

MONK and MCBEND: Current Status and Recent Developments

Simon D. Richards¹, Chris M. J. Baker¹, Adam J. Bird¹, Pat Cowan¹, Nigel Davies¹, Geoff P. Dobson¹, Tim C. Fry¹, Albrecht Kyrieleis², and Paul N. Smith¹

¹AMEC Clean Energy - Europe, Kimmeridge House, Dorset Green Technology Park, Winfrith Newburgh, Dorset, DT2 8ZB, United Kingdom

²AMEC Clean Energy - Europe, Thomson House, Birchwood, Risley, Warrington, Cheshire, WA3 6GA, United Kingdom

MONK[®] and MCBEND are Monte Carlo software packages for: nuclear criticality and reactor physics; and radiation shielding and dosimetry applications, respectively. The codes are actively developed, maintained and supported by AMEC's ANSWERS[®] Software Service in line with the ANSWERS vision of providing easy-to-use software that meets the current and emerging needs of the user community. This paper summarises the current status of MONK and MCBEND and the recent developments which have been carried out to the codes, and their supporting nuclear data libraries and visualisation package.

KEYWORDS: MONK, MCBEND, Monte Carlo, criticality, reactor physics, shielding, dosimetry, parallel, HPC, visualisation, graphics

I. Introduction

The Monte Carlo codes MONK[®] and MCBEND are part of the ANSWERS[®] codes suite.⁽¹⁾ ANSWERS codes are widely used in over thirty countries around the world and on a range of reactor types including: AGR, BWR, CANDU, MAGNOX, RBMK, PBMR, PWR, VVER and many experimental reactors. The codes have already been applied to some of the future reactor technologies (e.g. high temperature and fast breeder reactors in the Generation IV programme) that are being developed not only for electricity generation but additionally for other applications. The codes have also been used for a range of other applications including fusion reactor shielding and dosimetry assessments, medical shielding assessments, food irradiation and oil well logging.

1. MONK

1.1. Background

MONK is an advanced Monte Carlo neutronics code for the solution of criticality safety and reactor physics problems. The origins of MONK can be traced to a code called GEM, which came out of the post-war nuclear weapons programme in the United Kingdom. A more recent major programme of development was stimulated by the criticality needs of the UK reprocessing industry and resulted in the appearance of MONK6 in the late 1980s. This was followed by further enhancement and modernisation of the code and data (MONK7), together with the appearance of supporting productivity tools. In parallel with the above activity, a separate version of the code (MONK5W) was produced and subsequently enhanced specifically for reactor physics analyses, employing nuclear data from the WIMS⁽²⁾ reactor physics suite of codes. Options specific to this reactor physics version of the code included a microscopic burn-up

treatment, an adjoint solution capability, and point estimation of flux and fission rates.

In the late 1990s MONK8 brought together the above separate codes into a single version for use in both criticality and reactor physics applications; thus MONK8 superseded both MONK7 and MONK5W.

In 2006 the current the code, MONK9, was released. This version retains all the above capabilities of MONK8, and includes a range of new features, covering geometry, scoring and nuclear data. A large programme of work resulted in the production of the BINGO continuous energy nuclear data library and collision processor. This collision processor was incorporated into MONK9 and, along with the corresponding nuclear data libraries, provides a significant benefit to the analysis of models.

MONK10 is expected to be released late in 2013 or early in 2014 following an extensive period of development, verification and validation testing. This new version of the code contains a significant number of major enhancements, many of which are described in this paper.

MONK is developed, distributed and actively supported in use by AMEC as part of its ANSWERS Software Service. Prior to 2012 the code development was managed by a collaboration comprising ANSWERS and Sellafield Ltd ¹.

1.2. Overview

The primary aim of MONK is to calculate the neutron multiplication factor (k -effective) of systems by simulating the birth, migration and ultimate fate of a finite sample of typical neutrons. The actual number of neutrons followed or tracked determines the stochastic uncertainty associated with the calculated value of k -effective and other quantities such as fluxes and reaction

¹ Formerly British Nuclear Fuels plc (BNFL)

rates.

A model of the system to be assessed is assembled from simpler sub-systems using the MONK geometry package. The basic component of this package is a set of simple bodies, including the sphere, box, rod, prism, cone and torus. These bodies can have general orientation and can overlap each other if necessary. They are used as basic building blocks to form simple Parts of the geometry, each Part being defined quite independently of the rest of the system using local co-ordinates. These simple Parts may then be used to in the same way make more complex Parts and the process repeated until the whole system is assembled. Hole geometries (making use of a technique called Woodcock tracking⁽³⁾ or delta tracking) are used extensively in MONK to provide a lot of the more complicated fine geometric details, and to expedite the specification of some of the commonly-occurring replicating items.

For criticality calculations, neutron interactions are considered in the MONK collision processing packages DICE and BINGO. Standard MONK9 nuclear data libraries are available in hyper-fine group format based on UKNDL (8220 groups), JEF2.2, ENDF/B-VI and JENDL3.2 evaluated data files (all 13,193 groups). New BINGO continuous energy nuclear data libraries to be supplied with MONK10 are listed in section II-3.1. These libraries, together with the point-energy collision processing algorithms, provide a very detailed model of the physics, such that the ultimate accuracy of the MONK code largely depends on the accuracy of the basic nuclear data. This continuous energy package has been the subject of extensive validation studies and is therefore the recommended method for criticality assessments.

MONK calculates the k -effective for the system modelled using a staged (or iterative) calculation, with each *stage* consisting of a fixed number of neutron *superhistories*. A neutron superhistory is the set of tracks followed by a neutron and its fission progeny from birth to absorption or leakage, through a fixed number of fission generations (normally 10). Superhistory powering⁽⁴⁾ produces a stable calculation of the scored parameters and their variances, which are essentially unbiased and results in a calculation that concentrates on the most reactive parts of a system, thereby enabling MONK to be used with confidence even for highly-decoupled systems.

MONK has a proven track record of application to the whole of the nuclear fuel cycle and is well established in the UK criticality community. Furthermore it is increasingly being used for reactor physics applications. Example applications include:

- uranium enrichment, covering diffusion and centrifuge plant;
- fuel fabrication for thermal, fast and experimental reactors;
- new and spent fuel transportation both within countries and between countries;
- design studies for fissile material transport containers;
- spent fuel handling and long-term pool storage;
- spent fuel dry storage;

- fuel consolidation and dry cell handling;
- fuel dissolution;
- chemical separation involving mixer-settlers and pulsed-columns;
- reactor core loading assessment;
- product finishing and storage;
- waste treatment and handling;
- waste storage, including evaporation, vitrification, encapsulation and consolidation;
- plutonium metal production and handling;
- reactor analysis for all thermal reactor types; and
- burnup credit analyses.

