ELSEVIER

Journal of Nuclear Materials 280 (2000) 127–135

journal of nuclear materials

www.elsevier.nl/locate/jnucmat

Numerical algorithms for intragranular fission gas release

K. Lassmann [a,*], H. Benk [b]

[a] European Commission Joint Research Centre, Institute for Transuranium Elements, P.O. Box 2340, D-76125 Karlsruhe, Germany
[b] Technical University Darmstadt, FG Energietechnik und Reaktoranlagen, Petersenstraße 30, D-64287 Darmstadt, Germany

Received 22 November 1999; accepted 24 March 2000

Abstract

Complicated physical processes govern fission gas release in nuclear fuels. Besides the physical problem, there is a numerical problem since some solutions of the underlying diffusion equation have numerical errors that by far exceed the physical details. In this paper, the efficiency and the accuracy of some numerical solutions are analysed. Random operation histories were generated and the errors inherent in each algorithm evaluated over a wide range of up- and down-ramps by comparing the results with the quasi-exact ANS-5.4 algorithm. The URGAS algorithm can be considered as well balanced over the entire range of fission gas release. The new FORMAS algorithm is superior at fission gas release above $f \approx 0.05$ and may in a physical sense be considered as an exact solution in this range. Unfortunately, the deficiency of this most elegant and mathematically sound algorithm at low fission gas release could not be fully overcome. However, in view of the many inherent uncertainties, both algorithms are considered as sufficient to be used in a fuel rod performance code. All algorithms analysed in detail can be made available on request as FORTRAN subroutines.

1. Introduction

Olander [1] introduces his Chapter 13 on *Swelling Due to Fission Gases* by saying:

> Among the myriad phenomena that occur simultaneously in a nuclear fuel element under irradiation, none has so frustrated the designer, so challenged the experimentalist, or so intrigued the theorist as the behaviour of the fission products xenon and krypton.

Although written in 1976 this statement is still valid, and even after more than 30 years of research, fission gas release and swelling is still a subject of controversial discussions. Nevertheless, this area needs to be covered in fuel rod analyses because:

1. The fission gases xenon and krypton degrade the thermal conductivity of the gas (normally helium) inside the pin, i.e., decrease the gap conductance and thus enhance fuel temperatures. Enhanced fuel temperatures may further increase fission gas release and may initiate an unstable process called 'thermal feedback'.
2. The release of fission gases increases the inner pin pressure. This pressure increase may limit the lifetime of a fuel rod since the inner pressure should not exceed the coolant pressure (simplified design limit).
3. The swelling due to gaseous fission products may lead to enhanced pellet–cladding mechanical interaction (PCMI), especially in transients.
4. The release of radioactive gases (and of volatile solids) from UO_2 to the free volume decreases the safety margin of a nuclear plant. In that sense, the nuclear fuel UO_2 may be considered as the first barrier against the release of radioactivity.

Various isotopes of the fission gases xenon and krypton are directly created inside the grains by fission, but may also originate from decay processes. Since their solubility in UO_2 is very low, they diffuse inside the grains (diffusion process) or precipitate into intra- and inter-granular bubbles. Finally, they may reach the pin-free volume (gaps inside or between pellets, gaps between fuel and cladding, plenum volume) basically by inter-linkage of inter-granular bubbles and subsequent venting of the grain boundary inventory. The dominant process is thermal- or irradiation-induced diffusion of

* Corresponding author.

0022-3115/00/$ - see front matter
PII: S0022-3115(00)00044-1

single gas atoms. In the wake of fission spikes, small intragranular bubbles with a typical diameter of 1–2 nm are formed which grow by diffusion, but may also be destroyed fully or partially by fission spikes. There is general agreement that the intragranular bubbles are immobile under normal operational conditions and migrate only at very high temperatures (>2000°C). Thus, gas atoms in these bubbles may be trapped or resolved in the matrix (irradiation-induced resolution). Since trapped gas atoms cannot contribute to diffusion, the apparent diffusion is somewhat reduced, which is considered by an 'effective' diffusion coefficient. The concept of an effective diffusion coefficient was first derived by Speight [2] and is used in many fission gas release models.

If trapping and resolution is in equilibrium, the effective diffusion coefficient, D_{eff}, may be expressed as

$$D_{\text{eff}} = \frac{Db}{b+g}, \tag{1}$$

where D is the gas atom diffusion coefficient, b the probability per unit time of a gas atom in bubble being redissolved and g is the probability per unit time of gas atoms in solution being captured by a bubble. However, this formulation has a tendency to underpredict fission gas release during irradiation at high temperatures and during out-of-pile annealing tests [3]. Several attempts have been made to overcome this problem, for instance by considering a 'thermal resolution' that reduces the trapping effect at high temperatures [4], but one may also consider D_{eff} as a parameter, typically of the form

$$D_{\text{eff}} = a' \mathrm{e}^{-b/T}. \tag{2}$$

In all subsequent investigations we have used this form with $a' = 5 \times 10^{-8}$ m^2 s^{-1} and $b = 40262$ K, following a recommendation of Matzke [5].

The simplest model for diffusion of gas inside a spherical grain (intra-granular diffusion) may be written as

$$\frac{\partial c}{\partial t} = D_{\text{eff}}\left(\frac{\partial^2 c}{\partial r^2} + \frac{2}{r}\frac{\partial c}{\partial r}\right) + \beta, \tag{3}$$

where $c = c(r,t)$ is the local concentration of gas in solution and bubbles, $\beta = \beta(t)$ the rate at which gas is produced, $D_{\text{eff}} = D_{\text{eff}}(t)$ the 'effective' diffusion coefficient, t the time, and r is the co-ordinate of the spherical grain.

