Available online at www.sciencedirect.com
SCIENCE DIRECT®
European Journal of Pharmaceutics and Biopharmaceutics 56 (2003) 429–437
European Journal of Pharmaceutics and Biopharmaceutics
www.elsevier.com/locate/ejpb

Research paper

Handling of computational in vitro/in vivo correlation problems by Microsoft Excel: III. Convolution and deconvolution

Frieder Langenbucher*

BioVista LLC, Riehen, Switzerland

Received 29 July 2003; accepted in revised form 11 August 2003

Abstract

Convolution and deconvolution are the classical in-vitro-in-vivo correlation tools to describe the relationship between input and weighting/response in a linear system, where input represents the drug release in vitro, weighting/response any body response in vivo. While functional treatment, e.g. in terms of polyexponential or Weibull distribution, is more appropriate for general survey or prediction, numerical algorithms are useful for treating actual experimental data. Deconvolution is not considered an algorithm by its own, but the inversion of a corresponding convolution. MS Excel is shown to be a useful tool for all these applications.

Keywords: In-vitro-in-vivo correlation; Excel; Convolution; Deconvolution; Systems analysis; Polyexponentials; Laplace transforms; Point-area method; Trapezoidal method

1. Introduction

Two previous papers of this series dealt with the application of MS Excel to general in-vitro-in-vivo correlation (IVIVC) problems, in particular distribution functions [1,2]. The present paper discusses the mathematical relationship between drug release kinetics in vitro, and the corresponding body response in vivo, e.g. plasma concentration, urinary excretion, or any pharmacological effect. The intention is to summarize standard techniques and to show how these can be handled by means of MS Excel.

Convolution and deconvolution are standard mathematical tools for the analysis of linear systems, based on the validity of the superposition principle. Application to IVIVC problems dates back to the 1960's, e.g. by work of Silverman and Burgen [3], Rescigno and Segre [4], and Hanano [5]. Around 1980, several authors elaborated the technique, e.g. Vaughan and Dennis [6], Cutler [7–10], Veng-Pedersen [11–14], and the present author [15–19]. In several papers [20–24], Smolen et al. stressed the relationship with pharmacodynamic response. The usefulness of Excel was illustrated in two recent papers [25,26].

The technique as such became 'official' by guidelines of USP [27] and FDA [28]. According to these, it can be performed on three levels ranging from most to least informative, and briefly summarized as follows: (A) convolution and deconvolution of entire time profiles; (B) moments as metrics based on convolution relationships; (C) empirical metrics such as C_{max} or t_{max}. For the present state of art, the reader is referred to actual textbooks [29], monographs [30]; and relevant software packages [31,32].

2. Mathematical background

For any route of drug administration, a fastest possible 'bolus' input may be defined where all of the drug substance is available instantaneously. Obviously this bolus produces the fastest possible body response, whether this is the time profile of plasma concentration, urinary excretion, or any pharmacological effect.

* BioVista LLC, Ruechligweg 101, CH-4125 Riehen, Switzerland. Tel.: +41-61-603-2735; fax: +41-61-601-3474.
E-mail address: mysickaj@ivivc.com (F. Langenbucher).

0939-6411/$ - see front matter © 2003 Elsevier B.V. All rights reserved.
doi:10.1016/S0939-6411(03)00140-1

Route	'Bolus' input	'Delayed' input
Intravenous	Bolus injection	Infusion
Gastrointestinal	Aqueous oral solution	Tablet, capsule
Buccal	Aqueous buccal solution	Gum, lozenge
Transdermal	Aqueous or oily solution	Cream, TTS
Rectal	Aqueous or oily solution	Suppository

In contrast to the bolus, actual dosage forms are 'delayed' in that a release process contributes essentially to the input kinetics. In this context it plays no role whether the release is technically termed as 'immediate' or 'extended'. The only criterion is that the delayed input results in a delayed response.

2.1. Superposition principle

Backbone of input-response relationships is the superposition principle, well known from dosage regimen calculations. Based on two essential rules, it constitutes the system as linear.

- *Dose proportionality*: If an input $i(t)$ effects a response $r(t)$, a multiple or fraction $\varphi \times i(t)$ gives $\varphi \times r(t)$.
- *Time invariance:* If an input is given at time ϑ rather than 0, the response is the same but also shifted by ϑ.

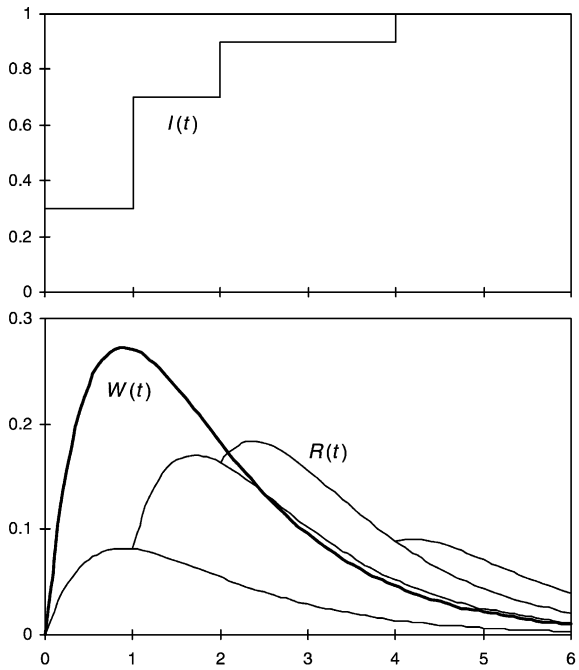
Hence, if several inputs are applied at various times, the total response is the sum of all partial responses, taking into account the time shifts between the inputs. This superposition is 'model-independent' in that no assumptions regarding compartments and transfers are involved.

