ELSEVIER

International Journal of Pharmaceutics 120 (1995) 229–234

international journal of pharmaceutics

Fitting of diffusion coefficients in a three-compartment sustained release drug formulation using a genetic algorithm ☆

R. Hirsch, C.C. Müller-Goymann *

Institute of Pharmaceutical Technology, TU Braunschweig, Mendelssohnstr. 1, 38106 Braunschweig, Germany

Received 11 August 1994; revised 5 December 1994; accepted 15 December 1994

Abstract

This article presents a method of fitting diffusion coefficients in a three-compartment drug formulation to data of concentration measurements. The volume of the central compartment is not constant, but increases with time up to a certain amount. The speed of growth is proportional to the actual distance from the final thickness. A model function based on Fick's second law of diffusion is used to describe the concentration with respect to location and time. In order to find the values of the diffusion coefficients they are encoded to data structures on which the mechanisms of evolution can be applied: mutation and selection. It is shown how the convergence speed is influenced by the optimization parameters: the more individuals are involved in the evolution process, the fewer the generations it takes the algorithm to fit the parameters. There are optimal values for the rate of mutation ($m \approx 0.008$ bit^{-1}) and the selection factor, which controls the influence of selection in the mating process. Its optimal value is less than unity, which means that the algorithm converges faster when sometimes the genetic information of the weaker of two individuals is passed on to the next generation.

Keywords: Diffusion; Fick's second law; Lamellar liquid crystal; Sustained release; Diffusion coefficient; Curvilinear parameter fitting; Genetic algorithm

1. Introduction

Hamann describes a reverse micellar solution of lecithin in oil with a fascinating property (Hamann, 1990; Müller-Goymann and Hamann, 1993): when water comes into contact with the surface of this solution a lamellar mesophase grows at the interface between water and reverse micellar solution (Fig. 1).

The diffusion coefficient of the mesophase is about 100-times smaller than that of the solution and of water, respectively. Thus, the growing phase inhibits diffusion and may be used for sustained release.

The transport rate of drug molecules in the three-compartment system depends on the values of the diffusion coefficients in each compartment. The knowledge of this value is the key to the practical use of these formulations.

Since the diffusion coefficients cannot be measured directly, they must be obtained by comparison of the concentration function $c(x,t)$ to a

☆ Dedicated to Professor J. Klein on the occasion of his 60th birthday.

* Corresponding author.

SSDI 0378-5173(94)00428-5

model function $\hat{c}(x,t,D_1,D_2,D_3)$ which describes the concentration with respect to the diffusion coefficients.

The aim of this article is to present a genetic algorithm which fits the diffusion coefficients as parameters of the model function to experimental data of the concentration-time curves.

2. Model function

The mathematical model of the diffusion process in the three-compartment system is based on Fick's second law of diffusion:

$$\Delta c - \frac{1}{D(x,t)} \frac{\partial c}{\partial t} = 0, \tag{1}$$

where Δ is the Laplace operator $\Delta = \frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2} + \frac{\partial^2}{\partial z^2}$ and D the diffusion coefficient which is location and time dependent.

To achieve an unequivocal solution of this second-order partial differential equation the following presumptions have been made:

(1) One dimensionality: The concentration is homogeneous in a plane normal to the donor-acceptor axis (x-axis). Since there is no gradient of concentration in the y- or z-direction this yields diffusion along the x-axis only (Fig. 1):

$$\bigwedge_{-\infty \le x \le \infty} \bigwedge_{t \ge 0} \frac{\partial c}{\partial t} = \begin{cases} D_1 \dfrac{\partial^2 c}{\partial x^2} \text{ for } x < s(t), \\ D_2 \dfrac{\partial^2 c}{\partial x^2} \text{ for } s(t) \le x \le 0, \\ D_3 \dfrac{\partial^2 c}{\partial x^2} \text{ for } x > 0 \end{cases} \tag{2}$$