Unfortunately, the solution of the simple equation (3) is difficult to obtain under (varying) reactor conditions and numerical errors may by far exceed the physical details. Therefore, the numerical aspects of various solutions are re-investigated.

2. Solutions for constant conditions (constant D_{eff} and β)

For an arbitrary initial concentration $c(r, t=0) = c_0(r)$ at time $t = 0$ and the boundary condition $c(r = a, t) = 0$ the exact solution of Eq. (3) is [6]

$$\begin{aligned} c(r,t) = {} & \frac{2}{ar}\sum_{k=1}^{\infty} \exp\left(-\frac{D_{\text{eff}}k^2\pi^2 t}{a^2}\right)\sin\frac{k\pi r}{a} \\ & \times \int_0^a r' c_0(r') \sin\frac{k\pi r'}{a}\,\mathrm{d}r' \\ & + \frac{2\beta a^3}{D\pi^3 r}\sum_{k=1}^{\infty}\frac{(-1)^k}{k^3}\exp\left(-\frac{D_{\text{eff}}k^2\pi^2 t}{a^2}\right)\sin\frac{k\pi r}{a} \\ & + \frac{\beta}{6D_{\text{eff}}}(a^2 - r^2). \end{aligned} \tag{4}$$

If the initial concentration is constant, i.e. $c_0(r) = \bar{c}_0$, Eq. (4) can be integrated to give the average concentration in the grain $\bar{c}(t)$ as a function of time

$$\begin{aligned} \bar{c}(t) = {} & \frac{6\bar{c}_0}{\pi^2}\sum_{k=1}^{\infty}\frac{1}{k^2}\exp\left(-\frac{k^2\pi^2 D_{\text{eff}}t}{a^2}\right) \\ & + \bar{c}_\infty\left\{1 - \frac{90}{\pi^4}\sum_{k=1}^{\infty}\frac{1}{k^4}\exp\left(-\frac{k^2\pi^2 D_{\text{eff}}t}{a^2}\right)\right\}, \end{aligned} \tag{5}$$

where $\bar{c}_\infty = (\beta a^2)/(15 D_{\text{eff}})$.

Fission gas release from the interior of the grain to the grain boundary, i.e., the intra-granular fission gas release is defined as

$$f(t) = \frac{\bar{c}_{\text{created}}(t) - \bar{c}(t)}{\bar{c}_{\text{created}}(t)}, \tag{6}$$

where $\bar{c}_{\text{created}}(t) = \beta t$. For $\bar{c}_0 = 0$ we obtain from Eq. (5)

$$f(t) = 1 - \frac{a^2}{15 D_{\text{eff}} t}\left\{1 - \frac{90}{\pi^4}\sum_{k=1}^{\infty}\frac{1}{k^4}\exp\left(-\frac{k^2\pi^2 D_{\text{eff}}t}{a^2}\right)\right\}, \tag{7}$$

which can be approximated with a relative accuracy better than 10^{-3} by

$$f(t) \approx 4\sqrt{\frac{D_{\text{eff}}t}{\pi a^2}} - \frac{3}{2}\frac{D_{\text{eff}}t}{a^2} \quad \text{for } f < 0.75. \tag{8}$$

A post-irradiation annealing solution assumes a uniform concentration $\bar{c}_0$ at time $t = 0$ and $\beta = 0$ for all times $t \geqslant 0$. This gives

$$f(t) = 1 - \frac{6}{\pi^2}\sum_{k=1}^{\infty}\frac{1}{k^2}\exp\left(-\frac{k^2\pi^2 D_{\text{eff}}t}{a^2}\right), \tag{9}$$

which can be approximated with a relative accuracy better than 10^{-3} by

$$f(t) \approx 6\sqrt{\frac{D_{\text{eff}}t}{\pi a^2}} - 3\frac{D_{\text{eff}}t}{a^2} \quad \text{for } f < 0.9. \tag{10}$$

The approximations given above were first derived by Booth [7].

3. Solutions for varying conditions, $D_{\text{eff}} = D_{\text{eff}}(t)$ and $\beta = \beta(t)$

During the irradiation the linear rating varies. But even if the linear rating were constant, the local temperature would vary, for instance due to pellet cracking, densification, swelling or cladding creep down. Thus, this is the normal case to be treated in any fuel rod performance code. The problem is that any particular power history leads to a specific, individual gas distribution inside the grain. This individual and time dependent distribution needs to be known in order to calculate the fission gas release to the grain boundaries during the time step $\Delta t = t_{n+1} - t_n$. The more information from the previous distribution is stored, the more accurate will be the new result.

Various numerical solutions have been proposed and we have analysed most of them. The solutions of Ronchi and co-workers [8,9], Rashid et al. [10] and the method of Arimescu [11] were not analysed. However, as will be seen below, we propose two algorithms that fulfill all requirements for use in a fuel performance code and need little computation time.

Common to most numerical solutions is the problem of dealing with a situation where fission gas has accumulated in the grain at a rather low temperature (very low diffusion coefficient) followed by a sudden increase of temperature (high diffusion coefficient). In order to study the numerical behaviour of a specific method in such a situation and in all other possible situations, randomly generated, arbitrary power histories were constructed that cover the entire range of situations including a wide range of up- and down-ramps. As a result, several thousands of numerical experiments were performed and analysed.

Three methods will be discussed in more detail:

1. The quasi-exact ANS-5.4 algorithm, which is used in this work as a reference.
2. The URGAS algorithm derived using some assumptions from the steady-state solution (5).
3. The algorithm of Forsberg and Massih (labelled FORMAS), which is the most elegant mathematical approach.