Fig. 1 illustrates the principle and the computation in an Excel worksheet, for a weighting function assumed as

$$w(t) = 1.2(e^{-0.8t} - e^{-1.5t})$$

The first column TIME contains equidistant time values between 0 and the desired final time, with a common time step $\Delta t = 0.5$. A column WEI lists the corresponding values of the weighting function. Further columns, denoted as RES1 through RES4, contain the contributions of all input pulses; they are computed from values of WEI, shifted vertically by the time of each input, and multiplied by the corresponding dose fraction. The last column RES gives the total response as the sum of all fractional responses.

In the graphs, bold curves refer to a 'bolus' of size 1 at time 0; the corresponding response is the weighting function $w(t)$. Light curves represent the same dose, but distributed as a series of fractions as indicated in the table. The corresponding response $r(t)$ is obtained by adding the contributions of all input pulses. In addition all columns of r_i are computed as the sum of the previous column plus the actual input.



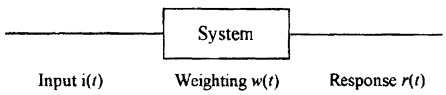
TIME	WEI	RES1 (0;0.3)	RES2 (1;0.4)	RES3 (2;0.2)	RES4 (4;0.1)	RES
0	0	0				0
0.5	0.238	0.071				0.071
1	0.271	0.081	0			0.081
1.5	0.235	0.070	0.095			0.166
2	0.183	0.055	0.109	0		0.163
2.5	0.134	0.040	0.094	0.048		0.182
3	0.096	0.029	0.073	0.054		0.156
3.5	0.067	0.020	0.054	0.047		0.121
4	0.046	0.014	0.038	0.037	0	0.089
4.5	0.031	0.009	0.027	0.027	0.024	0.087
5	0.021	0.006	0.018	0.019	0.027	0.071
5.5	0.014	0.004	0.013	0.013	0.023	0.054
6	0.010	0.003	0.009	0.009	0.018	0.039

Fig. 1. Illustration of the superposition principle. The charts show input (top) and response (bottom); the table illustrates computation in an Excel worksheet.

2.2. System analysis

Systems analysis (system theory) is the generalization of the superposition principle for connecting body response with continuous drug release. The technique is well established in other fields, e.g. biological cybernetics [33] or electronics engineering [34]. Although primarily devoted to linear systems, it may be expanded also to non-linear systems, see Appendix.

In IVIVC, the body system reacts to drug release $i(t)$ by a response $r(t)$, e.g. plasma concentration or urinary excretion of drug substance or a metabolite, or a pharmacological effect. The quantitative relationship between the three time profiles is depicted by this scheme:



Mathematically the relationship between them is given by the convolution integral

$$r(t) = \int_0^t i(\vartheta) w(t - \vartheta)\, \mathrm{d}\vartheta \quad (1)$$

In a shorthand notation, which stresses the analogy between convolution and ordinary multiplication, Eq. (1) and its possible inversions may be written as

$$r(t) = i(t)^* w(t) \quad (2a)$$

$$i(t) = r(t)//w(t) \quad (2b)$$

$$w(t) = r(t)//i(t) \quad (2c)$$

$$r_1(t) = i_2(t)^* [r_1(t)//i_1(t)] \quad (2d)$$

Here '*' denotes convolution, '//' the inverse operation of deconvolution which formally corresponds with ordinary division. Accordingly, Eq. (2a) is the equivalent of Eq. (1). Deconvolution estimates either input from Eq. (2b) or weighting from Eq. (2c), if the two others are known. A 'rule-of-three' conversion, Eq. (2d), permits to compute the response r_2 of a test product from i_1 and r_1 of a reference product. In IVIVC applications, all these functions are distribution functions with time t as the quantity whose distribution is described.

Response $r(t)$ and *weighting* $w(t)$ are necessarily of the same type and describe an in-vivo response of the body system in terms of the same physical quantity and identical units. Whereas $r(t)$ is the body response to any product with delayed input, $w(t)$ represents the borderline response after a bolus input, e.g. plasma concentration after an i.v. bolus or an oral solution.

- PDF: differential response, e.g. plasma concentration, gives $r(t) = i(t)^* w(t)$.
- CDF: cumulative response, e.g. cumulative urinary excretion or other pharmacodynamic response, gives $R(t) = i(t)^* W(t)$.

The input function $i(t)$ represents drug release of a dosage form; as observed in a predictive in vitro test (the terms 'release', 'dissolution' and 'delivery' may be considered as equivalent in the present context). It characterizes the time profile of the input (stimulus) as a dimensionless distribution function. In the convolution integral it is supplied as PDF; numerical algorithms may favor the use of its CDF analog $I(t)$, because in vitro release data are usually reported in this format.