2. MCBEND

2.1. Overview

MCBEND is a well-established, powerful Monte Carlo software tool for general radiation transport analysis for shielding and dosimetry applications. It is a sister code to MONK sharing a common input format, geometry package, material specification and nuclear data package. MCBEND has been applied to such problems as:

- the design of reactor plant, fuel transport flasks, reprocessing facilities and fusion devices;
- the interpretation and analysis of measurements on operating plant and in experimental facilities;
- the prediction of nuclear instrument and geophysical logging tool responses; and
- the calculation of personnel dose levels and radiation-induced material changes.

MCBEND has been developed by AMEC's ANSWERS Software Service in conjunction with Sellafield Ltd. The MCBEND package comprises not only the Monte Carlo code itself but also nuclear data libraries, user documentation, productivity tools of various kinds and user support services.

MCBEND is being continually developed to meet the needs of its users in line with the ANSWERS vision to provide easy-to-use software that meets the current and emerging needs of the user community. In the case of MCBEND this vision focuses on the key areas of accuracy, understanding of uncertainties, efficiency and user-friendliness. The current version is MCBEND11, which was released in 2013.

This paper details the developments incorporated into MCBEND11, covering improvements to the underlying physics and data libraries, improvements in calculation efficiency and improvements to the usability of the code.

II. Developments common to MONK10 and MCBEND11

1.2. Hole Geometry

1. Geometry modelling

MONK and MCBEND use a common package for geometry modelling and tracking comprising two complementary components: Fractal Geometry, which uses conventional ray tracing through geometrical bodies; and Hole Geometry, which uses the powerful and versatile Woodcock⁽³⁾ (delta) tracking.

1.1. Fractal Geometry

Fractal Geometry (FG) is a well-established system of solid geometry modelling in which the problem geometry is subdivided into *zones*, defined as the intersections and differences of mathematical bodies as described in section I-1.2. MONK and MCBEND have a large selection of body shapes ranging from simple bodies such as cuboids, cylinders and spheres to more complex bodies such as prisms, ellipses and tori. The bodies are assembled into structures called Parts, which are self-contained with their own local co-ordinate system to simplify the model construction. Parts may be included within other Parts to any depth of nesting and a given Part may be included more than once within the geometry. The ability to break down complex models into Parts simplifies the preparation and checking of the input data.

Improvements to the FG package in MONK10 and MCBEND11 include:

- a new TRI body to provide the user with a simpler alternative to the existing PRISM body when a triangular prism is required;
- a new TETMESH zone to support ray tracing through a zone containing materials in a tetrahedral mesh, such as can be produced by many CAD packages;
- a new IGES body to enable the user to import CAD geometry models in IGES format;⁽⁵⁾
- a new POLY body formed from a closed surface defined by a set of triangular facets;
- support for multiple successive rotations about different points; and
- improved automated checking of FG models to ensure that:
 - each body in a nest is enveloped by the next body;
 - all inner bodies of a simple cluster are enveloped by the container;
 - all the inner bodies of a simple cluster are non-intersecting; and
 - bodies in a cluster with declared overlaps intersect as stated.

The Woodcock tracking algorithm in MONK and MCBEND is implemented via the Hole Geometry package, and brings significant additional modelling power to the user. Hole geometries can be used to model common replicating arrangements and simple intersecting configurations in an expedient way. The Hole Geometry package also facilitates the exact modelling of certain complicated geometries which would be impossible to model in FG using simple bodies. MONK10 and MCBEND11 support the use of named Holes and named materials in Holes (see section II-1.3).

The Hole Geometry package continues to grow as additional Hole types are added in response to the needs of the industry. MONK10 and MCBEND11 incorporate the following improvements to the Hole Geometry package:

- a new PIPES Hole - The PIPES Hole geometry allows the user to model arrangements of pipes, with input parameters defining the location of the pipes; the radii of the annuli in the pipe; the materials or subsidiary Holes in each annulus; the pairs of straight pipes to be linked by curved or straight pipes; and an interstitial material that may contain a subsidiary Hole. An example is shown in Figure 1;

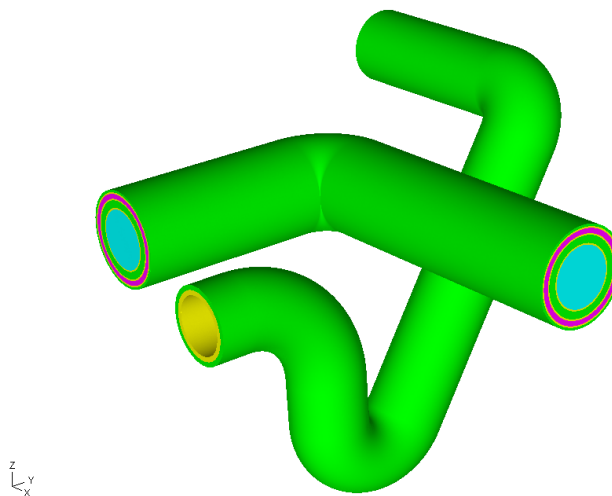


Figure 1: An example of the PIPES Hole.

- a new BENTPINS Hole². This Hole allows the user to model arrangements of fuel pins, some of which may be missing, that have undergone bending or buckling. The pins may be deformed either by being kinked or by being gently curved using a circular or sine-wave profile;
- a new VOXELA Hole used for the approximate representation of an object as an array of cubic voxels. This is designed to improve performance and storage efficiency for geometries sourced from, for example, computer tomography (CT) data;
- a new TETMESH Hole geometry enables CAD-generated tetrahedral mesh geometries to be imported into MONK and MCBEND and treated as a Hole Geometry;

²The BENTPINS Hole was present in MONK9 but is new to MCBEND11.

- a new RANDROD Hole³, shown in Figure 2, models a random distribution of rods in a cylindrical container (e.g. cut fuel pins in dissolvers). The rods are all of the same length, radius and composition;

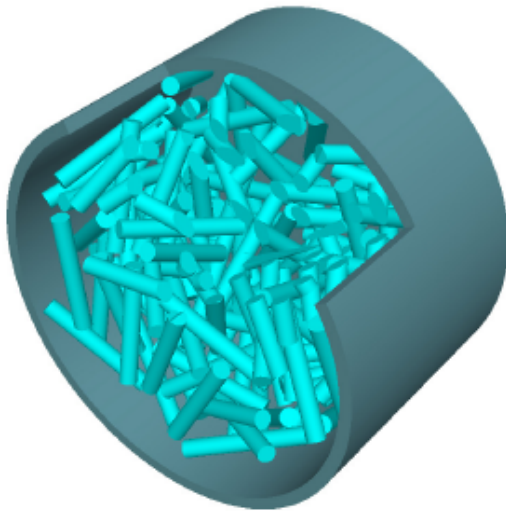


Figure 2: An example of the RANDROD Hole which could be used for cut fuel pins in a dissolver.