3.1. The quasi-exact ANS-5.4 algorithm

An analytical solution has been proposed by Rim [12], which is used in the ANS-5.4 algorithm [13]. This algorithm is only affected by errors due to discretisation of a real history into piecewise constant conditions. For such piecewise constant conditions the ANS-5.4 algorithm is exact, and will therefore serve in the following investigation as a reference. Since for arbitrary histories it exhibits a discretisation error, it is labelled 'quasi-exact' algorithm. The ANS-5.4 algorithm is optimum for applications where there are no tight constraints on computer storage or time. Its main drawback is that the numerical effort increases with the number of time steps so that this algorithm is calculationally too expensive for the full simulation of real fuel-pin operating histories within a fuel rod modelling code. The ANS-5.4 algorithm is programmed in the FORTRAN subroutine ANS54, which is available on request.

3.2. The URGAS algorithm

In the TRANSURANUS code [14], an algorithm called URGAS is used which has been improved considerably since its first publication [15]. The actual URGAS algorithm is based on Eq. (5).

It is assumed that the average gas concentration $\overline{c}_n$ at time t_n derives from a fictitious irradiation during the time $t_{\text{fic},n}$ with constant conditions corresponding to those during the time step $\Delta t^{(i)}$ under consideration. The gas production rate $\beta^{(i)}$ and the diffusion coefficient $D_{\text{eff}}^{(i)}$ are those of the actual time step $\Delta t^{(i)}$. The fictitious time $t_{\text{fic},n}$ can be determined by solving the equation

$$\overline{c}_n = \frac{6\overline{c}_0}{\pi^2}\sum_{k=1}^{\infty}\frac{1}{k^2}\exp\left(-\frac{k^2\pi^2 D_{\text{eff}}^{(i)} t_{\text{fic},n}}{a^2}\right) + \frac{\beta^{(i)}a^2}{15D_{\text{eff}}^{(i)}}\left\{1 - \frac{90}{\pi^4}\sum_{k=1}^{\infty}\frac{1}{k^4}\exp\left(-\frac{k^2\pi^2 D_{\text{eff}}^{(i)} t_{\text{fic},n}}{a^2}\right)\right\}, \tag{11}$$

which is written as

$$\overline{c}_n = \overline{c}_0 f_{\text{a}} + \overline{c}_{\infty} f_{\text{d}}. \tag{12}$$

The factors f_{a} and f_{d} stand for 'annealing' and 'diffusion'. The physical meaning of the constant $\overline{c}_0$ is the same as in Eq. (5). However, $\overline{c}_0$ is considered as an unknown for which the assumption

$$\overline{c}_0 = \overline{c}_{\text{created},n} - \beta^{(i)} t_{\text{fic},n} \quad \text{with } \overline{c}_0 \geqslant 0 \tag{13}$$

is made. The justification for this assumption is the following: $\overline{c}_{\text{created},n}$ is the gas which was created up to time t_n, the beginning of the time step. According to the assumptions made, the amount of gas $\beta^{(i)} t_{\text{fic},n}$ is produced during the fictitious irradiation. If the gas, which has already been created, $\overline{c}_{\text{created},n}$, is larger than $\beta^{(i)} t_{\text{fic},n}$, the difference is considered in Eq. (11) as 'initial condition'. The assumption (13) is due to White [4,16] and improves significantly the assumption of the previous URGAS algorithm.

The factor f_{a} is approximated by

$$f_{\text{a}}(t') \approx 1 + 0.2t' - 6\sqrt{\frac{t'}{15\pi}} \quad \text{if } t' \geqslant 1.16758, \tag{14}$$

$$f_a(t') \approx \frac{6}{\pi^2}\sum_{k=1}^{3}\frac{1}{k^2}e^{-k^2\pi^2 t'/15} \quad \text{if } t' > 1.16758, \tag{15}$$

where $t' = t/\tau$ and $\tau = a^2/15D_{\text{eff}}$. The relative error with respect to the exact solution is less than 6×10^{-7}.

The factor f_d is approximated by

$$f_d(t') \approx t' + 0.1t'^2 - 4\sqrt{\frac{t'^3}{15\pi}} \quad \text{if } t' \leqslant 1.16758, \tag{16}$$

$$f_d(t') \approx 1 - \frac{90}{\pi^4}\sum_{k=1}^{3}\frac{1}{k^4}e^{-k^2\pi^2 t'/15} \quad \text{if } t' > 1.16758. \tag{17}$$

The relative error with respect to the exact solution is less than 6×10^{-7}.

Applying the approximations given above, Eq. (11) is solved for the unknown $t_{\text{fic},n}$ by a Newton method. Special care is taken in the case where both branches of the approximations are taken in the course of the iteration.

The average concentration at the end of the time step is

$$\overline{c}_{n+1} = \overline{c}_0 f_a(t'_{\text{fic},n+1}) - \overline{c}_\infty f_d(t'_{\text{fic},n+1}), \tag{18}$$

where

$$t'_{\text{fic},n+1} = t'_{\text{fic},n} + \frac{\Delta t}{\tau}, \tag{19}$$

$$f_{n+1} = \frac{\overline{c}_{\text{created},n+1} - \overline{c}_{n+1}}{\overline{c}_{\text{created},n+1}}. \tag{20}$$

The URGAS algorithm is programmed in the FORTRAN subroutine URGAS, which is available on request.