3. Convolution and deconvolution methods

A fundamental problem of convolution is that it cannot be performed directly on observed experimental data, since these never meet the required criteria for the time intervals. In any case, interpolation or curve-fitting techniques are required. Predictions or surveys are best handled by mathematical *functions*, which emphasize the fundamental relationships and avoid the irregularities of experimental data; the functions and their parameters may be hypothetical or fitted to experimental data. On the other hand, actual data are better handled by *numerical* techniques using the raw data as far as possible; here all functions must be interpolated to a common time module, or some of them are used directly while others must be completely interpolated or fitted.

3.1. Functional convolution

Laplace transforms are most useful for any functional analysis [4,35]. Some selected transforms are shown in Table 1, where a time function $f(t)$ has a transform $f'(s)$ with the same set of parameters but a different functional form. A task difficult to solve in the time domain, is easily solved in the Laplace domain; back transformation into the time domain gives the desired result.

3.1.1. Polyexponential convolution

A simple situation arises if all functions involved are represented by polyexponential distributions. Let input and weighting functions be given as

$$i(t) = \sum_{j=1}^{m} a_j e^{-b_j t} \quad \Big| \quad i'(s) = \sum_{j=1}^{m} a_j/(s + b_j) \quad (3)$$

$$w(t) = \sum_{k=1}^{n} c_k e^{-d_k t} \quad \Big| \quad w'(s) = \sum_{k=1}^{n} c_k/(s + d_k) \quad (4)$$

where equations on the left apply to the time domain, those on the right to the Laplace domain. Convolution of both functions according to Eq. (2a) is replaced by multiplication

Table 1
Typical laplace transforms used in IVIVC

	Function $f(t)$	Transform $f'(s)$
Unit impulse		1
Unit step at $t = \tau$	0 ($t < \tau$); 1 ($t \geq \tau$)	$e^{-\tau s}/s$
Constant	A	a/s
Time lag t_0	$f(t - t_0)$	$f'(s) \exp(-t_0 s)$
Exponential	$a\, e^{-bt}$	$a/(s + b)$
	$a\, t\, e^{-bt}$	$a/(s + b)^2$
	$(a/b)(1 - e^{-bt})$	$a/[s(s + b)]$
Linearity	$a_1 f_1(t) + a_2 f_2(t)$	$a_1 f'_1(s) + a_2 f'_2(s)$
Integration	$\int^{0t} F(t)\, \mathrm{d}t$	$[1/s]\, f(s)$
Differentiation	$\mathrm{d}F/\mathrm{d}t$	$s\, f(s) - F_0$
Convolution	$f_1(t)^* f_2(t)$	$F'_1(s)\, f'_2(s)$

in the Laplace domain.

$$r(t) = i(t)^{*}w(t) \quad | \quad r'(s) = i'(s) \times w'(s) \tag{5}$$

If this multiplication is performed term by term of the relevant sums, it gives a total of $(m \times n)$ terms for $r'(s)$, each of which has this general format

$$\frac{ac}{(s+b)(s+d)} = \frac{ac/(b-d)}{s+d} - \frac{ac/(b-d)}{s+b} \tag{6}$$

The splitting of the left-hand expression into two additive terms is achieved by means of partial fraction decomposition, described in any advanced mathematical textbook, e.g. ref. [36]. It is easily verified by augmenting the numerator to read

$$ac = ac\,\frac{b-d}{b-d} = ac\,\frac{(s+b)-(s+d)}{b-d}$$

With these modifications, the left-hand expression Eq. (6) can be written as

$$= \frac{ac(s+b)/(b-d) - ac(s+d)/(b-d)}{(s+b)(s+d)}$$

Simplification by $(s+b)$ and $(s+d)$ gives the right-hand expression of Eq. (6).

Each of the two terms in Eq. (6) represents an exponential term of $r(t)$. As a result, rate constants are exactly those provided by $i(t)$ and $w(t)$. Coefficients are obtained by summing the contributions of the terms involved; each term $a_i c_j/(d_j - b_i)$ gives a positive contribution to b_i and a negative to d_j.

Fig. 2 shows an Excel worksheet [CONVEXPON], designed for a maximum of $m = n = 3$ terms of input and weighting functions; the extension of the scheme to more than three terms is obvious Example data are for a literature example [17], where input is a monoexponential $(m = 1)$, and weighting a biexponential $(n = 2)$; accordingly response is obtained with $(m + n) = 3$ terms:

$$i(t) = 0.41e^{-0.41t}$$

$$w(t) = \quad 12.37e^{-0.684t} + 7.86e^{-0.0725t}$$

$$r(t) = 8.96e^{-0.41t} - 18.51e^{-0.684t} + 9.55e^{-0.0725t}$$

In the Excel scheme, all data are arranged in a central block, bordered by double lines, and additional fields surrounding this block. Coefficients and rate constants of the input function are entered in two columns on the left, parameters of the weighting function in two rows on the top. The central block itself contains an array formula = RC1*R1C/(RC2 − R2C), which computes the expression $ac/(b-d)$ for any two terms in question. The coefficients of the response function are computed by vertical and horizontal summation. Below the central block, the formula = SUM(R3C:R5C) computes the rate constants stemming from $w(t)$. To the right, coefficients stemming from $i(t)$ are computed from the formula = − SUM(RC3:RC5).