- improvements to the QUADRIC Hole. This Hole models pipes with elliptical cross-sections and an axis which also forms an ellipse. For certain degenerate forms the existing QUADRIC Hole is unnecessarily complicated; the improvements allow concentric tori to be defined more simply, and also for some elliptical options to be specified more naturally in terms of major and minor radii; and
- new PBMR and PEBBLE⁴ Hole geometries for modelling pebble-bed reactors in great detail. A pebble-bed reactor might be expected to contain around 500,000 individual pebbles, each containing around 15,000 multi-layered coated particles of fuel in a carbon matrix. The PEBBLE Hole models each individual coated fuel particle in a pebble, and the PBMR Hole models each individual pebble in the reactor. Figure 3 shows a cut-away ray-trace image of an example of the PBMR Hole modelling fuel pebbles in a pebble-bed reactor. The interstitial coolant material has been excluded from this figure for clarity. Each pebble is modelled using a subsidiary PEBBLE Hole containing 15,000 multilayer fuel grains, as shown in the inset image with the matrix material removed for clarity.

1.3. Named materials, Parts and Holes

Historically materials, Parts and Holes have all been numbered sequentially in the input file so that they can be referenced from other parts of the model. MONK and MCBEND developments have been systematically transforming the component numbering system to a component naming system and in the latest versions materials, Parts and Holes can all be given a user-defined name. This makes the construction and maintenance of

³The RANDROD Hole was present in MONK9 but is new to MCBEND11.

⁴The PBMR and PEBBLE holes are licensed separately.

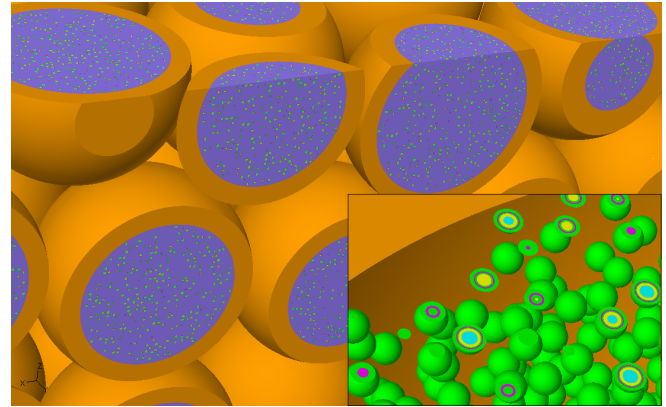


Figure 3: Cut-away ray-trace image of fuel pebbles in a pebble-bed reactor and (inset) the multi-layer coated grains within a pebble, modelled with the PBMR and PEBBLE Holes.

MONK and MCBEND models much more user-friendly and aids the quality assurance process.

2. RCARD package enhancements

RCARD is the system of free-format data input used by MONK and MCBEND. It provides a powerful method of using mathematical and logical functions to construct a flexible and efficient input model.

MONK10 and MCBEND11 have enhanced RCARD facilities including:

- formulae and parameters in the input file can use double precision arithmetic;
- formulae can contain exponentiation to any power;
- formulae can contain the functions INT, ATAN, ACOS and ASIN;
- variables can contain character string values;
- variables can contain logical conditions;
- parameters used for looping⁵ can contain variables;
- IF statement blocks can be nested;
- ELSEIF statements can be used;
- loop counters can be used in logical conditions;
- extended logical condition statements; and
- additional controls on how input is processed.

3. Physics features

3.1. Collision processing and nuclear data

The MONK code includes three main collision processing packages: DICE point energy; BINGO continuous energy; and WIMS broad group, each with their corresponding nuclear data

⁵Looping calculations are a powerful feature of MONK and MCBEND in which parameters can be given a range of values in the input file and MONK and MCBEND will repeat the calculation for each value in the supplied list. Nested loops are also possible.

libraries. The WIMS package is used mainly for depletion calculations, though recent developments have enabled BINGO to be used for the tracking phase of a depletion calculation. MCBEND also features both DICE and BINGO collision processing for neutrons, and also for gamma photons and electrons.

The DICE and BINGO packages are those used most frequently in MONK for criticality calculations, with the focus of current and future development being BINGO. Since the release of BINGO in MONK9 the BINGO collision processor has undergone a major rewrite.

BINGO libraries issued with MONK10 and MCBEND11 include JEF2.2, JEFF 3.1, ENDF/B VII and CENDL3.1. Data are available for neutrons in MONK and MCBEND, plus photons and electrons in MCBEND. Improvements in BINGO include:

- use of cross sections tabulated at energy points that are specific to each nuclide;
- improved variable temperature treatment (see section II-3.2);
- enhanced thermal scattering modelling including use of bound data for graphite;
- better representation of correlated energy/angle laws;
- more detailed representation of the tails of the fission spectrum; and
- explicit modelling of bremsstrahlung.

These capabilities give MONK10 and MCBEND11 the tools to model complex systems with a greater degree of realism than with the DICE package.

3.2. Run-time Doppler broadening

The BINGO collision processor includes a run-time Doppler broadening capability⁽⁶⁾ for treating temperature variations in material data. This technique does not pre-process the cross-section data for the specific material temperature, but at each collision event evaluates the effect of temperatures on the resonances and derives an appropriate set of cross-section data and event outcomes. Run-time Doppler broadening provides the capability to follow temperature variation throughout a reactor core and throughout its lifetime, opening links to thermal hydraulics feedback.⁽⁷⁾

With this method, the nuclear data library contains, for each nuclide, cross-section data at a relatively small number of specific base temperatures. The nearest lower temperature data to the user-specified temperature are selected, and those data used as the basis for the run-time broadening process. It is thus possible to model any material temperature within the overall temperature range present on the library (currently 293.6 K to 80,000 K). The accuracy of the run-time Doppler broadening decreases the further the target temperature is from the base temperature so a single base temperature cannot be used. Instead extra base temperatures are provided in the library and used before significant error can build up. The interval between base temperatures increases with temperature as the cross section becomes smoother with raised temperature.

4. Scoring

4.1. Unified Tally module

A new Unified Tally module (UT) allows scoring to be carried out in bodies and meshes that are independent of the geometry model and importance mesh. The method is based on the principle of scoring bodies or surfaces that may be subdivided to form localised scoring meshes of any required size. Any number of bodies may be defined and may overlap freely.