3.3. The FORMAS algorithm

This method was developed by Forsberg and Massih [17,18] and is therefore labelled FORMAS algorithm. The diffusion equation (3) is transformed to

$$\frac{\partial c}{\partial T} = \left(\frac{\partial^2 c}{\partial r^2} + \frac{2}{r}\frac{\partial c}{\partial r}\right) + P(T), \tag{21}$$

where $c = c(r,T)$, $T = \int_0^t D_{\text{eff}}(t)\,dt$ and $P(T) = \beta(T)/D(T)$.

The average gas concentration $\overline{c}(T)$ in the grain is obtained by

$$\overline{c}(T) = \frac{3}{4\pi a^3}\int_0^T K(T-s)P(s)\,ds, \tag{22}$$

where the kernel K is given by

$$K(T) = \frac{8a^3}{\pi}\sum_{k=1}^{\infty}e^{-k^2\pi^2 T/a^2}/k^2. \tag{23}$$

The kernel K is approximated by

$$K(T) \approx a^3\sum_{i=1}^{m}A_i e^{-B_i T/a^2} \tag{24}$$

and Eq. (22) can be written as

$$\overline{c}(T) \approx \frac{3}{4\pi}\sum_{i=1}^{m}c_i = \sum_{i=1}^{m}\int_0^T A_i \exp\left(-\frac{B_i(T-s)}{a^2}\right)P(s)\,ds. \tag{25}$$

The integration in Eq. (25) is performed under the assumption that the quantity P varies linearly in the time step $T_n \leqslant T < T_{n+1}$. With $m = 3$ the original FORMAS algorithm of Forsberg and Massih results, which is in contrast to the ANS-5.4 solution, an incremental (recursive) algorithm. The authors used

$$\{A\} = \begin{Bmatrix} 2.6391 \\ 0.865 \\ 0.6189 \end{Bmatrix} \quad \text{and} \quad \{B\} = \begin{Bmatrix} 9.9905 \\ 64.488 \\ 511.61 \end{Bmatrix}. \tag{26}$$

It is obvious that the approximation (24) with only three terms has limitations for small x since

$$K'(x) = \frac{dK(x)}{dx} \to -\infty \quad \text{for } x \to 0. \tag{27}$$

In other words, one would expect that the original FORMAS algorithm is inaccurate at low fission gas release. As will be shown in more detail below, this is indeed the case for $f < 0.05$. In order to overcome this principle deficiency, we have tried to introduce the much better approximation

$$K(x) \approx \frac{4}{3}\pi\left(1 - 6\sqrt{\frac{x}{\pi}} + 3x\right), \tag{28}$$

which approximates Eq. (23) in the range of $0 \leqslant x \leqslant 0.1$ with a relative error $<10^{-6}$. Unfortunately, with this specific analytic form, the integration in Eq. (25) does not lead to an incremental algorithm [19].

Numerical tests performed with the original FORMAS algorithm showed:

1. Numerical difficulties, mainly at the beginning of a sudden increase of temperature leading to a high sensitivity of the time-step length.
2. A systematic overprediction at low fission gas release.

In order to overcome the first problem, the algorithm was slightly modified. We found a much better performance by assuming that $P(s) = \overline{P} = \text{constant}$ in the interval $T_n \leqslant T < T_{n+1}$. With this assumption the incremental algorithm reads

$$c_i(T_{n+1}) = e^{-B_i\Delta T/a^2}\left\{c_i(T_n) - \frac{A_i\overline{P}'}{B_i}\right\} + \frac{A_i\overline{P}'}{B_i}, \tag{29}$$

where $\overline{P}' = \overline{P}^* a^2$.

In order to overcome the problem of the systematic overprediction at low fission gas release, numerous numerical investigations were performed:

1. The number m in Eq. (24) was varied from 1 to 5.
2. The coefficients A_i and B_i were fitted to Eq. (23) using a non-linear least-squares method; in addition to the standard fitting, a modification was used where $\sum_{i=1}^{m} A_i = \frac{4}{3}\pi$.
3. The fitting range $0 \leqslant x < x_{\max}$ was systematically varied.

From this work the following conclusions were obtained:

1. A reasonable compromise between accuracy and computational effort is $m = 4$.
2. The coefficients A_i and B_i depend strongly on the fitting range. The fitting range $0 \leqslant x < 0.1$ is a reasonable compromise for the whole range of application.
3. As will be shown in Chapter 4, the accuracy is still poor for fission gas release below $f = 0.05$. Therefore, an empirical correction has been developed to overcome this problem.

The algorithm based on Eq. (29) with newly fitted coefficients A_i and B_i, $i = 1,4$ and an empirical correction for fission gas release below $f = 0.05$ is labelled subsequently as new FORMAS algorithm. The coefficients are

$$\{A\} = \begin{Bmatrix} 2.66489114482 \\ 0.802561417596 \\ 0.403753353935 \\ 0.247275609033 \end{Bmatrix} \quad \text{and} \quad \{B\} = \begin{Bmatrix} 10.2378274585 \\ 56.0048233851 \\ 256.157980353 \\ 2018.61920274 \end{Bmatrix}. \tag{30}$$

The empirical correction is constructed in such way that for constant conditions the exact solution is obtained. After the calculation of the fission gas release $f(t)$ according to Eq. (6), the correction reads

$$\begin{aligned} f_{\text{corr}} &= 0.255\sqrt{x'} + 3.87^{*}x'^{2}, \\ x' &= f(t) - 1.68 \times 10^{-2}, \\ \overline{c}_{n+1} &= \overline{c}_{\text{created},n+1}(1 - f_{\text{corr}}). \end{aligned} \tag{31}$$

The empirical correction (31) is applied if $f(t) < 0.05$. Some proof for its correctness is given in Fig. 5.