	$w(t)\rightarrow$	12.37	7.858	0	c_j
$i(t)\downarrow$		0.684	0.0725	0	d_j
0.41	0.41	-18.510	9.546	-	8.964
0	0	0	0	#DIV/0!	#DIV/0!
0	0	0		#DIV/0!	#DIV/0!
a_i	b_i	-18.510	9.546	#DIV/0!	$\leftarrow r(t)\uparrow$

Fig. 2. Excel scheme for polyexponential convolution, designed for a maximum 3 + 3 = 6 terms. Parameters of input and weighting functions are entered to the left and on top of the central block computing the expression $ac/(b-d)$; coefficients of the response function are computed as sums to the right and below. Error codes indicate missing values.

In all data cells, non-existing terms may be entered as '0' or left blank. In this case, Excel will find that two rate constants are equal, and will return an error code. To avoid this error, arbitrary but differing rate constants could be specified for all missing terms.

3.1.2. Application examples

A typical application of polyexponential convolution is the prediction of in vivo response for hypothetic product formulations, where the response to an oral solution is expressed as a polyexponential, the same way as all in vitro release profiles. Fig. 3 illustrates such an 'input/response

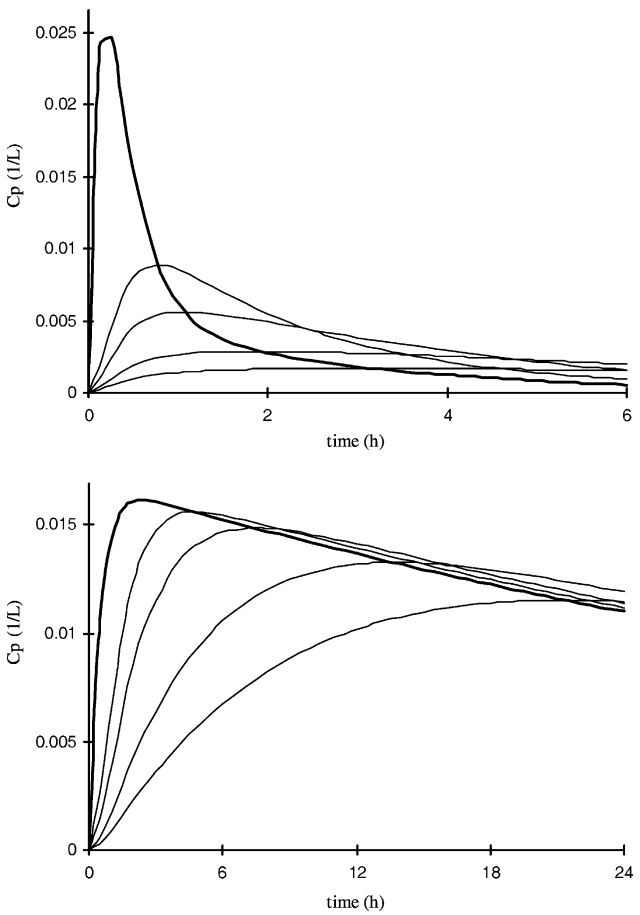


Fig. 3. Plasma input/response synopsis for an oral solution and four delayed tablets with MDT = 1, 2, 5, and 10 h, for two drug substances with extreme disposition characteristics: diclofenac (top) as fast and carbamazepine (bottom) as slow.

synopsis', i.e. the predicted response to a large variety of possible formulations with mean dissolution times MDT = 1–10 h, as summarized by these mean times:

	MDT	0	1	2	5	10
Diclofenac	MRT	0.8	1.8	2.8	5.8	10.8
Carbamazepine	MRT	55	56	57	60	65

Note that diclofenac as fast-disposing substance (MET ≈ 0.8 h) shows a different pattern than carbamazepine (MET ≈ 55 h) as a slow-reacting substance.

A special application is the correlation according to Eq. (2d) for a 'reference' and two 'side' batches differing in vitro by a small factor of ±20%. Such relationships are helpful to estimate changes due to instability, manufacturing processes, equipment, site of production, or hypothetical release specifications.

3.2. Numerical convolution

Numerical algorithms are close to the superposition principle in that discrete pulses of the input $i(t)$, in combination with points or segments of a weighting function $w(t)$, add-up to points or segments of the response $r(t)$. According to the nature of convolution, all time intervals involved must be consistent, i.e. with observations supplied at equal time points ('nodes'). Since this condition is never met by actual data, all numerical techniques require the generation of additional nodes. In the order of increasing smoothing, these may be obtained from interpolation, either linear or polynomial or by means of splines. A maximum of smoothing is achieved by representing the entire time profile by a prescribed (fitted) distribution function [7,8,11–14].

Methods based on equal time steps are attractive in that the computation scheme is simple; but at the cost of excessive data interpolation. Methods with unequal time steps are more general and convenient to apply to actual raw data, since they avoid excessive interpolation.

The response is generally computed as a series of *points* r_k of the response function $r(t)$, at prescribed time points. The interpretation of the two other functions is less obvious, and various techniques may be used to interpolate them between the observation nodes. Two approaches are widely used.

- *Point-area* methods, e.g. ref. [6], interpret the values of $i(t)$ and $w(t)$ as trapezoidal *areas*. Numerically this is equivalent to the interpretation of *means* or *midpoint* values.
- *Trapezoidal* methods, e.g. ref. [34], interpret $i(t)$ as a series of *pulses* at the beginning of each interval including $t = 0$. The product function $i(t) \times w(t)$ is interpreted as a straight line in each interval.