(2) Constancy of the diffusion coefficients: In each compartment the diffusion coefficients D_1–D_3 are constant with respect to location and time:

$$\bigwedge_{-\infty \le x \le \infty} \bigwedge_{t \ge 0} D_i(x,t) = D_i;\ i = 1,\ldots,3 \tag{3}$$

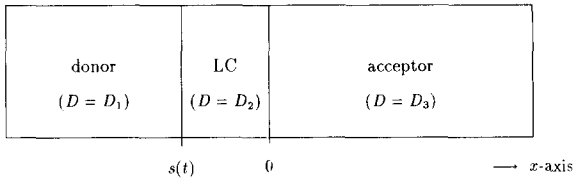

Fig. 1. The one-dimensional model of the three compartments. The donor consists of a reverse micellar solution, LC is the lamellar liquid crystalline phase and the acceptor is water. D_i represents the diffusion coefficient in each compartment.

Presuming a constant diffusion coefficient in the central compartment seems not to be justified, since the structure of the lamellar phase is not homogeneous along the x-axis. However, the calculation with an average diffusion coefficient in this compartment is sufficient for the purpose of developing an algorithm to fit the coefficients to a concentration function. Further investigations may be performed with other models without this limitation. There are numerical methods which can easily be adapted not only to diffusion coefficients that are concentration- or time-dependent, but also to more complex boundary conditions, e.g., (i) the finite-difference method (Smith, 1969) or (ii) the Crank-Nicholson implicit method (Crank, 1970). To study the convergence behaviour of parameter fitting algorithms, however, the use of a restricted but easy to compute model function is sufficient.

(3) Initial condition: At $t = 0$ the concentration is c_0 in the donor, and zero in the acceptor:

$$\bigwedge_{-\infty \le x \le \infty} c(x,0) = \begin{cases} c_0 \text{ for } x \le 0, \\ 0 \text{ for } x > 0 \end{cases} \tag{4}$$

(4) Boundary condition: Donor and acceptor do not have limits, i.e., the diffusion will never reach the physical boundaries:

$$\bigwedge_{t \ge 0} c(-\infty,t) = c_0, \tag{5}$$

$$\bigwedge_{t \ge 0} c(\infty,t) = 0 \tag{6}$$

(5) Steadiness: The concentration is steady with respect to location and time at the transition of the compartments (the partition coefficient between two compartments is assumed to be 1. This limitation can also be overcome by the use of numerical solutions of Eq. 1):

$$\lim_{\substack{x\to s(t)\\ x<s(t)}} c(x,t) = \lim_{\substack{x\to s(t)\\ x>s(t)}} c(x,t), \tag{7}$$

$$\lim_{\substack{x\to 0\\ x<0}} c(x,t) = \lim_{\substack{x\to 0\\ x>0}} c(x,t) \tag{8}$$

(6) Constancy of mass: The amount of mass in the whole system is constant, matter neither vanishing nor rising:

$$\frac{\partial}{\partial t}\int_{-\infty}^{\infty} c(x,t)\,\mathrm{d}x = 0 \tag{9}$$

(7) Growth function of the lamellar phase: The thickness of the rising lamellar phase is a function of time ($s(t)$). It is zero at $t = 0$ and grows asymptotically to a maximal value. The growing is assumed to be a Verhulst process:

$$s(t) = s_{\max}(1 - e^{-\lambda t}) \tag{10}$$

The parameters $s_{\max}$ and λ were obtained by curvilinear parameter fitting from experimental data with a threshold algorithm (Dueck et al., 1993).