The new FORMAS algorithm is programmed in the FORTRAN subroutine FORMAS, which is available on request.

3.4. The MacDonald–Weisman algorithm

Setting $m = 1$, $A_1 = \frac{4}{3}\pi$ and $B_1 = 15$ in Eqs. (24) and (29) gives the very simple incremental algorithm derived in Ref. [15] in a completely different way

$$\overline{c}_{n+1} = (\overline{c}_n + \overline{c}_\infty)\mathrm{e}^{-\Delta t^{*}} + \overline{c}_\infty, \tag{32}$$

where $\Delta t^{*} = \Delta t/\tau$, $\overline{c}_\infty = \beta\tau$ and $\tau = a^2/15D$.

This simplified algorithm was developed for high fission gas release and it exhibits a systematic underestimation of fission gas release up to $f \approx 0.8$. Nevertheless, it is basically the MacDonald–Weisman fission gas release model that has been used in the MATPRO series of models [20] and in the FRAPCON code [21]. The intragranular part of the MacDonald–Weisman model is given by

$$\overline{c}_{n+1} = \left\{ \overline{c}_n - \frac{\beta(1-k')}{k'k''} \right\} \mathrm{e}^{-k'k''\Delta t} + \frac{\beta(1-k')}{k'k''}, \tag{33}$$

where k' is the fraction of gas which moves directly from the grain to the grain boundary ($k' \ll 1$) and k'' is the probability per unit time of release of fission gases from gas trapped in intragranular bubbles. Setting $\tau = 1/k'k''$ gives Eq. (32). In order to overcome the systematic underestimation of fission gas release of the algorithm, the authors increased the effective diffusion coefficient by nearly two orders of magnitude compared with 'normal' coefficients. Clearly, in such a case one should speak of a fitting constant instead of an effective diffusion coefficient with a distinct physical meaning.

4. Numerical experiments

Random operation histories were generated and the errors inherent in each algorithm evaluated over a wide range of up- and down-ramps by comparing the results with the quasi-exact ANS-5.4 algorithm. The random histories were constructed according to the following principles:

1. Each individual power history was constructed by piecewise constant conditions in which temperature and fission gas production rates are constant. Such a history has the advantage that the quasi-exact ANS-5.4 algorithm is not affected by discretisation errors.
2. In each individual power history, the following quantities were considered as random variables:
   (a) the number of constant periods (2–10),
   (b) the time length of a constant period,
   (c) the temperature of a constant period (between 800 and 1800 K),
   (d) the fission gas production rate (corresponding to a linear rating between 10 and 40 kW/m).
3. 2000 individual, randomly generated power histories were used.

These principles ensure that all possible situations are covered. This can be easily seen in the subsequent figures since all individual calculational results are uniformly distributed over the whole range of fission gas release

($0 \leqslant f \leqslant 1$). Fig. 1 gives as an example the evaluation of the algorithm of Väth [22]. Each point in this figure is the result of a randomly generated history and shows the fission gas release as predicted by the algorithm of Väth versus the quasi-exact fission gas release. The closer the results are to the 45° diagonal, the better the algorithm is. As can be seen from Fig. 1, the spread is rather high. However, it must be noted that this algorithm was developed for application in fast breeder reactors where the fission gas release is usually very high.

Fig. 2 shows the characteristics of the simple algorithm (32). For $f < 0.8$ this algorithm gives a systematic underestimation. Reasonable results are obtained only above $f = 0.8$. Although this simple algorithm can be tailored for a certain range of applicability by using different parameter A_1 and B_1, it should never be used for the full range of fission gas release.

The URGAS algorithm is analysed in Fig. 3 ($0 \leqslant f \leqslant 0.1$) and Fig. 4 ($0 \leqslant f \leqslant 1$). Although there are some variations around the 45° line, there is no systematic under or overprediction and URGAS must be considered as a well-balanced algorithm. It should be noted that for constant conditions URGAS gives the exact values over the entire range.

The FORMAS algorithm is first analysed for constant conditions. Fig. 5 shows that in the low fission gas release range, the original and the new FORMAS algorithm without the empirical correction give very unacceptable systematic deviations, which indicates the relevance of the empirical correction. This can also be seen in Fig. 6 where the difference between the uncorrected and the corrected new FORMAS algorithm is shown for varying conditions. Both solutions have a tendency to overpredict fission gas release. The new FORMAS algorithm which includes the empirical

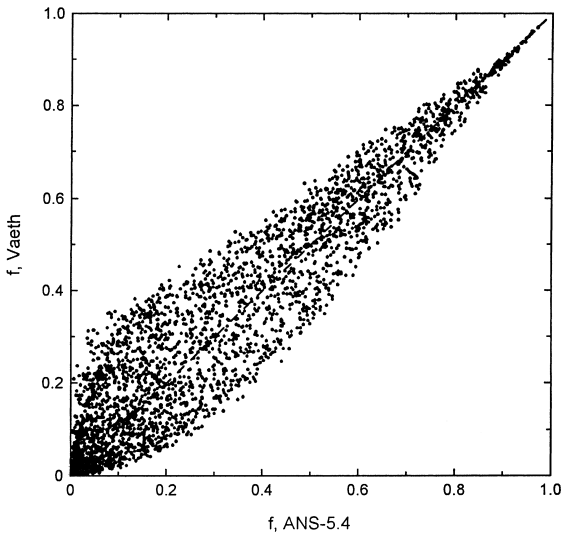

Fig. 1. Comparison between the fission gas release f of the algorithm of Väth [18] with the quasi-exact ANS-5.4 algorithm. Each point is the result of a calculation with randomly generated conditions. The deviation from the 45° line is a measure of the accuracy.