Each scoring body has a named shape echoing some of the simple bodies available in the FG package, e.g. BOX for a cuboidal body, or ZROD for a cylindrical body whose axis is parallel to the z -axis. The scoring bodies may be located by specifying their origins either in global model coordinates or the local coordinate system of any defined Part in the underlying geometry. The size of each scoring body is determined by shape parameters e.g. length, radius, height, and the bodies can be rotated as required. Each scoring body may be subdivided to allow localised scoring. All scoring bodies and their permitted methods of subdivision allow the code to calculate the volumes of the scoring mesh cells.

As many scoring bodies as required may be defined. The scoring in each body is independent of other bodies so that bodies may overlap. For example, it would be acceptable to superimpose two bodies of identical shape. One could use a fine internal mesh to obtain good resolution of results; the other could use a coarser mesh to obtain better statistics. Alternatively, superimposed bodies could be used for scoring in different group schemes. The UT module supports scoring of track length and collision density estimators of scalar flux in MONK and MCBEND, plus all other MCBEND scores apart from sensitivities and energy deposition, and it may be used in conjunction with all existing scoring facilities. UT scoring bodies may be added to existing models without the need for changes to the underlying model definition.

4.2. Scoring by material

Scoring capabilities in the UT module also include scoring by material. This divides the scores recorded in a given mesh between the events in individual materials. It is principally intended for scoring results in the components of Hole geometries, as Hole geometries do not have distinct *zones* of different materials. Since tracking through such materials does not identify surface crossings it is not possible to use track length estimation; material scoring is therefore confined to collision density estimation. Essentially the material identified at a collision point becomes an additional subscript in the scoring registers. Normalisation of the results requires the volume of each material in each mesh. An option is therefore provided in the code to estimate relevant volumes by the Monte Carlo method. Alternatively material volumes, where independently known, may be supplied by the user.

III. Developments specific to MONK 10

1. Scoring

1.1. Action tallies in Unified Tally meshes

The new UT module in MONK10 allows the scoring of neutron fluxes in user-defined meshes which are independent of the underlying model geometry (see section II-4.1). MONK10 also includes a new module for scoring action tallies in the UT meshes.

The AT-in-UT (action tally in unified tally) module allows the user to select scoring bodies (optionally subdivided to form meshes) defined in the UT module (including the relevant energy group scheme and optional material restriction list) in which to score the following reactions:

- capture;
- fission and number of fission children;
- elastic scatter;
- inelastic scatter;
- (n,2n); and
- (n,3n).

This flexible new action tally module allows the above reaction tallies to be requested in any defined UT scoring body (which may optionally be subdivided to form a 1D, 2D or 3D mesh), in any energy group scheme. The following scoring options are available:

- **body** - the total reaction tallies and their standard deviations in the scoring body, or in each subdivision of a subdivided body;
- **material** - the reaction tallies and their standard deviations for each material, or subset of materials, in the scoring body or mesh; and
- **nuclide** - the reaction tallies and their standard deviations for each nuclide in each material, or subset of materials, in the scoring body or mesh.

Results can either be requested to be absolute tallies of events, or normalised to 10,000 samples.

1.2. Action tallies

In addition to the new AT-in-UT module for scoring action tallies in UT bodies, the existing action tally module has also been enhanced to include:

- **region fission** - this new option allows the user to tally by energy and requested FG region(s) the samples that cause fission and those samples that emerge from fission events; and

- **neutron balance** - this option provides action tally data by energy group. The output comprises two tables for each material. The first table tallies the upper energy group boundary, number of production events, log mean energy and the relative number of productions per unit energy. The second table tallies log mean energy, number of captures, number of fission events, number of production events, number of (n,2n) reactions and number of (n,3n) reactions.

1.3. Shannon entropy

The use of MONK to estimate distributed quantities such as fluxes and reaction rates in spatial meshes typically requires a greater number of samples than is required to estimate k -effective to a given precision. Therefore the stochastic uncertainty on k -effective is not a good indication of the convergence of the estimates of distributed quantities.

A new feature of MONK10 is the facility to calculate the Shannon entropy of the scalar flux in a mesh defined by any subdivided UT body for each stage in the calculation. The Shannon entropy of the neutron flux in a mesh containing N cells may be written⁽⁸⁾

$$H_{\phi} = - \sum_{j=1}^N p_j \log_2(p_j)$$

where $p(j)$ is the fraction of the total flux appearing in cell j . Here we define $p_j \log_2(p_j) \equiv 0$.

The convergence of the Shannon entropy of the flux in a specified mesh can be used to estimate the convergence of the estimates of fluxes in that mesh. It may be noted that a potentially undesirable property of this equation is that the absolute value of the Shannon entropy depends on the number of cells, N , in the mesh. To overcome this dependence we also define a *modified* Shannon entropy:

$$H'_{\phi} = H_{\phi} - \log_2 N$$

Both of these entropies are available to the user in MONK10. Future developments to the Shannon entropy module will enable the Shannon entropy of the fission source distribution to be calculated to assist the user in determining whether the source term has converged prior to the scoring stages of the calculation.

1.4. Sensitivity to nuclear data

A first-order sensitivity option for estimating the sensitivity of k -effective to cross-section data has been available in MONK for the DICE point-energy nuclear data since MONK8B. The approach implemented used derivative operator sampling, as used in MONK's sister shielding code MCBEND, but with the addition of the limited weight recycling method pioneered in MONK6.

The sensitivity option in MONK10 has been further developed so that it is available for use with the continuous energy BINGO library. The sensitivity option in MONK with BINGO supports the sensitivity of k -effective to material density, one or more reaction cross-sections in a specific nuclide in a material,

or all available reactions excluding specific listed reactions. A user-defined energy group scheme can also be specified for calculating the sensitivity of k -effective to reactions by energy group. The sensitivity option in MONK 10 has also been extended to include the sensitivity to the mean number of secondary neutrons per fission, $\bar{\nu}$.

MONK10 also provides tools to combine these sensitivity results with tables of covariance data to estimate better the uncertainties on k -effective arising from the nuclear data.

In addition to these sensitivity analysis tools MONK also provides an option to *adjust* nuclear cross-sections by a user-specified amount both for use in sensitivity studies and for data assimilation. This option has been extended in MONK10 to use BINGO continuous energy data and also to allow the value of $\bar{\nu}$ to be adjusted in addition to cross-section data.

2. Control data

A number of enhancements have been made to the Control Data input unit which is used to control the MONK calculation.

2.1. Birth store dump

There is a new option to dump the contents of the birth store at the start of each stage, allowing the user to interpret the evolution of the fission source in their model during both settling and scoring.

2.2. Geometry units

By default the MONK geometry input is specified in units of centimetres. Optionally the INCHES keyword can be entered to change the default unit to inches. In MONK10 there is a new FACTOR keyword which allows the user to specify any required geometry scaling factor. This aids traceability of the geometry data by allowing the use of whatever units are provided by the original source of the geometry specification.