Under these conditions the following solution of the diffusion equation was found by integration with the substitution $z := x/\sqrt{2D_i t}$:

$$c_{\mathrm{Don}}(x,t) = k_1(t)\int_{-\infty}^{\frac{x}{\sqrt{2D_1 t}}} e^{-\zeta^2/2}\,\mathrm{d}\zeta + c_0 \tag{11}$$

$$c_{\mathrm{Mes}}(x,t) = k_2(t)\int_{\frac{s}{\sqrt{2D_2 t}}}^{\frac{x}{\sqrt{2D_2 t}}} e^{-\zeta^2/2}\,\mathrm{d}\zeta + c_1(s,t) \tag{12}$$

$$c_{\mathrm{Acc}}(x,t) = k_3(t)\int_{\infty}^{\frac{x}{\sqrt{2D_3 t}}} e^{-\zeta^2/2}\,\mathrm{d}\zeta \tag{13}$$

with D_1, D_2, D_3 are the diffusion coefficients in each compartment and $s(t)$ denotes the thickness of the liquid crystalline phase at time t.

The values of $k_1(t)$, $k_2(t)$ and $k_3(t)$ are given by the following system of equations:

$$\begin{pmatrix} \sqrt{D_1}e^{s^2/(4D_1 t)} & -\sqrt{D_2}e^{s^2/(4D_2 t)} & 0 \\ 0 & \sqrt{D_2} & -\sqrt{D_3} \\ \left(1+\mathrm{erf}\left(\frac{s}{2\sqrt{D_1 t}}\right)\right) & -\mathrm{erf}\left(\frac{s}{2\sqrt{D_2 t}}\right) & 1 \end{pmatrix}$$

$$\begin{pmatrix} k_1 \\ k_2 \\ k_3 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ -\sqrt{\frac{2}{\pi}}\,c_0 \end{pmatrix} \tag{14}$$

3. Parameter fitting

The next step is the calculation of D_1, D_2, D_3 which describe the experimental data best after being applied to the model function. Following the suggestion of Gauß (1821), who took the sum of the squared differences of each experimental data point $c(x,t)$ and the estimated concentration value $\hat{c}(x,t,D_1,D_2,D_3)$ of the model function, results in:

$$Q(D_1,D_2,D_3) := \sum_{i=1}^{n} \left(c(x,t) - \hat{c}(x,t)\right)^2 \tag{15}$$

This function has one global minimum, and the set $\{D_1, D_2, D_3\}$ which belongs to this minimum is defined to be the optimal fitting to the experimental data.

Gauß's definition of the quality function Q is not the only one possible, but it is the most common. (It is justified, in fact, for normal distributed measurement data.) The algorithm below is not bound to this least-squares definition. We are free to define the sum of absolute differences, for instance, as a quality function and the algorithm will also work fine.

There are several possible methods to find that set of diffusion coefficients that fit the experimental data best. One way is to determine the zeros of the partial derivatives of the model function. However, this is not a real solution, because vanishing derivatives of the diffusion coefficients is a necessary condition for the global minimum of the model function, but not a sufficient condition – so the wrong minimum may be found.

However, it is very easy to find the zeros for the partial derivatives in functions that are low-order polynomials. Hence, the model function could be linearized piecewise (probability paper) and then a linear regression could be performed. This however neglects the errors of the measurements: The quality function was defined as the sum of the squared residuals (SSR) between data and model function, not as the residuals between curvilinear transformed data and transformed model function. Therefore, the resulting minimum does not fulfil the definition in Eq. 15 (Ebel et al., 1989).

Another method to find a minimum is the simplex method (Spendley et al., 1962). Under certain conditions, however, it fails to converge correctly: the simplex is cycling beside the minimum, not around it. An improvement of the simplex method, the Nelder-Mead algorithm with variable lengths of the simplex (Nelder and Mead, 1965), does find a minimum, but if the quality function is multimodal the result may not be the global minimum. These classical algorithms proceed from the starting position straightforwardly down to the next local minimum. They will never cross a maximum to find another minimum with lower value of the quality function Q.

Some stochastical methods have been developed to surmount local maxima, so that there is a certain possibility for these algorithms to end at the global minimum. If they are run repeatedly they will result in a set of minima of which the lowest is taken to be the global minimum. Examples are the flood algorithm, the threshold algorithm (Dueck et al., 1993) or genetic algorithms as described by Rechenberg (1973) and Schwefel (1975).