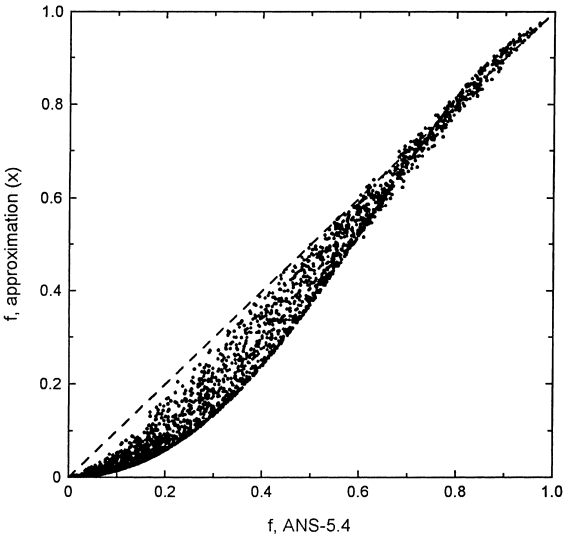

Fig. 2. Comparison between the fission gas release f of the approximate algorithm (32) with the quasi-exact ANS-5.4 algorithm. Each point is the result of a calculation with randomly generated conditions. The deviation from the 45° line is a measure of the accuracy.

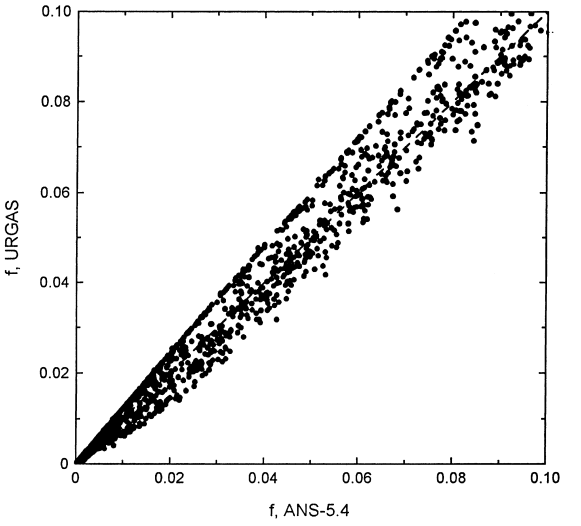

Fig. 3. Comparison between the fission gas release f of the URGAS algorithm with the quasi-exact ANS-5.4 algorithm. Each point is the result of a calculation with randomly generated varying conditions. The deviation from the 45° line is a measure of the accuracy. Note that only the range $0 \leqslant f < 0.1$ is shown.

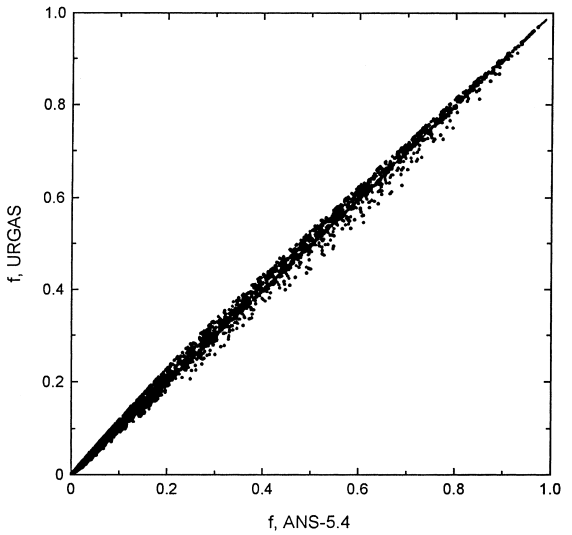

Fig. 4. Comparison between the fission gas release f of the URGAS algorithm with the quasi-exact ANS-5.4 algorithm. Each point is the result of a steady-state calculation with randomly generated conditions. The deviation from the 45° line is a measure of the accuracy.

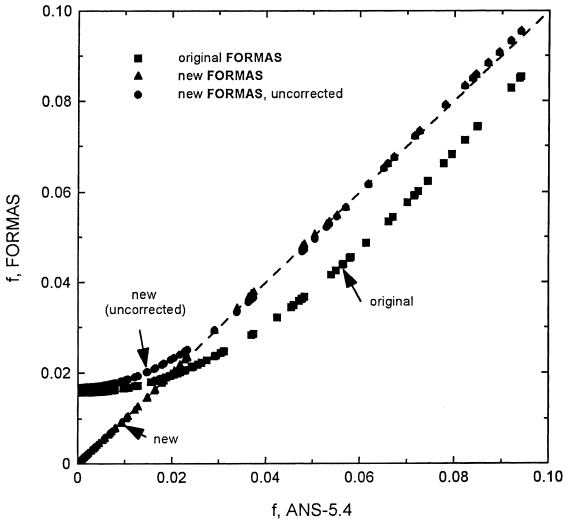

Fig. 5. Comparison between the fission gas release f of different versions of the FORMAS algorithm with the quasi-exact ANS-5.4 algorithm. Each point is the result of a steady-state calculation with randomly generated conditions. The deviation from the 45° line is a measure of the accuracy. Note that only the range $0 \leqslant f < 0.1$ is shown.

correction is analysed in Figs. 7 ($0 \leqslant f \leqslant 0.1$) and 8 ($0 \leqslant f \leqslant 1$). These figures are to be compared with Figs. 3 and 4 for the URGAS algorithm.

