	0.92
Lifting/lowering	27.64	27.41	24.78
Switching	0.01	0.02	0.24
Rotations	0.00	0.07	0.00
Total number of steps	205	205	205
Number of sipping steps	32	32	32
Max. # control rods out of core (≥ 1)	1	1	4
Max. # empty core locations (≥ 1)	3	6	7

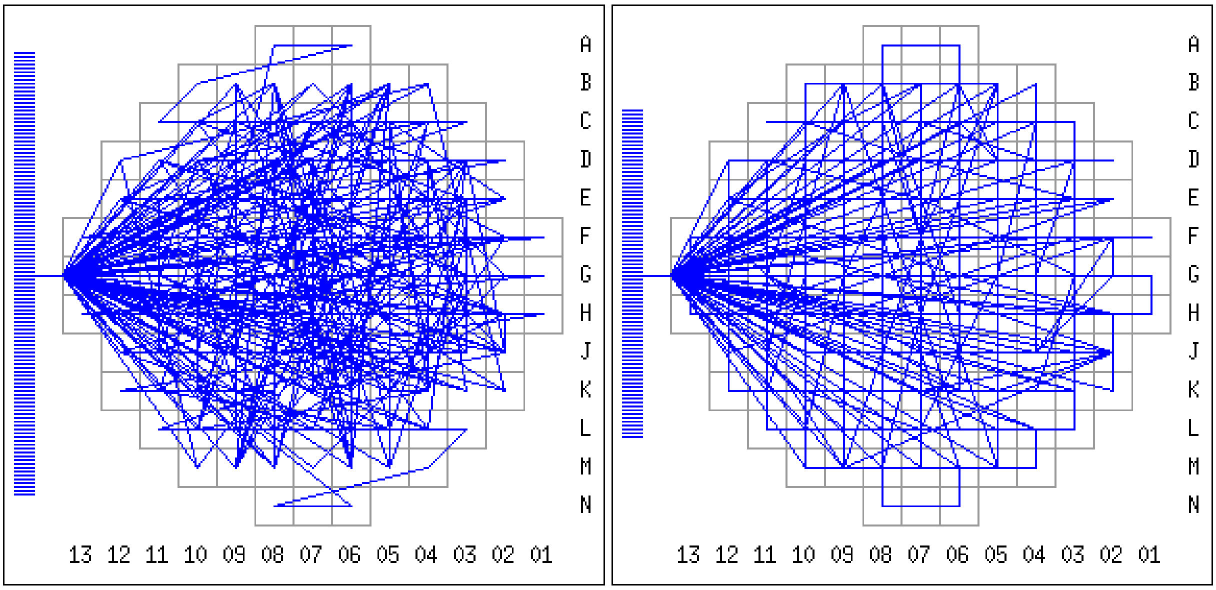

Figure 6 Total traveled path by the refueling machine in the core region. Left the hand-made shuffling plan, right the SOSA-1 shuffling plan. The column of lines on the left side of the figures indicates the number of trips (back or forth) to the spent fuel pool.