2.3. Tracking errors

The MONK tracking routines include internal checks to detect samples which do not appear to be in the expected model zone. Typically MONK models do not have any tracking errors unless the user has made a modelling error. However cases which are run for a very long time with a very large number of samples, and which have considerable geometrical complexity can sometimes report these tracking errors. In such cases the sample is abandoned and a replacement sample is started in its place. This ensures that the sample with the error does not contribute to the results with the possibility of introducing a bias, and the replacement sample ensures that tracking errors do not increase the stochastic uncertainty in the results. Current versions of MONK will only tolerate a fixed number of such tracking errors before the run terminates. A new keyword, MAXTRKERR, is provided in MONK10 to allow users to specify how many of these errors to allow before the calculation terminates. This enables models with highly challenging geometrical detail to complete successfully at the small computational expense of the additional samples. The default value is set to 50 in MONK10.

3. Fixed source for driven systems

The highly efficient superhistory powering algorithm in MONK allows the fission source to evolve very rapidly from an initial guess to a source distribution which is representative of the system being modelled. However, for systems with an external, fixed source such as accelerator-driven systems (ADS) a different approach is required. MONK10 therefore includes a new option to enable the user to run a single-stage, fixed source calculation in which the source term and source strength are provided by the user. The results are scaled using the supplied source strength.

Fixed source calculations are limited to systems in which the driven multiplication is less than 0.98. As the multiplication approaches unity the run-time becomes prohibitively longer, and for a multiplication of 1.0 would never complete.

4. Parallelisation

4.1. Background

MONK is increasingly being used not only for criticality calculations but also for reactor physics analyses, including the calculation of distributed quantities such as fluxes and reaction rates in whole reactor core models. This is facilitated by the new scoring capabilities described in section III-1. Achieving acceptable stochastic uncertainties on distributed quantities scored on a fine mesh in a large model requires a very large number of samples, leading to the requirement for parallel processing to support high performance computing (HPC) architectures.

4.2. Parallel algorithm

Monte Carlo codes such as MONK are well-suited to parallel processing as each sample may be treated as independent and tracked by parallel processes, with the quantities tallied during the tracking of each sample gathered and summed at the end of the tracking process. This is complicated by the staged nature of the calculation.

The parallel version of MONK is based on the distributed memory model using the Message Passing Interface (MPI) to handle communication between processes. A master-slave configuration is adopted in which one of the n processes is designated as the master and $n - 1$ are designated as slaves. The number of samples to be tracked in each stage is distributed equally amongst the slaves; the master, which does not track any samples, gathers the data from the slaves and calculates the estimators of k -effective and another quantities based on all of the samples.

During superhistory tracking, any fission neutrons remaining after the final generation are placed in a delay store from which the birth store for the subsequent stage is sampled. Each slave operates its own delay store, which means that no inter-process communication is required within each stage. Therefore at the end of each stage the master must gather the delay stores from each slave to form a master delay store. The master then samples from this delay store to form the birth store for the subsequent stage and broadcasts this birth store to the slaves.

5. Depletion (burn-up) capability

Material burn-up depends on the neutron flux history experienced by the material, which in turn depends on where in a reactor core (or other system) the material is located. To model the spatial-dependence of the burn-up of a given material in MONK it is necessary to specify a unique copy of each material in each region of the model for which unique burn-up is required. This means that the development, maintenance and modification of burn-up models can be very labour-intensive.

For this reason a new mesh-dependent burn-up methodology has been developed for MONK10.⁽⁷⁾ A user-defined Cartesian mesh, known as the burn-up or BU mesh, is overlaid on the underlying model geometry using the UT module described in section II-4.1. In the first cycle of a burn-up calculation a Monte Carlo sampling algorithm (parallelised in the parallel version of MONK) searches for user-defined materials in each cell of this mesh and creates a set of *artificial materials* with a unique mapping between each artificial material and an instance of a user-defined material in a single cell of the mesh. MONK then proceeds with the criticality calculation using the artificial materials and determines the reaction rates for the burn-up calculation in each artificial material. The artificial material mappings are stored in an *archive file* for use in subsequent cycles. The major advantage of this approach is that the spatial discretisation of the burn-up is independent of the underlying model geometry, greatly simplifying the development, maintenance and modification of burn-up models.

A further enhancement to the depletion capability has been the addition of the continuous energy BINGO nuclear data library and collision processor to improve the accuracy of the tracking process compared to the 172-group WIMS nuclear data used for burn-up calculation in MONK9. The increased accuracy of the BINGO method, and the use of the latest JEFF3.1 or ENDF/B-VII libraries, will result in better estimates of both k -effective and the reaction rate data tallied in each cycle of the depletion calculation. The improvements to the reaction rate estimates will provide better data for use in solving the depletion equations and calculate better estimates of the nuclide number densities for the subsequent depletion cycle. MONK10 also features a new solver for the system of ordinary differential equations which is formed by the set of depletion equations.

6. Thermal hydraulics coupling

The burn-up of materials depends on temporally and spatially varying reaction rates in the core. These reaction rates in turn depend on temporally and spatially varying material temperatures and densities. Furthermore the reaction rates depend on material compositions which vary both temporally and spatially as the materials burn up. There is therefore an implicit coupling between thermal hydraulics and core neutronics which is typically ignored in Monte Carlo burn-up calculations. To account for this developments have been carried out to couple MONK to a thermal hydraulics code.⁽⁷⁾

In addition to the overlaid BU mesh, a thermal hydraulics (TH) mesh is superimposed, and the combined (BUTH) mesh is used for the artificial material method described in section III-5 above. MONK calculates heating powers in this mesh which

are used as inputs to a thermal hydraulics model, in order to calculate temperatures and fluid densities. The revised temperatures and densities are applied to the artificial materials in the subsequent time-step of the burn-up calculation.

7. Gamma coupling

It is often assumed that the energy from fission reactions, is deposited in the fuel at the reaction sites. However, typically around 7% of the energy released from fission is associated with prompt and delayed gammas. These gammas can transport energy far from the reaction sites and deposit it in non-fuel materials. MONK does not have the facility to model gamma transport, so developments have been carried out to couple MONK to MCBEND, in order to account for the distribution of energy by gamma transport.⁽⁷⁾

The reaction rates calculated by MONK are used to derive gamma production rates in a standard 22 groups energy scheme for each artificial material. To do this the PHODAT module from WIMS⁽²⁾ has been incorporated into MONK. The derived gamma production rates are used as the gamma source for the shielding code, MCBEND, which tracks the gamma photons through the same model to determine gamma heat deposition rates in the artificial materials. These heating rates are returned to the burn-up calculation in MONK where they are added to the neutron and fission heating rates to determine the total heating rates in the TH mesh for the thermal hydraulics coupling.