The overall performance of both techniques seems to be comparable [17,18], but the point-area approach provides a simpler computational structure. While these methods rely on linear interpolation, more sophisticated interpolation techniques may be used, but their advantage is questionable.

3.2.1. Unequal time steps

A particularly useful time schedule is based on the in-vivo functions $w(t)$ and/or $r(t)$, since these are supplied with a few unequally spaced time points, and with rather irregular time profiles difficult to interpolate or represent by functions. In contrast, input data, i.e. observed release in vitro, are more suitable as prescribed (fitted) distribution function, since usually recorded in cumulative format $I(t)$, with less scatter and a more regular profile, and with as many data points as desired.

According to the convolution integral of Eq. (1), the generalized numerical algorithm can be written as

$$\begin{aligned} r_0 &= 0 \\ r_1 &= I|_0^1 \bar{w}|_0^1 \\ r_2 &= I|_0^1 \bar{w}|_1^2 + I|_1^2 \bar{w}|_0^1 \\ r_3 &= I|_0^1 \bar{w}|_2^3 + I|_1^2 \bar{w}|_1^2 + I|_2^3 \bar{w}|_0^1 \\ &\vdots \\ r_k &= I|_0^1 \bar{w}|_{k-1}^k + I|_1^2 \bar{w}|_{k-2}^{k-1} + \cdots + I|_{k-1}^k \bar{w}|_0^1 \end{aligned} \tag{7}$$

where the index $k = 0, 1, 2\ldots$ denotes discrete interpolation nodes representing either actual observations, interpolated values or values computed from a prescribed function. The index $k = 0$ uniquely indicates the origin $t = 0$, for which the response is $r_0 \equiv 0$ by definition. Further values r_k, are calculated by a triangular arrangement of terms, where each row computes a time point r_k as the sum of all inputs prior to k, multiplied by a corresponding interval of $w(t)$; the interval between two consecutive nodes is denoted by lower and upper indices. Hence, each term in Eq. (7) is defined as a product of two corresponding intervals of $i(t)$ and $w(t)$, in convolved (reversed) sequence and supplied at consistent time points.

According to the point-area approach, relevant terms are defined as follows.

$$\bar{w}|_{k-1}^k = (w_{k-1} + w_k)/2 \tag{8}$$

$$\Delta I|_{t_{l-1}}^{t_l} = I(t_l) - I(t_{l-1}) \tag{9}$$

For $w(t)$, an averaged value is computed by the trapezoidal formula; if desired, other formulas such as spline functions can be used. Input data are supplied as fractional input between t_{l-1} and t_l computed from $i(t)$ representing release in vitro, e.g. Weibull or polyexponential.

The computation in an Excel worksheet [CONVNUM] is illustrated in Table 2 and Fig. 4, for the example already used in Fig. 2 for polyexponential convolution. A data table lists the observations and calculated results. The rows are

Table 2
Numerical convolution according to Eq. (7), illustrated for three unequally spaced nodes at $r_1 = 0.5$, $r_2 = 1.5$, and $r_3 = 3$, with intervals of 0.5, 1, and 1.5

Node	TIME		Interval	ICUM	Interval	WAVE	ICUM × WAVE	RCALC
1	0.5	#1	0.5 − 0.0	0.1854 − 0.0000 = 0.1854	0.5 ÷ 0.0	16.364 ÷ 20.227 = 18.295	3.392	3.392
2	1.5	#1	0.5 − 0.0	0.3364 − 0.0000 = 0.3364	1.5 ÷ 1.0	11.481 ÷ 16.364 = 13.923	4.684	
		#2	0.5 − 1.5	0.4594 − 0.3364 = 0.1230	1.0 ÷ 0.0	16.364 ÷ 20.227 = 18.295	2.251	6.935
3	3	#1	0.5 − 0.0	0.4594 − 0.0000 = 0.4594	3.0 ÷ 2.5	7.913 ÷ 11.481 = 9.697	4.455	
		#2	1.5 − 0.5	0.6412 − 0.4594 = 0.1818	2.5 ÷ 1.5	11.481 ÷ 16.364 = 13.923	2.531	
		#3	3.0 − 1.5	0.7077 − 0.6412 = 0.0665	1.5 ÷ 0.0	16.364 ÷ 20.227 = 18.295	1.217	8.203

For WAVE, the symbol ÷ denotes the average in the relevant time interval.

given by the observations, i.e. the time points in column TIME, and the corresponding weighting data in WOBS; these points constitute the overall nodes used in the computation. A column ROBS is reserved for observed response data to be used for deconvolution according to Eq. (2b). Further columns of the data table contain computed values:

```
ICUM   {= WEIBULL(TIME;SHAPE;SCALE;1)}
WAVE   = (R[−1]C4 + RC4)/2
RCALC  {= TRANSPOSE(SUM)}
```

ICUM computes the cumulative input $I(t)$ from a prescribed function, e.g. a Weibull distribution with parameters supplied in fields SCALE and SHAPE. WAVE computes the weighting averages from the observations in WOBS, according to Eq. (8). RCALC returns calculated $R(t)$ values from a separate table WTRI. The table WTRI performs the convolution calculation according to Eqs. (7)–(9). It represents a triangular matrix, with rows and columns corresponding to the integration nodes as represented by the data table. The relevant time values are copied as table headings into the row above and the column left to the matrix table. All cells contain the identical Excel formula

```
WTRI = (R23C >= RC1)*R[−17]C5*
       (WEIBULL(ABS(R23C-R[−1]C1);
       SHAPE;SCALE;1) −WEIBULL
       (ABS(R23C-RC1);SHAPE;SCALE;1))
```

which computes differences of cumulative input from the prescribed function, multiplied by the corresponding WAVE = R[−17]C5 for the interval in question. The factor (R23C >= RC1) assures that values are computed only for the upper-right half. Column sums of WTRI are computed in the row SUM below the table, and are copied from there into the column RCALC of the data table.