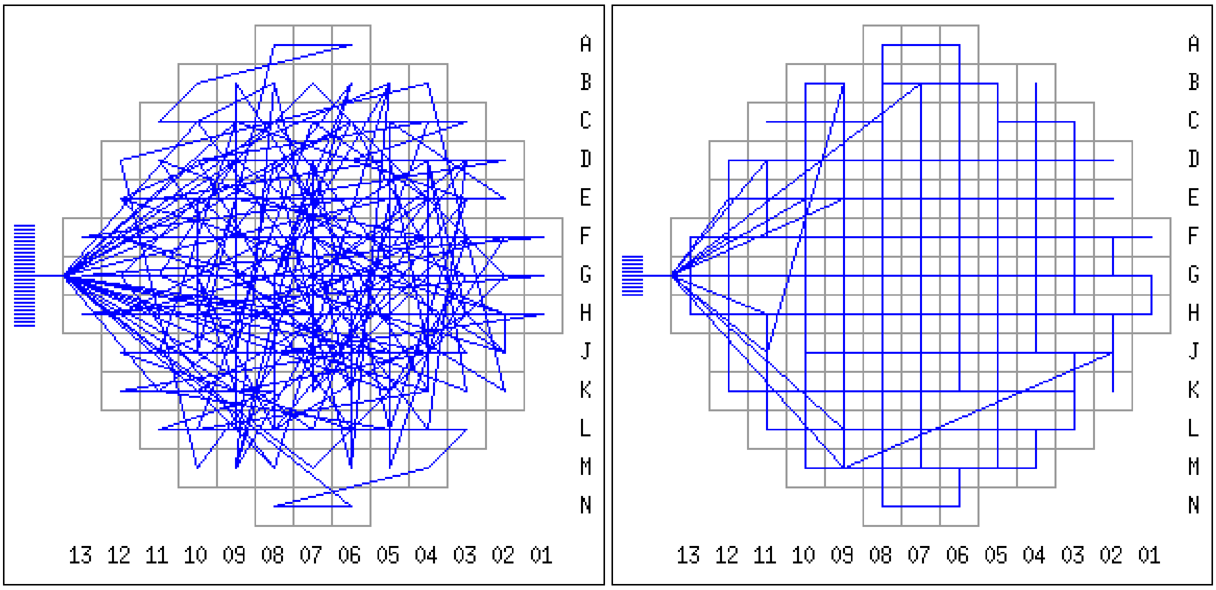

Figure 7 Traveled path by the refueling machine in the core region, empty trips only. Left the hand-made shuffling plan, right the SOSA-1 shuffling plan. The column of lines on the left side of the figures indicates the number of trips (back or forth) to the spent fuel pool.

Results and conclusions

Both above mentioned optimization methods have been successfully applied for the cycle 28 design and (first) reloading campaign with incore shuffling of the Dutch Borssele plant in 2001. In the core design 26% of the assemblies were kept on the same core location and the number of insert shuffles (control rods + thimble plugs) was reduced by approximately 30% compared to previous cycles. With respect to a maximum energy core design with the same inventory the outage time could be reduced by approximately 7 h. This was equivalent 3.3 h of extra production. In the subsequent shuffling sequence optimization about 8% of the time (2.7 h) was saved with respect to the reference hand-made shuffling plan. Later redesign with lifted constraints showed a possible extra reduction of 2.5 h.

For the Borssele cycle 29 design (with an excore shuffling campaign due to inspection needs) it was possible to reduce the number of insert shuffles by about 40% with respect to old cycles.

It can be concluded that both core design and shuffling sequence optimization can be exploited to reduce the time needed for reloading a core with an incore shuffling campaign. Excore shuffling campaigns can still have substantial benefit from a core design with a minimized number of insert shuffles.

References

1. H.-G. Johann, D. Porsch, “Schrittfolgeplanung für das Umsetzen von Brennelementen im DWR,” *Jahrestagung Kerntechnik ’99*, Fachsitzung “Kernbeladung: Anforderungen - Methoden - Lösungen”, p. 71 (1999).
2. M. van der Schaar, F.C.M. Verhagen, W.J.M. de Kruijf, T.F.H. van de Wetering, “ROSA: a code for fuel management,” *Jahrestagung Kerntechnik ‘96*, p. 15 (1996).
3. F.C.M. Verhagen, M. van der Schaar, W.J.M. de Kruijf, T.F.H. van de Wetering, R.D. Jones, “ROSA: a utility tool for loading pattern optimization,” *Proc. of the ANS Topical Meeting – Advances in Nuclear Fuel Management II*, Vol. 1, p. 8-31 (1997).
4. N. Metropolis, A. Rosenbluth, M. Rosenbluth, A. Teller, E. Teller, “Equation of State Calculations by Fast Computing Machines,” *J. Chem. Phys.,* Vol. 21, p. 1087 (1953).