4. Genetic algorithm

To use the tools of evolution for mathematical minimization purposes, the mechanisms of mutation and selection have to be adopted to the problem mentioned above. For this purpose the genetic terms 'gene', 'individual', 'pool', 'mutation', 'crossing over' and 'selection' must be redefined.

A gene is the representation of one diffusion coefficient. It consists of a vector of n (16 say) bits:

$$\vec{g} := \{0,0,0,1,0,1,\ldots,0,1\} \tag{16}$$

(This vector represents a natural number between zero and $2^n - 1$ in binary notation.)

Since the diffusion coefficient may vary from 1.00×10^{-12} to 1.00×10^{-9} $m^2\ s^{-1}$ a logarithmic encoding scheme is used to represent the value of a diffusion coefficient in a gene. The expression of a gene is then defined as:

$$\mathrm{val}(\vec{g}) := 10^{-12+3(2^{-n}\vec{g}\cdot\vec{a})}\ m^2\ s^{-1}, \tag{17}$$

$$\vec{a} := (a_\nu),\ a_\nu = 2^{\nu-1},\ \nu = 1,\ldots n \tag{18}$$

An individual is a set of three genes, the diffusion coefficients of each of the three compartments. The optimisation process involves a certain amount of individuals, the pool.

The following biological mechanisms can be applied to this pool: Mutation: With probability m the value of each bit in a gene is changed to its complement ($m \approx 0.01$/bit). Mating: The genetic information of two parent encodings is mixed. Crossing over: The chains of bits of two parent encodings are broken and crosswise reassembled. Selection: Individuals with a low fitting criterion (sum of squared residuals) are less likely to be chosen for mating or crossing over. Spontaneous generation: Individuals can be generated from scratch (random gene sequences) to refresh the gene pool.

In the fitting program a population of k individuals is set up by spontaneous generation (Fig. 2). From this population two pairs of randomly chosen individuals are compared by the quality function corresponding to their encoded diffusion coefficients. The two individuals of each pair perform a rut fight, which the fittest of both wins with a probability s of about 85%. The genetic information of the two winners is mixed by crossing over and forms a new individual.

For the crossing over process a crossing point is chosen at random. The genome of each parental individual is broken at the crossing point, and the first part of the first individual and the second part of the second are concatenated to the child.

define parameters
- codelength
- pool size k
- rate of mutation m
- selection factor s

spontaneous generation of k individuals

compute fitting parameter (SSR) for each individual

WHILE convergence criterion not satisfied
- mate individuals
- mutate gens
- determine fitting parameters

print the results

Fig. 2. Structogram of the genetic algorithm.

Rut fight and crossing over are repeated k times, so that the child generation consists of as much individuals as the parental generation. (A constant number of individuals, however, is not necessary for this algorithm to work. The population may also grow and shrink during the evolution process.)

Now, after the child generation has been generated, with a certain probability m, the 'mutation rate', each bit of each gene of each individual in the pool becomes converted. Then the next generation is determined by selection, crossing over and mutation, and these processes cycle until no more improvement in the quality function is achieved over a certain amount of generations.

The algorithm was implemented in Mathematica (Wolfram Research), running on a HP-9000 Workstation, Model 715/33. The source code is available by email on request from r.hirsch@tu-bs.de.

5. Variation of fitting parameters

The algorithm described in the previous section is dependent of the length n of one single gene in bits, the pool size k, the probability s with which the fittest of two individuals wins the rut fight and the mutation rate m.

The length of a gene n implies the precision with which a diffusion coefficient can be encoded. The larger n is, the longer the calculation of one generation cycle takes. But n has no influence on the speed of convergence.