Several unsuccessful attempts have been made to find a more suitable correction for varying conditions. Obviously, there is no mathematical approach other than the correction applied, i.e., a correction that gives an excellent agreement in steady-state situations. What remains to be found is a correction based on physical arguments and this leads straightforward to the use of the URGAS algorithm at low fission gas release.

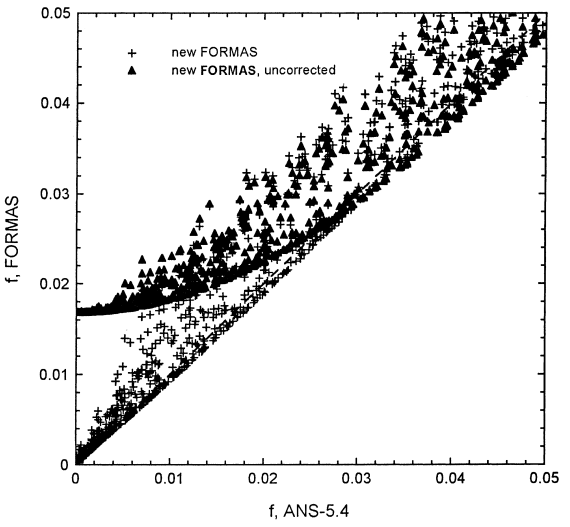

Fig. 6. Comparison between the fission gas release f of the corrected and uncorrected new FORMAS algorithm with the quasi-exact ANS-5.4 algorithm. Each point is the result of a calculation with randomly generated varying conditions. The deviation from the 45° line is a measure of the accuracy.

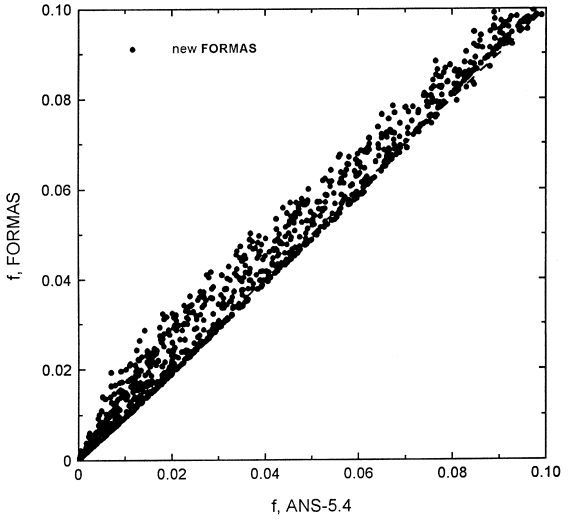

Fig. 7. Comparison between the fission gas release f of the new FORMAS algorithm with the quasi-exact ANS-5.4 algorithm. Each point is the result of a calculation with randomly generated varying conditions. The deviation from the 45° line is a measure of the accuracy. Note that only the range $0 \leqslant f < 0.1$ is shown.

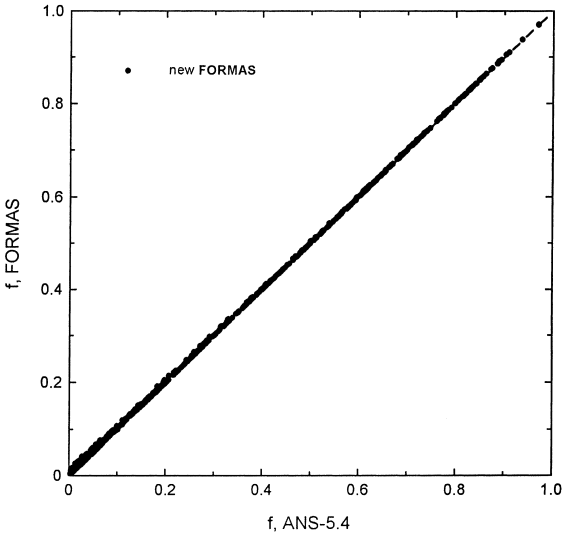

Fig. 8. Comparison between the fission gas release f of the new FORMAS algorithm with the quasi-exact ANS-5.4 algorithm. Each point is the result of a calculation with randomly generated varying conditions. The deviation from the 45° line is a measure of the accuracy.

5. Evaluation of the different algorithms

From what has been said above it can be stated that the new FORMAS algorithm and the URGAS algorithm are potentially suitable algorithms. The URGAS algorithm is based on reasonable physical considerations and shows a well-balanced overall behaviour, whereas the new FORMAS algorithm is based on a sound mathematical approach. Therefore, both algorithms are analysed in more detail. Fig. 9 shows the relative errors of the URGAS and the new FORMAS algorithm as a function of the fission gas release $0 \leqslant f < 0.25$. Clearly, the URGAS algorithm is superior in the range $f < 0.05$, whereas the new FORMAS algorithm may be considered as an almost exact solution above $f = 0.1$. Even with an empirical correction for low fission gas release, the new FORMAS algorithm gives: (a) a systematic overprediction and (b) rather high relative errors in this range. It is interesting to note that the algorithm of Matthews and Wood [23] which is very similar to the original FORMAS algorithm (see the corresponding discussion in Ref. [17]) showed a very similar behaviour. It was also found that at low fission gas release a correction is needed [24]. However, the soundness of the correction was shown only for a few examples.