3.2.2. Equal time steps

Convolution methods based on equal time steps are attractive in that the underlying computation is closer to the underlying superposition principle, hence easier to visualize. Accordingly, point-area [6] as well as trapezoidal [34] techniques have been proposed in the early literature, the overall performance of which was shown to be comparable [17,18]. For computation in an Excel worksheet, these approaches suffer from the need to interpolate all functions according to a smallest common time module DELT, from time 0 up to a maximum FINTIM. In a typical case with DELT = 0.5 and FINTIM = 24, this requires a total of 48 nodes, where only eight are true observations and 40 are interpolated. A triangular 48 × 48 matrix is required to handle these data.

Of course, convolution with equal time steps can be handled the same way as the case of unequal steps discussed above. A simpler technique is provided by matrix multiplication, as illustrated by the Falk schemes in Fig. 5. In both methods, an Excel formula such as

```
RCALC = DELT*MMULT(WTRI;INP)
```

computes the vector of response values, where DELT denotes the common time step. For *point-area*, input data are supplied as a column vector INP representing averaged midpoint values. Weighting data are supplied as a true triangular matrix WTRI, generated from the corresponding 'data', and representing average values for the individual intervals.

In the *trapezoidal* approach, INP is simpler in that it represents the 'data' as such. On the other hand, WTRI has one row and column more than a triangular matrix, and is

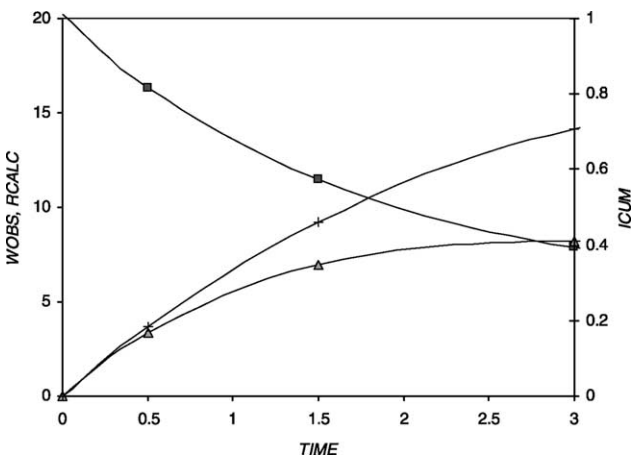


Fig. 4. Time profiles according to Table 2.

Point-area method

$r_0 \equiv 0$

$r_1 = (w_0 i_0 + w_0 i_1 + w_1 i_0 + w_1 i_1)/4$

				$(i_0+i_1)/2$
				$(i_1+i_2)/2$
				$(i_2+i_3)/2$
				$(i_3+i_4)/2$
$(w_0+w_1)/2$	0	0	0	r_1
$(w_1+w_2)/2$	$(w_0+w_1)/2$	0	0	r_2
$(w_2+w_3)/2$	$(w_1+w_2)/2$	$(w_0+w_1)/2$	0	r_3
$(w_3+w_4)/2$	$(w_2+w_3)/2$	$(w_1+w_2)/2$	$(w_0+w_1)/2$	r_4

Trapezoidal method

$r_0 \equiv 0$

$r_1 = (w_0 i_1 + w_1 i_0)/2$

					i_0
					i_1
					i_2
					i_3
					i_4
$(w_1)/2$	$(w_0)/2$	0	0	0	r_1
$(w_2)/2$	w_1	$(w_0)/2$	0	0	r_2
$(w_3)/2$	w_2	w_1	$(w_0)/2$	0	r_3
$(w_4)/2$	w_3	w_2	w_1	$(w_0)/2$	r_4

Fig. 5. Falk schemes for numerical convolution with equal time steps, according to point-area and trapezoidal methods.

composed from a mixture of direct and halved terms. This complex structure makes handling of this algorithm more difficult.

3.3. Deconvolution algorithms

Mathematically, deconvolution is an inverse 'trial-and-error' algorithm related to convolution, the same way that subtraction, division and differentiation are inverse operations of addition, multiplication and integration. Strangely enough, most of the relevant IVIVC literature, scientific papers as well as commercial software, focus on deconvolution, as if this were a mathematical operation by itself, without any regard to convolution.

As for any inverse operation, the main problem of deconvolution is its inherent numerical instability. When treating real data, the experimental errors will increasingly affect all computational steps; from a certain limit results start to oscillate and become invalid. This instability seems to be the reason that by far more methods deal with deconvolution than with convolution. A recent paper [37] compared the performance of six deconvolution algorithms.