To show the dependence of the convergence speed on pool size k, selection factor s and mutation rate m, the number of generation cycles has been determined which were necessary to compute the best fitting diffusion coefficients to a given set of data points. The data were calculated from the model function and were not due to any error. So the algorithm was only halted when a zero sum of squared residuals was achieved (exact fitting). The values of the parameters k, s and m were varied.

The results are shown in Fig. 3: The convergence speed increases with the number of indi-

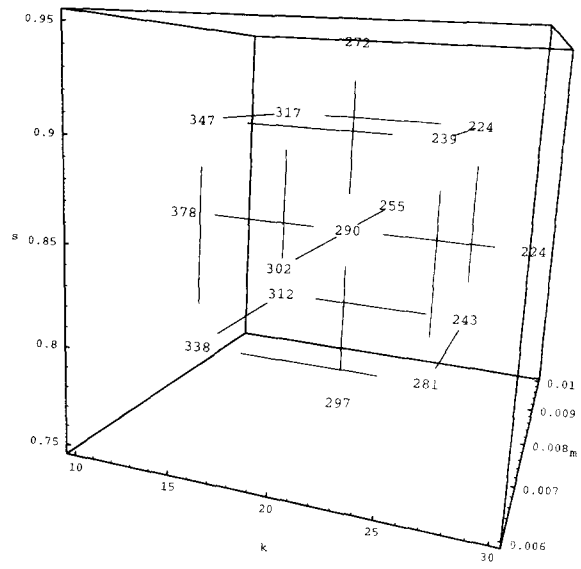

Fig. 3. Convergence of the genetic algorithm with respect to pool size k, selection factor s and mutation rate m. Shown are the numbers of generations needed to achieve convergence.

viduals k. However, it is not desirable to calculate with a very large population, because calculation time increases with the pool size. To find the optimum of k the CPU time could be used instead of the generation cycles.

The optimal mutation rate is found to be approx. 0.008 bit^{-1}. The result for the selection factor is approx. 0.85, which is surprising, because it is less than 1. That means that it is more desirable for the evolution process to carry sometimes the genetical information of a weak individual into the child generation than that of a fitter one.

References

Crank, J., *The Mathematics of Diffusion*, 2nd Edn, Clarendon Press, Oxford, 1970.

Dueck, G., Scheuer, T. and Wallmeier, H., Toleranzschwelle und Sintflut: Neue Methoden zur Optimierung, *Spektrum d. Wissenschaft*, März, 1993, pp. 42–51.

Ebel, S., Fleischer, P., Ledermann, M. and Wüstenhagen, U., Vorhersage der Stabilität von Arzneiformen aus der Kinetik der Abbaureaktion des Arzneistoffes: Teil 1: Auswertung kinetischer Daten. *Acta Pharm. Technol.*, 35 (1989) 210–217.

Gauß, C.F., *Theoria Combinationis observationum erroribus minimis obnoxiae*, 1821, Göttingen.

Hamann, H.-J., *Wechselwirkungen mesogener Arzneistoffe mit kolloidalen Lecithinassoziaten am Beispiel des Fenoprofens*, Braunschweig, 1990.

Müller-Goymann, C.C. and Hamann, H.-J., Sustained release from reverse micellar solutions by phase transformations into lamellar liquid crystals. *J. Controlled Release*, 23 (1993) 165–174.

Nelder, A. and Mead, R., A simplex method for function minimization. *Comput. J.*, 7 (1965) 308–313.

Rechenberg, I., *Evolutionsstratigie: Optimierung technischer Systeme nach Prinzipien der biologischen Evolution*, Friedrich Frommann, Stuttgart, 1973.

Schwefel, H., *Evolutionsstrategie und numerische Optimierung*, Berlin, 1975.

Smith, G.D., *Numerical Solution of Partial Differential Equations*, Oxford University Press, London, 1969, pp. 9–17.

Spendley, W., Hext, G.R. and Himsworth, F.R. Sequential Application of Simplex Designs in Optimisation and Evolutionary Operation. *Technometrics*, 4 (1962) 441–461.