8. Validation

The validation of MONK has been expanded from just the DICE library to include all the BINGO libraries currently available. The MONK validation database comprises analyses based on experimental data, mostly taken from the International Criticality Safety Benchmark Evaluation Project (ICSBE), amounting to around 800 configurations. The systems modelled comprise U, Pu and mixed, in the form of metals, solutions and compounds. The purpose of the database is to provide the user with a measure of the accuracy of MONK for the range of systems analysed.

MONK includes a simple categorisation scheme based on six properties that are used to differentiate between system types. The six properties used are: type of fissile nuclide; non-fuel absorption; leakage; resonance absorption; fast fission; and hydrogen fuel content. These properties are tallied during tracking, and are thus derived from the neutronic behaviour of the system. The six properties are combined to give a categorization number, and these are listed in the MONK User Guide⁽⁹⁾ Validation Section for each configuration analysed. The analyst can use the categorisation number from their own calculations with the index supplied in the MONK User Guide to identify quickly comparable systems, from which they can then choose validation cases that most closely match their calculation.

The latest BINGO nuclear data library to be prepared is based on the Chinese CENDL3.2 library, and this is being run through the MONK automated test tool. This test tool automatically runs all the MONK verification and validation test cases, compares the results with a set of expected results, and tabulates the outcome of this process on a series of intranet

web pages. The comparison also includes a range of diagnostic prints that MONK produces to aid the analyst. The automated testing is designed to take the tedious job of submitting calculations and sorting through the numerous output files away from the analyst, who can then concentrate on just the results and diagnostic prints; the original output files are archived in case they are needed by the analyst.

IV. Developments specific to MCBEND11

1. Coupled calculations

Earlier versions of MCBEND execute a coupled neutron-gamma calculation in two steps. In the first step a neutron calculation writes a collision file that contains all the events that could produce secondary gamma photons. In the second step, the collision file is read and processed to derive sources for the gamma calculation. This method has been enhanced and it is now possible for the secondary gamma calculations to be executed by processing up to nine collision files which could be generated in parallel calculations.

MCBEND11 enables coupled neutron-gamma calculations to be executed in a single step for increased ease of use. MCBEND11 also enables the summing of the dose from a neutron response and the dose from a gamma response to give a single result. A biological dose is an example of where this would be valuable.

2. Variance reduction

2.1. Multi-map

An importance map is used in MCBEND to control the splitting/Russian roulette variance reduction process. It is defined as an orthogonal XYZ or $R\theta Z$ mesh that is superimposed on the underlying FG model.

The MULTIMAP feature allows different importance maps to be used in different parts of the FG model.

For example, in the configuration shown in Figure 4 the objective of a MCBEND calculation is to track particles efficiently from inside the drums of active waste, out through the shielding of the drums, across the store and through the walls to the detector region of interest.

Neither of the previously available importance mesh systems are suitable for the whole geometry: an $R\theta Z$ mesh more comfortably fits the drums; an XYZ mesh more comfortably fits the store and its boundary walls.

By allowing multiple importance maps to be specified in a MCBEND calculation the MULTIMAP feature enables the $R\theta Z$ mesh (light grey in Figure 5) in the source drums to be defined independently of the XYZ mesh over the remainder of the problem.

2.2. Forced flight enhancements

Forced flight variance reduction causes deterministic flights to be made from each source or collision point to user-defined surfaces. The method is typically used to direct particles into

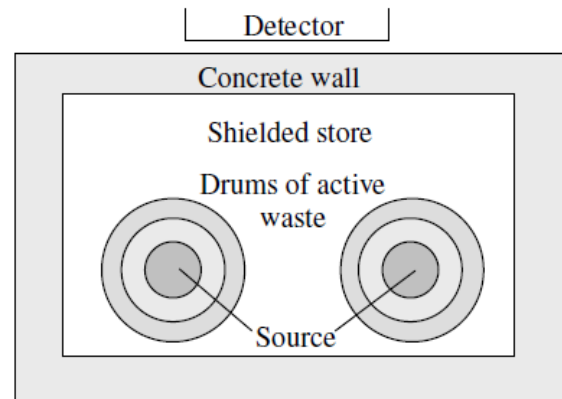


Figure 4: A simple shielding problem: the calculation of radiation transport from sources within shielded drum in a shielded store, to a detector region of interest outside.

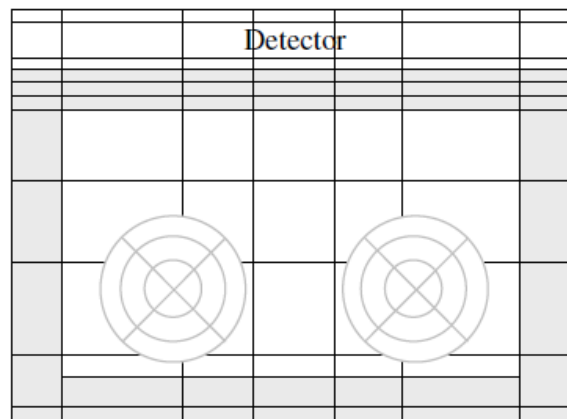


Figure 5: The use of MULTIMAP importance maps to accelerate the problem shown in Figure 4.

the mouth of a duct or collimator. A number of forced flight interface surfaces can be defined.

The forced flight module has been enhanced to allow a sequence of forced flight interfaces to be defined such that collisions after crossing one interface will lead to forced flights towards the next interface in the sequence.

3. Response functions

The response function library used by MCBEND11 allows the user access to a wide range of response functions in a detailed group scheme.

The new response library has been updated to include response functions from the IRDF-2002 dosimetry file, released in 2006. The IRDF-2002 data have been processed using NJOY⁽¹⁰⁾ to produce a detector cross-section library and a detector covariance library for use with MCBEND11. The cross-sections were processed into an ultrafine 13,230 energy group scheme which is fine enough to account for resonance self shielding of single resonance reactions and also gives a good representation of threshold reactions, whilst the covariances were processed into 25 broad groups.

The ICRP-74 neutron and gamma-ray dose rate conversion factors are also available and MCBEND11 can calculate dose rates due to beta radiation.

4. Support for GRID calculations in MCBEND

4.1. Random number generator

In GRID processing the same calculation is run on many processors, with independent random number sequences. This is achieved by advancing (*nudging*) the random number generator by a different (and large) number of steps on each processor.

In MCBEND11 this is effected by using the same seeds, but nudging the random generator on by $n \times 2^{39}$ where n is the processor number. Since only one of the random number generators used by MCBEND is nudged the sequence is independent on each processor even if more than 2^{39} numbers are used.