3.3.1. 'Procedural' algorithms

These algorithms work 'stepwise' in that each observation node is estimated exactly, on the base of all previous steps. Obviously these algorithms are particularly prone to error propagation and unstable results. A method to ensure numerical stability, proposed early by Kiwada et al. [38], is to suppress all meaningless steps and to replace them by '0'. Since all relevant functions are restricted to non-negative values, any negative result is handled this way. Likewise, if the function to be estimated is monotonically increasing, e.g. cumulative input, any step with negative increase is suppressed. The technique delimits the error to a 'reasonable' value, improves the starting condition for the next step, and avoids wild oscillations.

3.3.2. 'Least-squares' algorithms

Algorithms designed to find an 'overall' solution by fitting all data points according to a criterion such as the minimized sum of squared differences, provide a better approach [7,8]. The Excel's Solver tool supports such an least-squares algorithm. In fact, any valid convolution Excel worksheet designed to compute a column RCALC = IOBS*WOBS, can be adapted to deconvolution by adding two features:

- A column ROBS next to the column RCALC is used to enter the given response data;
- A target cell SSQ = SUMXMY2(RCALC;ROBS) computes the sum of the squared differences between RCALC and ROBS.

With these modifications, the Solver tool can be invoked to minimize SSQ by varying values in either IOBS or WOBS, whatever is the range to be estimated. As result, the Solver replaces initial guesses in the relevant column by fitted values. Typically the input is estimated according to Eq. (2b), likewise the weighting function may be estimated according to Eq. (2c).

This algorithm is based on established convolution methods and requires no special deconvolution algorithms: If the profile to be estimated is represented by data points, these are fitted; if represented by a function, the parameters of this function are optimized; for a poly-exponential this may include adding or dropping of complete terms.

When fitting a particular time profile, the Solver may be invoked more than once, in order to improve the actual solution, with altered specifications. During execution the user has complete control over the entire algorithm. He may start with '0' or '1' as initial guesses for all relevant estimates, but he may also supply more intelligent guesses to improve the computational performance. He may decide to optimize all estimates together in a single iteration; if this does not work, he may start with single estimates or reasonable groups of them. Once a satisfactory fit is found, all relevant estimates should be varied in a final step, to achieve overall optimization.

Appendix

A.1. Non-linear systems

The principles of superposition and systems analysis primarily deal with linear systems exhibiting dose proportionality and time invariance. Although this requirement is never met completely in biological systems, it provides

a sound approximation in most applications. If the system is essentially non-linear, e.g. the body response to alcohol input, the simplified linear treatment would fail to give reasonable results with respect to plasma concentration or physiological effects of drunkenness.

The principles of systems analysis can be expanded also to essentially non-linear cases. According to the complexity of static and dynamic non-linearities in biological systems, no general closed mathematical solutions are available at present, and individual solutions must be found instead. While a detailed discussion is certainly beyond the scope of the present paper, the interested reader is referred to a few selected publications dealing with this subject. Cruse [33] discussed the problem in the context of biological cybernetics, which makes full use of corresponding applications in electronic systems. In the IVIVC literature, these techniques were stressed in the work of Smolen [20–24], who correlated drug input with pharmacodynamic response by means of established control-engineering techniques. More recently, Gillespie [39] discussed two special cases of non-linearity, namely 'truncated absorption' and 'saturable presystemic elimination'.

A.2. Notation

General notation is the same as in previous papers of this series. Differential time functions (PDF) are denoted as $f(t)$, cumulative functions (CDF) as $F(t)$. Correspondingly, differential input, weighting and response functions are denoted as $i(t)$, $w(t)$, $r(t)$, cumulative functions as $I(t)$, $W(t)$, $R(t)$. In the context of Laplace transformation, transformed functions are distinguished by apostrophes, e.g. $f'(s)$, $i'(s)$, $w'(s)$, $r'(s)$, where s is the transformed time variable.

References

[1] F. Langenbucher, Handling of computational IVIVC problems by Microsoft Excel: I. Principles and some general algorithms, Eur. J. Pharm. Biopharm. 53 (2002) 1–7.

[2] F. Langenbucher, Handling of computational IVIVC problems by Microsoft Excel: II. Distribution functions and moments, Ibid 55 (2003) 77–84.

[3] M. Silverman, S.V. Burgen, Application of analogue computer to measurement of intestinal absorption rates with tracers, J. Appl. Physiol. 16 (1961) 911–913.

[4] A. Rescigno, G. Segre, Drug and Tracer Kinetics, Blaisdell Publishing Co, Waltham, MA, 1966.

[5] M. Hanano, Studies on absorption and excretion of drug. VII: a new estimation method for the release of drugs from dosage forms and the availability in vivo, Chem. Pharm. Bull. 15 (1967) 994–1001.

[6] D.P. Vaughan, M. Dennis, Mathematical basis of point-area deconvolution method for determining in vivo input functions, J. Pharm. Sci. 67 (1978) 663–665.

[7] D.J. Cutler, Numerical deconvolution by least squares: use of prescribed input functions, J. Pharmacokin. Biopharm. 6 (1978) 227–241.

[8] D.J. Cutler, Use of polynomials to represent the input function, Ibid 6 (1978) 243–263.

[9] D.J. Cutler, Linear systems analysis in pharmacokinetics, J. Pharmacokin. Biopharm. 6 (1978) 265–282.

[10] D.J. Cutler, Assessment of rate and extent of drug absorption, Pharmacol. Ther. 14 (1981) 123–160.