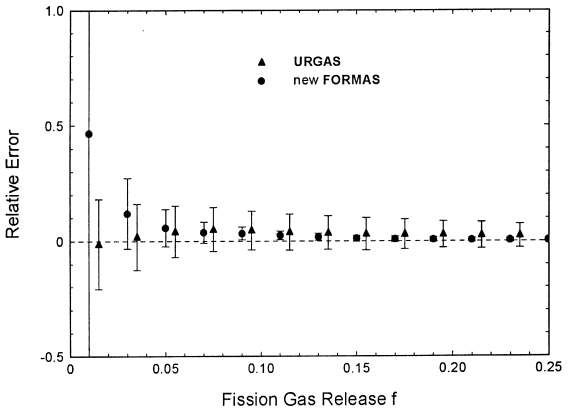

Fig. 9. Relative errors of the URGAS algorithm and the new FORMAS algorithm as a function of the fission gas release $0 \leqslant f < 0.25$.

6. Conclusions

The URGAS algorithm can be considered as well balanced over the entire range of fission gas release. The new FORMAS algorithm is superior at fission gas release above $f \approx 0.05$ and may, in a physical sense, be considered as an exact solution in this range. Unfortunately, the deficiency of this most elegant and mathematically sound algorithm at low fission gas release could not be fully overcome. The reason for this behaviour originates from Eq. (27). In a situation where fission gas has been collected at a low temperature, the radial distribution is rectangular. When the temperature is increased and diffusion starts the infinite gradient at the outer boundary can only be approximated.

The numerical tests have shown that on average the URGAS algorithm takes approximately three times more computational time than the new FORMAS algorithm, due to the solution of the non-linear equation (11). On contrast, computer storage is roughly four times more for the new FORMAS algorithm.

The numerical errors have to be seen in the light of the many inherent uncertainties: The local temperature in a grain is certainly not exactly known, the grains are no spheres, differ in size and may vary due to many local irradiation effects. In view of these uncertainties, both algorithms are considered as sufficient to be used in a fuel performance code.

Taking all arguments together, we came to the simple solution of incorporating both algorithms into the TRANSURANUS code. We recommend using the URGAS algorithm for irradiations in which a low fission gas release $f < 0.05$ is expected and the new FORMAS algorithm else. The advantage of the new FORMAS algorithm for transients with high fission gas release is obvious. By comparing the results of both algorithms, the user may get an indication of the numerical errors encountered. These numerical errors must be compared with all other uncertainties.

All algorithms analysed are programmed as FORTRAN subroutines in double precision and are available on request. The corresponding subroutines have an identical argument list so that they can easily be exchanged.

Acknowledgements

The authors would like to thank R. Loth for his support of this work and A. Massih, T. Turnbull, C. Ronchi and I. Ray for many valuable suggestions.

References

[1] D.R. Olander, Fundamental aspects of nuclear fuel elements, Report TID-26711-P1, 1976.
[2] M.V. Speight, Nucl. Sci. Eng. 37 (1969) 180.
[3] T. Turnbull, personal communication, 1999.
[4] R.J. White, M.O. Tucker, J. Nucl. Mater. 118 (1983) 1.
[5] Hj. Matzke, personal communication, 1982.
[6] H.S. Carslaw, J.C. Jaeger, Conduction of Heat in Solids, Oxford University, London, 1959.
[7] A.H. Booth, A method of calculating fission gas diffusion from UO_2 fuel and its application to the X-2-f loop test, Report CRDC-721, 1957.
[8] C. Ronchi, J. Sakellaridis, C. Syros, Nucl. Sci. Eng. 95 (1987) 282.
[9] M. Gardani, C. Ronchi, Nucl. Sci. Eng. 107 (1991) 315.
[10] Y.R. Rashid, R.S. Dunham, Y.M. Lu, FREY-01: fuel rod evaluation system, volume 1: theoretical and numerical basis, Report EPRI NP-3277-CCM, 1983.
[11] V.I. Arimescu, Modelling stable-fission-gas diffusion inside the grain, in: IAEA Research Coordination Meeting on Fuel Modelling at Extended Burnup (FUMEX), 1–5 April 1996, Bombay, India (presented paper).
[12] C.S. Rim, Background and derivation of ANS-5.4 standard fission product release model, Report NUREG/CR-2507, 1982.
[13] W.N. Rausch, F.E. Panisco, ANS54: A computer subroutine for predicting fission gas release, Report NUREG/CR-1213, 1979.
[14] K. Lassmann, J. Nucl. Mater. 188 (1992) 295.
[15] P.T. Elton, K. Lassmann, J. Nucl. Mater. 101 (1987) 259.
[16] R.J. White, personal communication, 1996.
[17] K. Forsberg, A.R. Massih, J. Nucl. Mater. 127 (1985) 141.
[18] K. Forsberg, A.R. Massih, J. Nucl. Mater. 135 (1985) 140.
[19] H. Benk, K. Servaty, R. Loth, S. Ertem, M. Schäfer, K. Lassmann, Untersuchungen zur Berechnung der Gasfreisetzung im Brennstoff, Abschlußbericht Projekt-Nr. CR/10/97, 1998, Fachgebiet Energietechnik und Reaktoranlagen, Technische Universität Darmstadt, Fachgebiet Numerische Berechnungsverfahren im Maschinenbau, Technische Universität Darmstadt, Institut für Transurane, Internal Report, 1998.
[20] MATPRO-Version 11 (Revision 2), A handbook of material properties for use in the analysis of light water reactor fuel rod behaviour, Report NUREG/CR-0497, TREE-1280, Rev. 2, 1981.
[21] FRAPCON-2: A computer code for the calculation of steady state thermal–mechanical behaviour of oxide fuel rods, Report NUREG/CR-1845, R3, 1981.
[22] L. Väth, J. Nucl. Mater. 99 (1981) 324.
[23] J.R. Matthews, M.H. Wood, J. Nucl. Mater. 56 (1980) 439.
[24] H.J. Hedger, The calculation of gas diffusion from grains in fuel modelling, Report AERE-M3151, 1980.