The results are then combined within MCBEND11 enabling a complete GRID calculation to be easily achieved. Using the same random number sequence in a single calculation allows the user to reproduce the results of a GRID calculation exactly in a single run.

4.2. Merge multiple dump Files

A MCBEND dump file contains a snapshot of the state of a given calculation when the dump is written. It contains the values of all the numbers required for restarting the calculation from that point. MCBEND11 is able to merge dump files in a restart case. The combination of dump files may then be used to print combined results, or to continue the calculation.

V. Visual Workshop

Visual Workshop is a model viewer, editor and results display package for the ANSWERS shielding and criticality codes.

1. Introduction

The long term goal of Visual Workshop is to be an Integrated Development Environment (IDE) for preparing, modifying, checking, running and analysing results from the ANSWERS shielding codes MCBEND and RANKERN, the Monte Carlo reactor physics and criticality code MONK, and the deterministic reactor physics and criticality code WIMS.

Visual Workshop 1 was released in 2010. The current version is Visual Workshop 2 which was released in 2011. Visual Workshop 3 is expected to be released with MONK10.

The key features of Visual Workshop 1 included:

- interactive 2D and 3D ray trace display of the model using the same routines as the analysis code;
- wire frame display of the model;
- built-in editing;
- running and diagnostic capabilities; and
- a simple results display.

The main additional capabilities of Visual Workshop 2 are:

- results displays using 2D cell plots, 2D contour plots and 3D iso-surface plots overlaid on the model geometry; and
- display of event type, location and particle tracks overlaid on the model geometry.

2. Results visualisation

Visual Workshop 2 is able to scan a MONK or MCBEND output file and extract results data which can then be displayed overlaid on the model in the wire frame display.

The following result types are supported in Visual Workshop 2:

- importance map values and particle inflows (MONK and MCBEND);
- flux values in UT meshes (MONK and MCBEND);
- response/dose values in UT meshes (MCBEND); and
- action tally values in UT meshes (MONK).

These results can be plotted as 3D contours or iso-surfaces, 2D contour plane plots or a 2D cell plots. The cell plot displays the results by cell as they are tallied in MONK or MCBEND whereas the contour plots interpolate the contour based on the mid point of a cell.

Figure 6 shows a 2D contour plot of gamma dose contours from a MCBEND calculation of a dry storage facility.

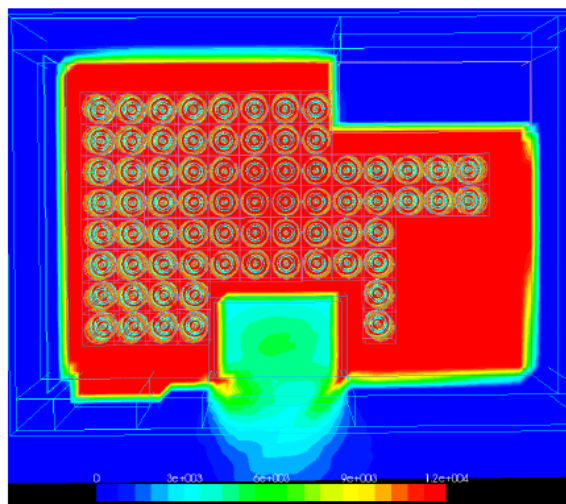


Figure 6: 2D Gamma dose contours from a MCBEND calculation of a dry storage facility.

Figure 7 shows a 3D contour plot of total neutron flux from a simple MONK criticality calculation with fissile material distributed equally in four drums.

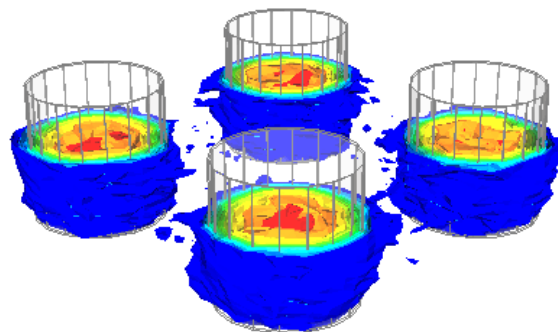


Figure 7: 3D neutron flux contours from a MONK calculation with fissile material in four drums.

3. Event location and track visualisation

MONK and MCBEND can write event locations and particle history information to a range of files. Visual Workshop 2 is able to read these files and display the particle histories on the 3D wire frame, 3D ray trace and 2D ray trace images.

The event files supported by Visual Workshop 2 are:

- history log and collision log data (MONK and MCBEND);
- source log, forced flight log and leakage log data (MCBEND); and
- birth store data (MONK)

The events and tracks can be selectively displayed by sample number, energy, weight, time, location, reaction type and the nuclide or material with which the reaction occurs.

Figure 8 and Figure 9 show examples of particle track display.

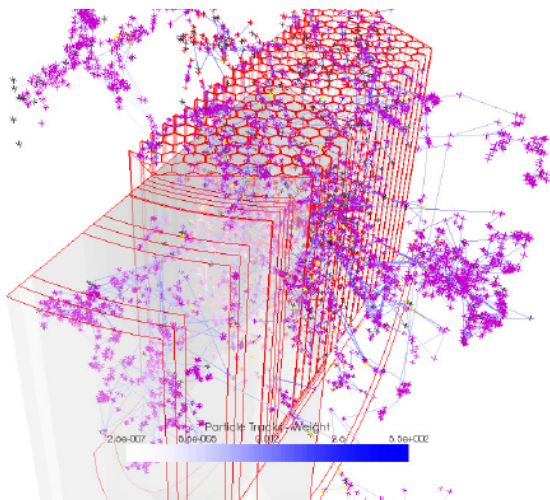


Figure 8: History log on a translucent wire frame display of a reactor with the geometry cut away.

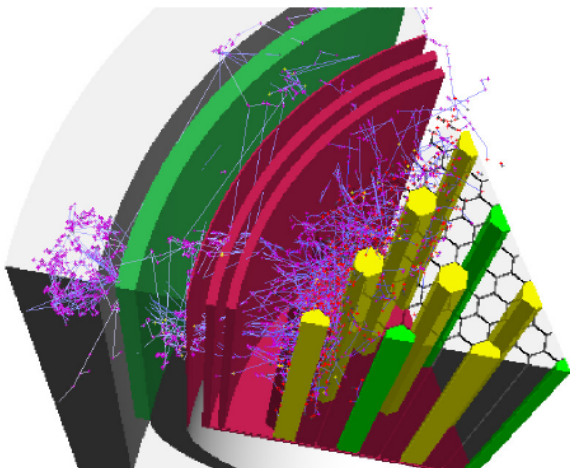


Figure 9: History log on a 3D ray trace with cut away geometry, transparent materials and hidden line removal.