[11] P. Veng-Pedersen, Model-independent method of analyzing input in linear pharmacokinetic systems having polyexponential impulse response. I. Theoretical analysis, J. Pharm. Sci. 69 (1980) 298–305.

[12] P. Veng-Pedersen, Model-independent method of analyzing input in linear pharmacokinetic systems having polyexponential impulse response. II. Numerical evaluation, Ibid 69 (1980) 305–312.

[13] P. Veng-Pedersen, Novel deconvolution method for linear pharmacokinetic systems with polyexponential impulse response, J. Pharm. Sci. 69 (1980) 312–318.

[14] P. Veng-Pedersen, An algorithm and computer program for deconvolution in linear pharmacokinetics, J. Pharmacokin. Biopharm. 8 (1980) 463–481.

[15] F. Langenbucher, Numerical convolution/deconvolution as a tool for correlating in vitro with in vivo drug availability, Pharm. Ind. 44 (1982) 1166–1172.

[16] F. Langenbucher, Improved understanding of convolution algorithms correlating body response with drug input, Pharm. Ind. 44 (1982) 1275–1278.

[17] F. Langenbucher, H. Moeller, Correlation of in vitro drug release with in vivo response kinetics. I. Mathematical treatment of time functions, Pharm. Ind. 45 (1983) 623–628.

[18] F. Langenbucher, H. Moeller, Correlation of in vitro drug release with in vivo response kinetics. II. Use of function parameters, Ibid 45 (1983) 629–633.

[19] F. Langenbucher, J. Mysicka, In vitro and in vivo deconvolution assessment of drug release kinetics from oxprenolol oros preparations, Br. J. Clin. Pharmacol. 19 (1985) 151S–162S.

[20] V.F. Smolen, W.A. Weigand, Optimally predictive in vitro drug dissolution testing for in vivo bioavailability, J. Pharm. Sci. 65 (1976) 1718–1724.

[21] V.F. Smolen, Theoretical and computational basis for drug bioavailability determination using pharmacological data. I. General considerations and procedures, J. Pharmakin. Biopharm. 4 (1976) 337–353.

[22] V.F. Smolen, Theoretical and computational basis for drug bioavailability determination using pharmacological data. II. Drug input ↔ response relationships, Ibid 4 (1976) 355–375.

[23] V.F. Smolen, R.J. Erb, Predictive conversion of in vitro drug dissolution data into in vivo drug response versus time profiles exemplified for plasma levels of warfarin, J. Pharm. Sci. 66 (1977) 287–304.

[24] V.F. Smolen, Practical pharmacodynamic engineering in the design, development and evaluation of optimal drug products, in: H.S. Bean, A.H. Beckett, J.E. Carless (Eds.), Advances in Pharmaceutical Sciences, Academic Press, London, 1982.

[25] A. Deslandes, J.F. Westphal, J.H. Trouvin, R. Farinotti, Adaptive computer program for determination of absorption profiles by numerical deconvolution: application to amoxicillin absorption, J. Pharm. Sci. 81 (1992) 802–807.

[26] P. Buchwald, Direct, differential-equation-based in-vitro-in-vivo correlation (IVIVC) method, J. Pharm. Pharmacol. 55 (2003) 495–504.

[27] USP23/NF18 (1995): ⟨1088⟩. In vitro and in vivo evaluations of dosage forms, 924–929.

[28] FDA, Guidance for Industry: extended release oral dosage forms: development, evaluation and application of in vitro/in vivo correlations, Diss. Technol. (1997) 23–32.

[29] J. Gabrielsson, D. Weiner, Pharmacokinetic and Pharmacodynamic Data Analysis: Concepts and Applications, Swed. Pharm. Press, Stockholm, Sweden, 1997.

[30] D. Young, J.G. Devane, J. Butler (Eds.), In Vitro-In Vivo Correlations, Plenum Press, New York, 1997.

[31] WinNonlin 3.1, Pharsight Corp., Mountain View, CA, 1999.

[32] Kinetika 2000, Innaphase Corp., Philadelphia, PA, 2000.
[33] H. Cruse, Biologische Kybernetik: Einführung in die lineare und nichtlineare Systemtheorie, Verlag Chemie, Weinheim, FRG, 1981.
[34] E. Stepanek, Praktische Analyse linearer Systeme durch Faltungsoperationen, Geest & Portig KG, Leipzig, DDR, 1976.
[35] H. Margenau, G.M. Murphy, The Mathematics of Physics and Chemistry, D. van Nostrand Co, Princeton, NJ, 1956.
[36] A. Jeffrey, Handbook of Mathematical Formulas and Integrals, Academic Press, San Diego, CA, 1995, pp. 60–62.
[37] F.N. Madden, K.R. Godfrey, M.J. Chappell, R. Hovorka, R.A. Bates, A comparison of six deconvolution techniques, J. Pharmacokin. Biopharm. 24 (1996) 283–299.
[38] H. Kiwada, K. Morita, M. Hayashi, S. Awazu, M. Hanano, A new numerical calculation method for deconvolution in linear compartment analysis of pharmacokinetics, Chem. Pharm. Bull. 25 (1977) 1312–1318.
[39] W.R. Gillespie, Convolution-based approaches for in vivo-in vitro correlation modeling. In [25], (1997) 53–65.