4. Support for development features

Visual Workshop provides ray-traced images of the model using the same particle tracking Fortran code used by MONK and MCBEND. The correct Fortran ray-tracing engine is chosen at run time based on the code version selection. The decoupling of the GUI and the ray-trace functionality allows Visual Workshop to support and test some development features.

The development features include:

- Import of CAD geometry in IGES format: This provides support for complex surfaces with no approximation. An example is shown in Figure 10.
- Import of polygon surface geometries: This provides high performance support for complex surfaces with some geometric approximation, an example of which is shown in Figure 11.

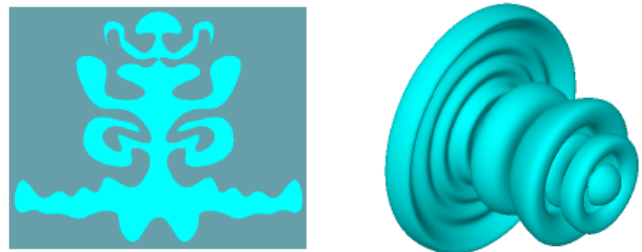


Figure 10: An example of an imported IGES geometry.

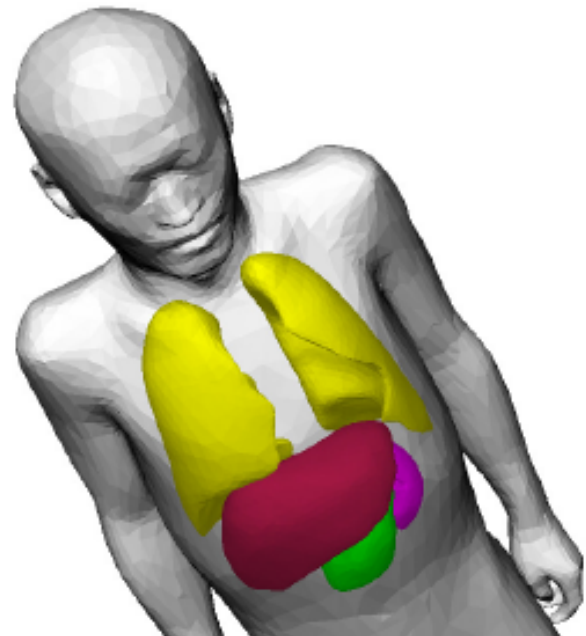


Figure 11: An example of a polygon surface geometry.

5. Wire frame Stereo viewing

Visual Workshop 2 includes functionality to allow the contents of the 3D wireframe view to be displayed in stereo using a compatible combination of graphics card, monitor and stereo

glasses, e.g. the NVIDIA 3D Vision Pro system. This functionality uses OpenGL quad-buffered stereo to display stereo images.

This is currently a development status feature which can be enabled using a command line argument if users with suitable hardware wish to try the functionality.

VI. Conclusions

The MONK Monte Carlo criticality and reactor physics code and the MCBEND Monte Carlo radiation shielding and dosimetry code have each undergone a significant number of developments over the last few years. These are part of the continuing support for MONK and MCBEND as major codes in the ANSWERS Software Service of AMEC. The codes have a varied and powerful geometry specification package, comprising simple bodies, Holes, and now CAD import. They can use a range of nuclear data libraries, and this enables the effect of different evaluations on k -effective and other estimators/tallies to be determined.

The input syntax available to the analyst is also undergoing development to simplify specifying ever more sophisticated models and data: the aim is to not only reduce the user effort in preparing such models, but to reduce the model verification and quality assurance burden by making models easier to check.

The validation database supplied by ANSWERS is continuously reviewed and updated. The aim is to provide the user with the latest data on the accuracy of the MONK and MCBEND codes for a wide range of system types.

The sophisticated visualisation, input editor, results viewer, and job launch tool provide the user with methods for preparing, checking, running and viewing their MONK and MCBEND models and results. The input editor and visualisation options not only ensure that an input is syntactically correct, but also enable the user easily to identify and deal with any geometry data problems. The job launch tool provides a simple interface to enable the user to launch one or more cases, and to monitor the job as it runs. The output display capabilities can be used to monitor the source convergence process in MONK, as well as view tallies accumulated during tracking.

MCBEND11 was released in 2013; MONK10 and Visual Workshop 3 are expected to be released in late 2013 or early 2014.

Acknowledgments

The authors would like to acknowledge the major contributions to the development of MONK made by Malcolm Armishaw prior to his retirement in July 2013.

References

- 1) P. Smith, J. Lillington, and C. Middlemas, "Radiation transport modelling and the ANSWERS codes suite," *Nuclear Future*, **7**, 2, 44–49 (2011).
- 2) ANSWERS Software Service, *WIMS. A Modular Scheme for Neutronics Calculations. User Guide for Version 9*, 2004.
- 3) E. R. Woodcock, T. Murphy, P. J. Hemmings, and T. C. Longworth, "Techniques used in the GEM code for Monte Carlo neutronics calculations in reactors and other systems of complex geometry," *Proc. Conf. on the Application of Computing Methods to Reactor Problems*, ANL-7050, p. 557–579, 1965.
- 4) R. J. Brissenden and A. R. Garlic, "Biases in the Estimation of k_{eff} and its Error by Monte Carlo Methods," *Ann. Nucl. Energy*, **13**, 2, 63–83 (1986).
- 5) A. Bird and A. Kyreleis, "Experience Using Models Imported From CAD Software for Shielding Calculations In MCBEND," *Proc. 12th Int. Conf. on Radiation Shielding (ICRS-12) and 17th Topical Meeting on Radiation Protection and Shielding (RPSD-2012)*, Nara, Japan, September, 2012.
- 6) C. J. Dean *et al.*, "Validation of Run-time Doppler Broadening in MONK with JEFF3.1," *Proc. International Conference on Nuclear Data for Science and Technology*, Korea, 2010.
- 7) S. D. Richards, N. Davies, M. J. Armishaw, G. P. Dobson, and G. A. Wright, "Parallelisation of MONK with Coupling to Thermal Hydraulics and Gamma Heating Calculations for Reactor Physics Applications," *Proc. this conference*, Paris, France, October, 2013.
- 8) F. B. Brown, "Convergence testing and acceleration in MCNP5 criticality calculations," *Proc. 8th International Conference on Nuclear Criticality Safety*, p. 356–359, St. Petersburg, Russia, 2007.
- 9) ANSWERS Software Service, *MONK. A Monte Carlo Program for Nuclear Criticality Safety and Reactor Physics Analyses. User Guide for Version 9*, 2006.
- 10) R. E. MacFarlane and D. W. Muir, "NJOY99.0 Code System for Producing Pointwise and Multigroup Neutron and Photon Cross Sections from ENDF/B Data," PSR-480/NJOY99.0, Los Alamos National Laboratory (2000).